Cisco Catalyst 9800 Series Wireless Controller Software Configuration Guide, Cisco IOS XE 26.x

Cisco Catalyst 9800 Series Wireless Controllers
Updated October 7, 2026

© 2026 Cisco Systems, Inc. All rights reserved.

Full Cisco Trademarks with Software License

Full Cisco Trademarks with Software License

THE SPECIFICATIONS AND INFORMATION REGARDING THE PRODUCTS IN THIS MANUAL ARE SUBJECT TO CHANGE WITHOUT NOTICE. ALL STATEMENTS, INFORMATION, AND RECOMMENDATIONS IN THIS MANUAL ARE BELIEVED TO BE ACCURATE BUT ARE PRESENTED WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. USERS MUST TAKE FULL RESPONSIBILITY FOR THEIR APPLICATION OF ANY PRODUCTS.

THE SOFTWARE LICENSE AND LIMITED WARRANTY FOR THE ACCOMPANYING PRODUCT ARE SET FORTH IN THE INFORMATION PACKET THAT SHIPPED WITH THE PRODUCT AND ARE INCORPORATED HEREIN BY THIS REFERENCE. IF YOU ARE UNABLE TO LOCATE THE SOFTWARE LICENSE OR LIMITED WARRANTY, CONTACT YOUR CISCO REPRESENTATIVE FOR A COPY.

The Cisco implementation of TCP header compression is an adaptation of a program developed by the University of California, Berkeley (UCB) as part of UCB's public domain version of the UNIX operating system. All rights reserved. Copyright © 1981, Regents of the University of California.

NOTWITHSTANDING ANY OTHER WARRANTY HEREIN, ALL DOCUMENT FILES AND SOFTWARE OF THESE SUPPLIERS ARE PROVIDED “AS IS" WITH ALL FAULTS. CISCO AND THE ABOVE-NAMED SUPPLIERS DISCLAIM ALL WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING, WITHOUT LIMITATION, THOSE OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT OR ARISING FROM A COURSE OF DEALING, USAGE, OR TRADE PRACTICE.

IN NO EVENT SHALL CISCO OR ITS SUPPLIERS BE LIABLE FOR ANY INDIRECT, SPECIAL, CONSEQUENTIAL, OR INCIDENTAL DAMAGES, INCLUDING, WITHOUT LIMITATION, LOST PROFITS OR LOSS OR DAMAGE TO DATA ARISING OUT OF THE USE OR INABILITY TO USE THIS MANUAL, EVEN IF CISCO OR ITS SUPPLIERS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

Any Internet Protocol (IP) addresses and phone numbers used in this document are not intended to be actual addresses and phone numbers. Any examples, command display output, network topology diagrams, and other figures included in the document are shown for illustrative purposes only. Any use of actual IP addresses or phone numbers in illustrative content is unintentional and coincidental.

All printed copies and duplicate soft copies of this document are considered uncontrolled. See the current online version for the latest version.

Cisco has more than 200 offices worldwide. Addresses and phone numbers are listed on the Cisco website at www.cisco.com/go/offices.

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of Cisco trademarks, go to this URL: https://www.cisco.com/c/en/us/about/legal/trademarks.html. Third-party trademarks mentioned are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1721R)

Topics included

1 Overview of the Controller....79
Cisco 9800 Series Wireless Controllers....80
Configuration data model....80
Initial controller setup methods....80
Restrictions for initial controller setup....81
Interactive help....82
Cisco Support Assistant....82
Product analytics....83

Part I: System Configuration....87

2 New Configuration Model....89
New configuration model, tags, and profile....90
Configuration workflow....90
Tags and their associated profiles....92
RF Tags....92
RF Profiles....93
Site Tags....93
Flex profiles....94
AP join profiles....94
Policy Tag....94
WLAN profiles....95
Policy profiles....95
Methods for associating APs with tags....96
Configure a wireless profile policy (GUI)....97
Configure a wireless policy profile (CLI)....98
Configure a flex profile (GUI)....99
Configure a flex profile....100
Configure an AP profile (GUI)....101
Configure an AP profile (CLI)....106
Configure user for AP management (CLI)....108
Set a private configuration key for password encryption....108
Configure an RF profile (GUI)....110
Configure an RF profile (CLI)....110
Configure a site tag (GUI)....112
Configure a site tag for FlexConnect APs (CLI)....112
Configure policy tag (GUI)....113
Configure a policy tag (CLI)....114

Configure wireless RF tag (GUI)....116
Configure wireless RF tag (CLI)....116
Attach a policy tag and site tag to an AP (GUI)....117
Attach policy tag and site tag to an AP (CLI)....118
Enhanced site tag-based load balancing....119
Prerequisites for Enhanced Site Tag-Based Load Balancing....120
Use cases....121
Configure site load (CLI)....121
Verify enhanced site tag-based load balancing configuration....122
Wireless radio profiles....122
Restrictions....123
Configure a wireless radio profile (GUI)....124
Configure a radio profile and beam selection....124
Configure the number of antennas in a wireless radio profile....125
Configure a slot for each radio in the RF tag profile....126
Verify a radio profile....126
AP filters....128
Set tag priority (GUI)....128
Set tag priority 128
Create an AP filter (GUI)....129
Create an AP filter (CLI)....130
Configure filter priority (GUI)....131
Configure filter priority (CLI)....131
Verify AP filter configuration 132
Location configuration....133
Restriction for location configuration....133
Configure a location for an AP (GUI)....133
Configure a location for an AP (CLI)....134
Add an AP to location (GUI)....135
Adding an Access Point to the Location (CLI)....136
SNMP MIB....137
Verify location configuration....137
Verify location statistics....138

3 Wireless Management Interface....139
Wireless management interfaces....140
Best practices and restrictions for WMI....140
Change the WMI interface when RMI is configured....141
WMI Configuration Tasks....141
Migrate VLANs safely for wireless management....141
Sample interface configuration....142
Configure the WMI interface of a controller (CLI)....142
Verify WMI Settings....144

4 Configure Network Address Translation for Wireless Management IP.....147
Network address translation.....148
CAPWAP discovery.....149
Configure wireless management interface with a NAT public IP (CLI).....149
CAPWAP discovery IP response configuration.....150
Verify NAT settings.....152

5 Hardware Features.....155
BIOS Protection on the Controller.....156
BIOS or ROMMON Upgrade with BIOS Protection.....156
Upgrading BIOS.....156
USB disablement.....157
Disable USB (CLI).....157

6 Smart Licensing Using Policy.....159
Smart Licensing Using Policy.....160

7 Management over Wireless.....161
Management over wireless.....162
Restrictions on management over wireless.....162
Configure management of controller over wireless network (GUI).....162
Configure management of controller over wireless network (CLI).....162

8 Boot Integrity Visibility.....165
Boot integrity visibility.....166
Feature history for boot integrity visibility.....166
Verify software image and hardware.....166
Verify platform identity and software integrity.....167

9 SUDI99 Certificate Support.....171
SUDI certificates.....172
Software selection for SUDI trustpoint certificates.....172
SUDI99 certificate and software support.....173
Backward compatibility.....174
Restrictions of SUDI99 certificates.....174
Disable SUDI99 migration (GUI).....174
Disable SUDI99 migration (CLI).....175

10 Link Aggregation Group.....177
Link aggregation group.....178

Link aggregation control protocol....178
Port aggregation protocol....179
PAgP LAG configuration attributes....179
Port channel interface number....179
Flow control....180
LAG configuration in ON mode....180
Multi-LAG....181
Multichassis link aggregation group....181
Prerequisites for Multi-LAG....181
Restrictions for Multi-LAG....181
Supported topologies....181
Configure link aggregation group....183
Configure a port channel interface (GUI)....183
Create a port-channel interface (CLI)....184
Configure LAG in ON mode (CLI)....184
Add an interface to a port channel (LACP) (CLI)....185
Add an interface to a port channel (PAgP) (CLI)....186
Add a VLAN to a port channel (CLI)....187
Remove a port channel group from a physical interface (CLI)....187
Verify the LAG configuration....188

11 Reload Reason History....191
Reload reason history....192
Feature history....192
Request reload reason history using YANG....192
Reload reason history verification....194

Part II: System Upgrade....199

12 Upgrade the Cisco Catalyst 9800 Wireless Controller Software....201
Software upgrade for Cisco Catalyst 9800 Series Wireless Controller....202
Upgrade the controller software using GUI....203
Upgrade the controller software (CLI)....204
Convert from bundle-mode to install-mode....205
Copy a WebAuth tar bundle to the standby controller (GUI)....209

13 In-Service Software Upgrade....211
In-service software upgrade (ISSU)....212
Prerequisites for performing ISSU....212
Guidelines for ISSU....213
Upgrade software using ISSU (CLI)....214
Upgrade software using ISSU (GUI)....215
Upgrade software using ISSU with delayed commit (CLI)....216

Monitor ISSU....217
Troubleshoot ISSU....220

14 Staggered AP Upgrade....221
Access point upgrade configurations....222
Feature history for access point upgrade management enhancements....222
Access point upgrade configuration and behavior....223
Staggered and serial AP upgrade concepts....223
Controller CLI enhancements for AP upgrade management....224
Warning: Extended upgrade durations for serial and one percent batch modes....224
Configure staggered or serial AP upgrades....224
CLI and GUI options for upgrading APs....225
YANG Model Update....225

15 Software Maintenance Upgrade....227
Software maintenance upgrade....228
Install a SMU (GUI)....231
Install SMU (CLI)....232
Roll back an image (GUI)....232
Roll back SMU (CLI)....233
Deactivate SMU (CLI)....233
Configuration examples for SMU....234
AP device package....234
SMU package and AP image changes....235
Install AP device package (GUI)....236
Install AP device package (CLI)....236
Verify APDP on the Controller 237
Per site or per AP model service pack (APSP)....238
Restrictions and workflow....238
Rolling AP upgrade....239
Install AP service package (GUI)....240
Install AP service package (CLI)....241
Add a site to a filter (CLI)....242
Deactivate an image (CLI)....243
Roll back APSP (CLI)....243
Cancel the Upgrade (CLI)....244
Verify the upgrade....244
Verify AP upgrade on the controller....247

16 Efficient Image Upgrade and Out-of-band AP Image Download....249
Efficient image upgrades for FlexConnect APs....250
Restrictions....250
Efficient image upgrade download....250

Enable pre-download in a FlexConnect AP (GUI)....251
Configure an efficient image upgrade for FlexConnect APs....251
Out-of-band AP image download....258
Feature history for out-of-band AP image download....258
Restrictions for Out-of-Band AP image download....259
Download AP image from controller using HTTPS (CLI)....259
Download AP Image from Controller Using HTTPS (GUI)....260
Verify image upgrade....261

17 Predownloading an Image to an Access Point....265
Image pre-downloads to AP....266
Feature history for AP pre-download....266
Restrictions and best practices for pre-downloading an image to an AP....266
Procedures....267
AP image download time enhancement....273
Configure AP Image Download Time Enhancement (GUI)....274
Configure AP Image Download Time Enhancement (CLI)....275
Verify AP image download time enhancement configuration....276

18 N+1 Hitless Rolling AP Upgrade....279
N+1 hitless rolling AP upgrade and version skew....280
How N+1 hitless rolling AP upgrades work....281
Configure hitless upgrade....281
Verify hitless upgrade....283
Site-based rolling AP upgrade in N+1 network....284
Prerequisites for site-based rolling AP upgrade in N+1 networks....284
Restrictions for site-based rolling AP upgrade in N+1 networks....285
Best practices for N+1 redundancy deployments....285
Configure site-based N+1 upgrade (CLI)....285
Client steering enhancements....297
Client deauthentication....297
Best practice for preserving client connectivity during upgrade....297
Deauthenticate clients....297

19 NBAR Dynamic Protocol Pack Upgrade....299
Network-based application recognition protocol pack upgrade....300
Application visibility controls with DPI....300
NBAR limitations....301
Protocol pack upgrades and custom application configuration....301
Upgrade the NBAR2 protocol pack (CLI)....301
Verify the NBAR2 protocol pack....302

20 Wireless Sub-Package for Switch....303

Wireless sub-packages....304
Install wireless packages....304
Remove wireless package....305
Upgrading wireless packages....305
Boot in install mode....306
Install sub-package in a single step (GUI)....306
Install sub-package in a single step....307
Install multi-step sub-package....307
Install the package on a stack....308
Upgrade to new version of wireless package....309
Deactivate wireless package....309
Enable or disable auto-upgrade....310

Part III: Lightweight Access Points....311

21 Global Use Access Points....313
Global use access points....314
Benefits of global use AP....314
AP mode of options....314
Cloud IDs....314
Onboarding process for Wi-Fi 7 APs....315
Fast offline migration....316
Local status page....319
Offline migration....319
Prevent false or accidental migration....319
Cisco regulatory domain....320
Country code and regulatory domain....320
Best practices for Cisco regulatory domain....321
Examples of country code resolution....321
Regulatory activation file....323
E-Label display....326
E-LABEL compliance status....326
E-Label compliance status of AP models....327
Configure AP E-Label compliance status (GUI)....327

22 Countries and Regulations....329
Country codes....330
Prerequisites for configuring country codes....330
Configure country codes (GUI)....330
Configure country codes (CLI)....331
Configuration examples for configuring country codes....333
Regulatory compliance domain....334
Global country-level domains....334

Remove 8 channel restriction on slot 1 of Cisco Catalyst 9166 series APs....334
Configure XOR 5/6GHz radio slot for 6GHz band (CLI)....335
Ensure 6 GHz radio band persistence after CAPWAP reset....336

23 Security....337
Data Datagram Transport Layer Security....338
Configure data DTLS (GUI)....339
Configure data DTLS (CLI)....339
802.1X authentication....340
EAP-FAST protocol....341
EAP-TLS/EAP-PEAP Protocol....341
Access ports with dual port authentication....342
802.1X authentication limitations....342
Topology....343
Configuring 802.1X authentication type and LSC AP authentication type (GUI)....344
Configure 802.1X authentication type and LSC AP authentication type (CLI)....344
Enable 802.1X on the switch port (CLI)....347
Verify 802.1X on the switch port....349
Verify the authentication type....349
Verify access ports dual port authentication....349
Access point client ACL counters....355
Feature history for Access Point Client ACL Counter....356
AP MAC authorization....356
Feature history for AP MAC authorization....357
Create a AAA RADIUS server (GUI)....357
Configure a AAA RADIUS server (CLI)....358
Create a AAA Group configuration (GUI)....359
Configure a AAA Group configuration (CLI)....359
Create a AAA Group Server configuration (GUI)....361
Create a AAA Group Method List configuration (GUI)....361
Configure AP MAC authorization (CLI)....361
Verify MAC authorization details....362

24 AP Joining....365
Failover priority for APs....366
Guidelines to configure AP failover priority....366
Set AP priority (GUI)....366
Set AP priority (CLI)....366
AP Plug-n-Play....367
Provision AP from PnP server....367
Verify AP tag configuration....368
AP fallback to controllers using AP priming profile....368
AP controller selection and onboarding behavior....369
Restrictions for AP fallback to controllers using AP priming profile....369

Configure AP priming profile (CLI)....370
Configure AP priming using filters (CLI)....371
Configure Per-AP priming (CLI)....372
Verify priming filter configuration....372

25 AP Crash File Upload....375
AP crash files....376
Restrictions for AP crash file upload....376
Configure AP crash file upload process (CLI)....376

26 LED States for Access Points....379
LED states for APs....380
Configure the LED State on APs (GUI)....380
Configure LED state for APs (CLI)....381
Configure LED state in the AP profile....381
Verify LED state for APs....382

27 AP Support Bundle....383
AP support bundle....384
Export an AP support bundle (GUI)....384
Export an AP support bundle (CLI)....385
Monitor the status of support bundle export....386

28 Access Point Memory Information....387
Access point memory information....388
Verify AP memory information....388

29 Access Point Tag Persistency....389
AP tag persistency....390
Recommendation: Rejoin APs after enabling tag persistency....390
Requirement: Match tags across controllers....390
Tag persistency for redundancy scenarios....390
Configure access point tag persistency (GUI)....391
Configure AP tag persistency (CLI)....392
Verify AP tag persistency....392

30 AP Power Save....395
Access point power save....396
Tips for AP power save configuration....397
Requirements for AP power save configuration....397
Best practice for AP power save configuration....397
Warnings for AP power save configuration....397

AP power policies....397
Power-save modes....398
PoE profiles....398
Client wakeup threshold....404
Smart energy scenarios....405
How AP power save policies work....406
Configure power policy profile (GUI)....407
Configure a power policy profile (CLI)....407
Configure a calendar profile (GUI)....408
Configure a calendar profile (CLI)....409
Configure a power policy in an access point join profile (GUI)....410
Map a power profile under an AP profile (CLI)....411
Configure client wakeup threshold (CLI)....412
Configure PoE-Out Interface in Power Profile using GUI....412
Configure PoE-out interface in the power profile....413
Configuration example of power profile....414
Verify access point power policy (GUI)....414
Verify the access point power profile....415
Verify radio spatial streams....416
Verify client threshold....417
Verify PoE-out details....417

31 Real-Time Access Points Statistics....419

Access point real-time statistics....420
SNMP traps and intervals for APs....420
Restrictions for AP radio monitoring statistics....421
Configure access point real time statistics (GUI)....421
Configure access point real-time statistics (CLI)....422
Configure the access point radio monitoring statistics (CLI)....424
Monitor access point real-time statistics (GUI)....425
Verify access point real-time statistics....426

32 AP Configuration....429

AP console configurations....430
Configure the AP console (GUI)....430
Configure the AP console (CLI)....430
Verify the AP console status....431
AP audit configuration....432
Restrictions for AP audit configuration....433
Configure AP audit parameters (CLI)....433
Verify AP audit report summary....433
Verify AP audit report detail....434
Radio support for specified slot number....434
Configure 2.4-GHz radio support for the specified slot number (CLI)....434

Configure 5-GHz radio support for the specified slot number (CLI)....436
Configure 6-GHz radio support for the specified slot number....439
XOR radio support....441
Dual-band radios in Cisco AP models....441
Limitations....442
Configure default XOR radio support (CLI)....442
Configure XOR radio support for the specified slot number (GUI)....445
Configure XOR radio support for the specified slot number (CLI)....446
Verify XOR radio support....449
Receiver only dual-band radio support....449
Enable CleanAir with receiver Only dual-band radio on a Cisco AP (GUI)....449
Enabling CleanAir with receiver only dual-band radio on a Cisco AP (CLI)....449
Disable receiver only dual-band radio on a Cisco AP (GUI)....450
Disable receiver only dual-band radio on a Cisco AP (CLI)....450
Configure client steering (CLI)....451
Verify Cisco APs with dual-band radios....452
OFDMA support for 802.11ax APs....452
Supported modes on 802.11ax APs....453
Configure 802.11AX (GUI)....453
Configure channel width (CLI)....454
Configure 802.11ax radio parameters (GUI)....454
Configure 802.11ax radio parameters (CLI)....455
Set up the 802.11ax radio parameters (CLI)....456
Configure OFDMA on a WLAN (CLI)....456
UNII-3 channels in the -E domain....458
Configure UNII-3 channels for -E domain (CLI)....458
Restrictions and supported APs....459
Verify UNII-3 channel configurations....459
Verify channel width....460
Verify client details....461
Verify radio configuration....462
Cisco flexible antenna port....465
Configure a Cisco flexible antenna port (GUI)....466
Configure a Cisco flexible antenna port (CLI)....466
Verify flexible antenna port configuration....466
Environmental sensors in APs....467
Use cases....468
Configure environmental sensors in an AP profile (CLI)....468
Configure environment sensors (CLI)....469
Verify the AP sensor status....470
Accelerometer sensor support....471
CAPWAP LAG support....474
Supported models....475
Restrictions for CAPWAP LAG support....475
Enable CAPWAP LAG support on controller (GUI)....475

Enable CAPWAP LAG support on controller (CLI)....476
Enable and disable CAPWAP LAG globally on controller....476
Enable CAPWAP LAG for an AP profile (GUI)....477
Enable CAPWAP LAG for an AP profile (CLI)....477
Disable CAPWAP LAG for an AP profile (CLI)....478
Disable CAPWAP LAG support on controller....479
Verify CAPWAP LAG support configurations....479
Configure bulk AP provisioning (GUI)....480
Activate tracking for deactivated infrastructure devices (GUI)....480
CAPWAP message aggregation....481
Control and provisioning of wireless APs message aggregation....481
Guidelines of CAPWAP message aggregation....482
Use cases of CAPWAP message aggregation....482
Configure CAPWAP message aggregation (CLI)....482
Verify CAPWAP message aggregation....483

33 Secure Data Wipe....485
Secure data wipe....486
Feature history....486
Restrictions....486
Supported AP models and software versions....486
Verify data wipe....487

34 Wi-Fi 7 Operations....489
Wi-Fi 7 AP....490
Multi-link operation (MLO)....491
Multi-link statistics....492
Configure 802.11be network parameters (CLI)....492
Configure 802.11be network parameters (GUI)....493
Configure A-MPDU window size in RF radio profile (GUI)....494
Configure A-MPDU window size in RF radio profile (CLI)....494
Enable or disable client serving on AP (GUI)....495
Configure 802.11be MCS rates (CLI)....495
Configure 802.11be radio parameter frequencies (GUI)....496
Configure multi BSSID 802.11be parameters (GUI)....496
Configure multi BSSID 802.11be parameters (CLI)....497
Configure 802.11be parameters (GUI)....498
Monitor 5 GHz radio WLAN slot data (GUI)....498
Monitor 802.11be radio parameters (GUI)....498
Verify multi-link statistics....499
Verify 802.11be parameters....503
Verify 320 MHz details (GUI)....504
Verify multi BSSID details....504
Per WLAN Wi-Fi 7 toggle....505

Feature history for per WLAN Wi-Fi 7 toggle....505
Per-WLAN 802.11be configuration....506
Supported security configurations for 802.11be....507
Beacon-Protection status....508
MBSSID....508
Tx-VAP WLAN of MBSSID set for 6 GHz band....509
Create a new 802.11be profile (GUI)....509
Map the 802.11be profile to a WLAN (GUI)....510
Create and map 802.11be profile (CLI)....510
Verify 802.11be profile details....512
Verify 802.11be profile parameters with MLO mapping....512
Verify 802.11be profile mapping to WLAN....512
Verify VAP status for 802.11ax and 802.11be WLAN on 6 GHz band....513
Verify WLAN downgrade from 802.11be to 802.11ax mode....515
Verify AP WLAN information on 6 GHz band with MBSSID slot....519

35 Troubleshooting Lightweight Access Points....521
Support resources for technical issue resolution....522
Support articles....522
Provide feedback on support documents....522
Disclaimer and caution....523

Part IV: Access Points Modes....525

36 Sniffer Mode....527
Sniffer....528
XOR radio roles....528
Essential hardware and software for sniffer setup....529
Restrictions on sniffer....529
Configure an access point as sniffer (GUI)....529
Configure an access point as sniffer (CLI)....530
Enable or disable sniffing on the AP (GUI)....531
Enable or disable sniffing on the AP (CLI)....532
Configure XOR radio role sniffer support on the access point (CLI)....533
Verify sniffer configurations....534
Verify XOR radio role sniffer configuration....534
Examples for sniffer configurations and monitoring....535

37 Monitor Mode....537
Monitor mode....538
Enable monitor mode (GUI)....538
Enable monitor mode (CLI)....539

38 Management Mode Migration in Cisco Catalyst 916X Access Points.....541
Management mode migration for Cisco Catalyst Wireless 916X Series AP.....542
Regulatory domain.....543
Configure management mode migration (GUI).....547
Export APs migrated to Meraki management mode (GUI).....548
Configure the access point management mode (CLI).....549
Verify the management mode migration details.....550

39 Configure the controller for FlexConnect.....551
FlexConnect Authentication.....552
How FlexConnect authentication works.....553
Controller discovery methods.....554
FlexConnect authentication and switching modes.....555
Local authentication in a FlexConnect AP.....556
Local and backup RADIUS server configuration.....556
Restrictions.....557
Guidelines and restrictions for FlexConnect.....558
Configure a site tag (CLI).....559
Configure a policy tag (CLI).....560
Attach policy and site tags to an access point (GUI).....562
Attach policy tag and site tag to an AP (CLI).....563
Link an ACL policy to the defined ACL (GUI).....564
Apply access control lists on FlexConnect.....565
Configure FlexConnect.....566
Configure the switch at a remote site.....566
Configure the controller for FlexConnect.....567
Configure an access point for FlexConnect.....570
Configure an access point for local authentication on a WLAN (GUI).....570
Configure an access point for local authentication on a WLAN (CLI).....570
Connect client devices to WLANs.....571
FlexConnect ethernet fallback.....571
Configure FlexConnect AP local authentication (GUI).....572
Configure FlexConnect access point local authentication.....573
Configure FlexConnect access point local authentication with external RADIUS server.....576
Configuration example: FlexConnect with central and local authentication.....579
NAT-PAT functionalities in FlexConnect.....580
Create a WLAN.....580
Configure a wireless profile policy and NAT-PAT (GUI).....581
Configure a wireless profile policy and NAT-PAT (CLI).....581
Map a WLAN to a policy profile (CLI).....583
Configure a site tag.....583
Attach policy and site tags to an access point (GUI).....584
Attach a policy tag and a site tag to an access point (CLI).....584

Split tunneling for FlexConnect....585
Restriction: split tunneling for FlexConnect....586
Configure Split Tunneling for a WLAN or Remote LAN....586
VLAN-based central switching for FlexConnect....594
Configure VLAN-based central switching (GUI)....594
Configure VLAN-based central switching (CLI)....595
OfficeExtend AP for FlexConnect....596
Configure OfficeExtend AP....596
Disable OfficeExtend AP....597
Best practices for OfficeExtend AP for FlexConnect....598
OEAP personal SSID support....598
Clearing personal SSID from an OfficeExtend access point....601
Example: viewing OfficeExtend configuration....601
Proxy address resolution protocol....602
Enable proxy ARP for FlexConnect access points (GUI)....602
Enable proxy ARP for FlexConnect access points (CLI)....602
Overlapping client IP address in flex deployment....604
Enable overlapping client IP address in flex deployment (GUI)....604
Enable overlapping client IP address in flex deployment....604
Verify overlapping client IP address in flex deployment (GUI)....605
Verify overlapping client IP address in flex deployment....605
FlexConnect high scale mode....607
Enable PMK propagation (CLI)....607
Flex resilient with Flex and bridge modes....608
Configure a flex profile (GUI)....609
Configure a Flex profile (CLI)....609
Configure a site tag (CLI)....611
Configure a mesh profile (CLI)....611
Associate the wireless mesh to an AP profile (CLI)....612
Attach site tag to an access point (CLI)....613
Configure switch interface for APs (CLI)....614
Flex resilient with flex and bridge mode AP configuration verification....615
SuiteB ciphers in FlexConnect mode....616
Configure SuiteB ciphers (GUI)....617
Configure Suite-B ciphers (CLI)....618
Configure GCMP-128, GCMP-256, or CCMP-256 (CLI)....619
Verify SuiteB cipher status....620

40 OEAP Link Test....621
OfficeExtend Access Point link test....622
Configure OEAP link test (CLI)....622
Perform OEAP link test (GUI)....623
Verify OEAP link test....623

41 Cisco OEAP Split Tunneling....625
Cisco OEAP split tunneling....626
Cisco OEAP split tunneling traffic management....626
Prerequisites for Cisco OEAP split tunneling....627
Restrictions for Cisco OEAP split tunneling....627
Use cases for Cisco OEAP split tunneling....628
How Cisco OEAP split tunneling works....628
Create an IP address ACL (CLI)....628
Create a URL ACL (CLI)....629
Add an ACL to a FlexConnect profile (GUI)....630
Enable split tunneling in a policy profile....632
Verify the Cisco OEAP split tunnel configuration....633

42 Access Points Survey Mode....635
AP survey modes....636
Restrictions for access points survey mode....636

43 AP Deployment Mode....637
AP deployment mode....638
Restrictions for AP deployment mode....638
Configure an AP deployment mode (GUI)....638
View deployment status....639
Configure AP deployment mode (CLI)....639
Verify AP deployment mode....640

Part V: Smart Spaces....643

44 Access Point Location....645
AP auto location support....646
Advanced functionalities in Cisco IOS XE 17.12.1....647
Ultra-wideband ranging....647
Global Navigation Satellite System raw data streaming through gRPC....648
Configure AP geolocation derivation using ranging (GUI)....649
Configure AP geolocation derivation using ranging (CLI)....649
Configure the AP Ranging Parameters through the GUI....650
Configure AP ranging parameters (CLI)....650
Configure AP location (CLI)....651
Configure on-demand AP ranging (CLI)....652
Enable fine time measurement (802.11mc) responder (GUI)....653
Configure fine time measurement (802.11mc) responder....654
Configure air pressure reporting (CLI)....655

Configure AP ultra wide band parameters (GUI)....655
Configure access point ultra wide band parameters (CLI)....656
Verify AP geolocation information....657

45 SS IOT: Implement Wi-Fi Coexistence with ESL....661
Wi-Fi-IoT radio coexistence....662
Feature history for Wi-Fi-IoT radio coexistence....662
Best practices for Wi-Fi-IoT coexistence....663
How Wi-Fi-IoT coexistence works with IOx applications....663
Configuration priority for Wi-Fi-IoT coexistence....664
Create an RF profile for Wi-Fi-IoT coexistence (CLI)....664
Configure Wi-Fi-IoT coexistence (CLI)....665
Tag the RF profile to an AP (CLI)....666
Verify Admin state on RF profile....667
Verify coexistence configuration applied to an AP on controller....669
Verify coexistence summary of all the APs....672
Verify coexistence Wi-Fi statistics....673

46 Smart Spaces - AnyLocate capabilities....675
AnyLocate feature set....676
Feature history for Smart Spaces - AnyLocate capabilities....677
Prerequisites for asset tracking....677
Time difference of arrival (TDOA)....678
Automatic reclustering for clock synchronization....679
Prerequisites for wayfinding configurations....679
Impact of AP downtime on location services....679
Enable UWB wayfinding (GUI)....680
Enable UWB wayfinding (CLI)....680
Configure height above the floor (GUI)....681
Configure height above the floor (CLI)....681
Verify AP UWB wayfinding status....682
Verify AP UWB asset-tracking status....683
Verify AP location information....684
Verify the prerequisites for asset tracking....684

Part VI: Radio Resource Management....687

47 Radio Resource Management....689
Radio resource management....690
Radio resource monitors....690
Radio frequency groups....691
Transmit power control....696
Dynamic channel assignment....696

Coverage hole detection and correction....699
Cisco AI enhanced RRM....699
Restrictions....701
How to configure RRM....701
Configure neighbor discovery type (GUI)....701
Configure neighbor discovery type (CLI)....702
Configure RF groups....703
Configure transmit power control....708
Configure 802.11 RRM parameters....711
Configure advanced 802.11 RRM....721
Monitor RRM parameters and RF group status....724
Monitor RRM parameters....724
Verify RF group status....725
Event-Driven Radio Resource Management 726
Configure ED-RRM on the controller (CLI)....726
Rogue detection - rogue channel width....728
Configure rogue channel width (CLI)....728
Configure rogue classification rules (GUI)....730
Verify rogue channel width....733

48 Coverage Hole Detection....735
Coverage hole detection and correction....736
Configure coverage hole detection (GUI)....736
Configure coverage hole detection (CLI)....736
Configure CHD for RF tag profile (GUI)....739
Configuring CHD for RF profile (CLI)....739

49 Optimized Roaming....741
Optimized roaming....742
Restrictions for optimized roaming....742
Configure optimized roaming (GUI)....742
Configure optimized roaming (CLI)....743

50 Cisco Flexible Radio Assignment....745
Flexible radio assignment actions....746
Configure FRA action in default RF profile (CLI)....746
Configure FRA action in 2.4-GHz RF profile (CLI)....747
Verify FRA action configuration....748

51 XOR Radio Support....749
Dual-band radios in Cisco AP models....750
Limitations....751
Configure default XOR radio support (CLI)....751

Configure XOR radio support for the specified slot number (GUI)....754
Configure XOR radio support for the specified slot number (CLI)....754
Verify XOR radio support....757

52 Cisco Receiver Start of Packet....759
Receiver start of packet detection threshold....760
Restrictions for Rx SOP....760
Permitted range for the Rx SOP threshold....760
Known behavior....760
Configure Rx SOP (CLI)....760
Customize RF profile (CLI)....761

53 Client Limit....763
Client limits....764
Limitations....764
Configure client limit per WLAN (GUI)....764
Configure client limit per WLAN (CLI)....765
Configure client limit per AP (GUI)....766
Configure client limit per AP (CLI)....767
Configure client limit per radio (GUI)....767
Configure client limit per radio (CLI)....768
Verify client limit....769

54 IP Theft....771
IP theft....772
Lower the idle timeout to prevent false IP theft....772
IP preference levels for IP theft detection....772
Configure IP theft (GUI)....772
Configure IP theft (CLI)....773
Configure the IP theft exclusion timer (CLI)....773
Add static entries for wired hosts (CLI)....774
Verify IP theft configuration....774
SAE client exclusion visibility....776
Configure client exclusion timeout (GUI)....777
Configure client exclusion timeout (CLI)....778
Verify client exclusion data....779

55 Unscheduled Automatic Power Save Delivery....781
Unscheduled automatic power save delivery overview....782
View unscheduled automatic power save delivery....782

56 Target Wake Time....783

Target wake time....784
Optimize power-saving with target wake time....785
Configure target wake time at radio level (CLI)....785
Enable target wakeup time on WLAN (CLI)....787
Disable target wakeup time on WLAN (CLI)....788
Configure target wake time (GUI)....789
Verify target wakeup time....789

57 Enabling USB Port on Access Points....791
USB port as power source for access points....792
Configure an AP profile (CLI)....792
Configure USB settings for an access point (CLI)....794
Configure USB settings for an access point (GUI)....795
USB configuration monitoring for access points (CLI)....796

58 Dynamic Frequency Selection....797
Dynamic frequency selection....798
Feature history for channel availability check (CAC)....798
Channel availability check....798
DFS verification....799
Zero wait dynamic frequency selection....800
Configure zero wait dynamic frequency selection globally....800
Configure zero wait dynamic frequency selection globally (CLI)....800
Enable zero wait dynamic frequency selection on a RF profile (GUI)....801
Enable zero wait dynamic frequency selection on a RF profile (CLI)....802
Zero wait dynamic frequency selection configuration verification....802

59 Cisco Access Points with Tri-Radio....805
Cisco access points with tri-radio....806
Guidelines and restrictions for tri-radio access points....807
Configure tri-radio for AP (GUI)....808
Configure the tri-radio (CLI)....808
Configure 5-GHz dual radio mode for AP (GUI)....808
Configure the dual radio mode and enable slots (CLI)....809
Set radio roles for slots (CLI)....810
Configure the tri-radio dual radio role (CLI)....811
Tri-radio configuration verification on the controller....811

60 Cisco Catalyst Center Assurance Wi-Fi 6 Dashboard....813
Cisco Catalyst Center assurance Wi-Fi 6 dashboard....814
Configure Cisco Catalyst Center assurance Wi-Fi 6 dashboard parameters (CLI)....815
AP DFS counter verification (CLI)....817
Wi-Fi 6 access point parameter verification....817

61 Antenna Disconnection Detection....819
Antenna disconnection detection....820
Limitations....820
How antenna disconnection detection works....821
Configure antenna disconnection detection....821
Configure antenna disconnection detection using GUI....823
Detect broken antenna using SNMP trap (CLI)....823
Detect broken antenna using SNMP trap (GUI)....824
Verify antenna disconnection detection (Reference)....824
Verify antenna disconnection detection (GUI)....825

62 Neighbor Discovery Protocol Mode on Access Points....827
NDP mode....828
Configuration Tasks....829
Configure RRM NDP mode (GUI)....829
Configure the NDP mode (CLI)....830
Configure the NDP Type (CLI)....830
Configure NDP mode in the RF profile (GUI)....831
Configure NDP mode in the RF profile (CLI)....831
Monitor radio statistics and NDP using GUI....832
Verify NDP mode....833

63 6-GHz Band Operations....837
6-GHz Band Support....838
Feature history....838
Configure preferred scanning channels in the RF profile (GUI)....838
Configure preferred scanning channels in RF profile (CLI)....838
Configure broadcast probe response in RF profile (GUI)....839
Configure broadcast probe response in RF profile (CLI)....840
Configure FILS discovery frames in RF profile (GUI)....840
Configure FILS discovery frames in RF profile (CLI)....841
Configure multi BSSID profile (GUI)....842
Configure multi BSSID profile (CLI)....842
Configure multi BSSID in RF profile (GUI)....843
Configure multi BSSID in RF profile (CLI)....843
Configure dynamic channel assignment freeze (CLI)....844
6-GHz client steering....844
Configure 6-GHz client steering in global configuration mode (GUI)....845
Configure 6-GHz client steering in global configuration mode (CLI)....845
Configure 6-GHz client steering on WLAN (GUI)....846
Configure 6-GHz client steering on WLAN (CLI)....847
Verify 6-GHz client steering....847

Part VII: Network Management.....849

64 AP Packet Capture.....851

AP client packet capture.....852
Limitations of AP client packet capture.....852
Enable packet capture (GUI).....852
Enable packet capture (CLI).....853
Create AP packet capture profile and map to an AP join profile (GUI).....853
Create AP packet capture profile and map to an AP join profile.....854
Start or stop packet capture.....855

65 DHCP Option82.....857

DHCP Option 82.....858
Configure DHCP Option 82 format.....859
Configure DHCP Option 82 global interface.....861
Configure DHCP Option 82 globally through server override (CLI).....861
Configure DHCP Option 82 through server override (CLI).....861
Configure DHCP Option 82 globally through different SVIs (GUI).....862
Configure DHCP Option 82 globally through different SVIs (CLI).....863
Configure DHCP Option 82 through a VLAN interface.....863
Configure DHCP Option 82 through option-insert command (CLI).....863
Configure DHCP Option 82 through server-ID-override command (CLI).....865
Configure DHCP Option 82 through a subscriber-ID (CLI).....866
Configure DHCP Option 82 through server-ID-override and subscriber-ID commands (CLI).....867
Configure DHCP Option 82 through different SVIs (CLI).....869
AP DHCP Option 82 support on FlexConnect local switching mode.....870
Configure AP DHCP Option82 support.....870
Verify AP DHCP Option82 support.....871

66 RADIUS Authentication and Accounting.....873

RADIUS realms.....874
Enable RADIUS realm.....875
Configure realm to match the RADIUS server for authentication and accounting.....875
Configure AAA policy for WLANs.....876
Verify RADIUS-realm configuration.....879
RADIUS accounting of AP events.....880
Configure accounting method-list for an AP profile.....881
Verify AP accounting information.....882
Configure AAA accounting using default method list (CLI).....883
Configure HTTP command accounting using named method list (CLI).....883
RADIUS call station identifier.....884
Configure a RADIUS call station identifier.....886

RADIUS VSA....886
Create an attribute list....888
Map a AAA policy to WLAN policy profile....889
Map the WLAN policy profile to a WLAN....889

67 Cisco StadiumVision....891
Cisco StadiumVision....892
Configure Cisco StadiumVision parameters (GUI)....892
Configure Cisco StadiumVision parameters (CLI)....892
Verify StadiumVision configurations....893

68 Persistent SSID Broadcast....897
Persistent SSID broadcast....898
Configure persistent SSID broadcast (CLI)....898
Verify persistent SSID broadcast....899

69 Network Monitoring....901
Network monitoring 902
Status information received synchronously - configuration examples....902
Alarm and Event Information Received Asynchronously - Configuration Examples....904

70 Creating a Lobby Ambassador Account....907
Lobby ambassador account....908
Create a lobby ambassador user account....908
Create a lobby ambassador account (CLI)....910

71 Lobby Ambassador Account....911
Lobby ambassador account....912
Create a lobby ambassador user account....912
Create a lobby ambassador account (CLI)....914
Configure WLAN (GUI)....915
Client allowed list....916
Restrictions for client allowed list....916
Creating a client allowed list (GUI)....916
Manage guest users (GUI)....917
View a client allowed list....917

72 Guest User Accounts....919
Create guest user accounts....920
Create a guest user account (GUI)....920
Create a guest user account (CLI)....920
Verify guest user account....922

Assign username to guest users in a WLAN (CLI)....923

73 Link Local Bridging....925

Link local bridging....926

Guidelines and restrictions for link local bridging....927

Enable link local bridging per policy profile (GUI)....927

Enable link local bridging per policy profile (CLI)....927

Verify link local bridging....928

74 Web Admin Settings....931

Web admin settings....932

Configure HTTP/HTTPS access (GUI)....932

Configure HTTP trust point (GUI)....933

Configure NETCONF YANG (GUI)....934

Configure timeout policy (GUI)....934

Configure VTY (GUI)....935

Post-quantum cryptography support....935

75 Web UI Configuration Command Accounting in TACACS Server....939

Web UI configuration command accounting in TACACS+ server....940

Feature history for Web UI configuration command accounting in TACACS+ server....940

Guidelines for Web UI configuration command accounting in TACACS+ server....940

Configure AAA accounting using default method list (CLI)....940

Configure HTTP command accounting using named method list (CLI)....941

76 Embedded Packet Capture....943

Embedded packet capture....944

Feature history for embedded packet capture....944

Configure embedded packet capture (CLI)....944

Verify embedded packet capture....947

77 Layer 3 Access....951

Layer 3 access....952

Restrictions for Layer 3 access....952

Use cases for Layer 3 access....953

Enable Layer 3 access on policy profile (GUI)....954

Enable Layer 3 access on policy profile (CLI)....954

Configure a client gateway (GUI)....955

Configure a client gateway (CLI)....955

Enable internal DHCP with VRF (CLI)....956

Verify Layer 3 access details....958

OSPF protocols....960

Configure OSPF....960
Verify routing protocol details....964
PIM sparse mode....972
Enable multicast traffic without VRF (GUI)....973
Enable multicast traffic without VRF (CLI)....973
Enable multicast traffic with PIM-SSM (CLI)....975
Verify multicast traffic details....976
Network address translation (NAT)....979
Selective NAT support....980
Verify static NAT details....987
Verify dynamic NAT details....988
Verify NAT details....990
Verify NAT timeout details....991

78 IPv6 Layer 3 Access....993
IPv6 Layer 3 access....994
Feature history for IPv6 Layer 3 Access....994
DHCPv6 relay agents....994
How IPv6 Layer 3 client forwarding works....995
How DHCPv6 relay agents work....996
Guidelines for IPv6 Layer 3 forwarding....996
Restrictions for IPv6 Layer 3 forwarding....997
Use cases for IPv6 Layer 3 forwarding....997
Enable Layer 3 access on a wireless policy profile....997
Configure DHCPv6 relay or LDRA....998
Configure OSPFv3 for IPv6 Layer 3 access....1000
Configure IPv6 multicast routing for wireless clients....1000

Part VIII: System Management....1003

79 Network Mobility Services Protocol....1005
Network mobility services protocol....1006
Radioactive tracing for NMSP....1006
Enable NMSP on premises services (CLI)....1006
Modify the NMSP notification interval for clients, RFID tags, and rogues (CLI)....1007
Modify the NMSP notification threshold for clients, RFID tags, and rogues (CLI)....1008
Configure NMSP strong cipher (CLI)....1009
Verify NMSP settings....1009
Examples: NMSP settings configuration....1012
NMSP by AP groups with subscription list from CMX....1012
Verify NMSP by AP groups with subscription list from CMX....1013
Probe RSSI location....1014
Configure probe RSSI (CLI)....1015

RFID tag support....1016
Configure RFID tag support (CLI)....1016
Verify RFID tag support....1017

80 Application Visibility and Control....1021
Application visibility and control....1022
Guidelines for application visibility and control....1022
Prerequisites for application visibility and control....1023
Restrictions for application visibility and control....1023
Configure AVC....1024
Create a flow monitor (CLI)....1024
Configure a flow monitor (GUI)....1026
Create a flow record (CLI)....1026
Create a flow exporter (CLI)....1028
Configure a policy tag (CLI)....1030
Attach a policy profile to a WLAN interface (GUI)....1031
Attach a policy profile to a WLAN interface (CLI)....1031
Attach a policy profile to an AP (CLI)....1032
Verify the AVC configuration (CLI)....1033
Default DSCP on AVC....1034
Configuring default DSCP for AVC profile (GUI)....1034
Guidelines to configure default DSCP marking for unclassified AVC packets....1034
Create class map (CLI)....1035
Create policy map (CLI)....1035
AVC-based selective reanchoring....1037
Restrictions for AVC-based selective reanchoring....1037
Configure the flow exporter (CLI)....1037
Configure the flow monitor (CLI)....1038
Configure the AVC reanchoring profile (CLI)....1039
Configure the wireless WLAN profile policy (CLI)....1040
Verify AVC reanchoring....1041

81 Software-Defined Application Visibility and Control....1047
Software-defined application visibility and control....1048
Supported platforms for SD-AVC IPv6....1049
Enable software-defined application visibility and control on a WLAN (CLI)....1050
Configure software-defined application visibility and control global parameters (CLI)....1051
List of package files....1052
Verify SD-AVC controller connection status....1052
Verify SD-AVC details in AP....1053

82 Cisco Hyperlocation....1055
Cisco Hyperlocation....1056

Restrictions on Cisco Hyperlocation....1057
Configure Cisco Hyperlocation (GUI)....1057
Configure Cisco Hyperlocation (CLI)....1057
Verify Cisco Hyperlocation....1059
NTPS and BLE configuration....1062
Best practices for NTP configuration....1063
AP group NTP server....1064
Configure an AP group NTP server (CLI)....1065
Configure AP timezone (CLI)....1066
BLE concurrent scanning and beaconing....1066
Verify BLE concurrent scanning and beaconing....1067

83 FastLocate for Cisco Catalyst Series Access Points....1071
FastLocate....1072
Restriction for fabric deployments....1072
Supported access points....1072
FastLocate network components and functions....1073
Configure FastLocate (GUI)....1073
Verify FastLocate....1074

84 IoT Services Management....1077
IoT services management....1078
gRPC connection configuration details....1078
Enable the 802.15 radio (CLI)....1079
Configure the gRPC token (CLI)....1080
Enable gRPC in an AP profile (CLI)....1080
Verify BLE state and mode....1081
Verify BLE details....1081
Verify gRPC summary, status, and statistics....1082

85 IoT Module Management in the Controller....1085
IoT module management in the controller....1086
Enable a USB on the controller....1086
Verify the USB modules....1087

86 EDCA Parameters....1089
EDCA parameters....1090
Configure EDCA parameters (GUI)....1090
Configure EDCA parameters (CLI)....1090

87 Adaptive Client Load-Based EDCA....1093
Adaptive client load-based EDCA....1094

Use case for adaptive client load-based EDCA....1094
Restrictions for adaptive client load-based EDCA....1094
Configure adaptive client load-based EDCA (GUI)....1094
Configure adaptive client load-based EDCA (CLI)....1095
Verify adaptive client load-based EDCA configuration....1096

88 802.11 parameters and Band Selection....1099
Band select....1100
802.11 bands....1100
Restrictions for band selection, 802.11 bands, and parameters....1101
Configure band selection (GUI)....1101
Configure band selection (CLI)....1102
Configure the 802.11 bands (GUI)....1103
Configure the 802.11 bands (CLI)....1104
Configure a band-select RF profile (GUI)....1106
Configure a band-select RF profile (CLI)....1107
Configure 802.11n parameters (GUI)....1108
Configure 802.11n parameters (CLI)....1108
Configure 802.11h parameters (CLI)....1110
Verify configuration settings using band selection and 802.11 bands commands....1112
Example: view the configuration settings for the 6-GHz band....1112
Example: view the configuration settings for the 5-GHz band....1114
Example: view the configuration settings for the 2.4-GHz band....1115
Example: view the status of 802.11h parameters....1117
Example: verify the band-selection settings....1117
Examples: band selection configuration....1119
Examples: 802.11 bands configuration....1119
Examples: 802.11n configuration....1120
Examples: 802.11h configuration....1120

89 NBAR Protocol Discovery....1123
NBAR protocol discovery....1124
Restrictions of NBAR....1124
Configure NBAR protocol discovery (CLI)....1124
Verify protocol discovery statistics....1125

90 Conditional Debug, Radioactive Tracing, and Packet Tracing....1127
Conditional debugging....1128
Radioactive tracing....1128
Feature history for conditional debug and radioactive tracing....1129
Conditional debugging and radioactive tracing....1129
Location of tracefiles....1129
Configure conditional debugging (GUI)....1130

Configure conditional debugging (CLI)....1130
Radioactive Tracing for L2 Multicast....1132
Recommended workflow for trace files....1132
Copy tracefiles off the box....1133
Configuration examples for conditional debugging....1134
Verify conditional debugging....1134
Example: verify radioactive tracing log for SISF....1134
Packet tracing....1135
Configure conditional debugging packet tracing (CLI)....1136
Configure conditional debugging packet tracing per AP (CLI)....1137
Configure conditional debugging packet tracing per client (GUI)....1138
Configure conditional debugging packet tracing per client (CLI)....1138
Verify conditional debugging packet tracing configuration....1139
Wireless client debug bundle....1139
Types of logs collected....1140
Collect wireless client debug bundle (CLI)....1141

91 Aggressive Client Load Balancing....1143
Aggressive client load balancing....1144
Guidelines of aggressive client load balancing....1144
Restrictions of aggressive client load balancing....1144
Enable aggressive client load balancing (GUI)....1144
Configure aggressive client load balancing (GUI)....1145
Configure aggressive client load balancing (CLI)....1145

92 RF based Automatic AP Load Balancing....1147
RF based automatic AP load balancing....1148
Restrictions and guidelines for RF based automatic AP load balancing....1149
Configure RF based automatic AP load balancing....1149
Disable RF based automatic AP load balancing (CLI)....1151
Verify automatic WNCd load balancing....1152

93 Accounting Identity List....1155
Configure accounting identity list (GUI)....1156
Configure accounting identity list (CLI)....1156
Configure client accounting (GUI)....1156
Configure client accounting (CLI)....1157

94 Accounting Session ID....1159
Accounting session ID....1160
Configure an accounting session ID (CLI)....1160
Verify an account session ID....1162

95 Interim Accounting....1163
Interim accounting....1164
Disable interim accounting (CLI)....1164
Verify interim accounting....1165

96 Wireless Multicast....1167
Wireless multicast....1168
Multicast optimization....1169
IPv6 global policies....1169
Prerequisites for configuring wireless multicast....1169
Restrictions on configuring wireless multicast....1170
Configuring wireless multicast....1170
IPv6 snooping....1172
IPv6 device tracking....1173
IPv6 address glean....1174
Restrictions for IPv6 snooping....1174
IPv6 multicast over multicast....1175
Configure IPv6 multicast-over-multicast (GUI)....1176
Configure IPv6 multicast-over-multicast (CLI)....1176
Verify IPv6 multicast-over-multicast....1176
Verify the multicast connection between the controller and the AP....1177
Directed multicast service....1177
Configure directed multicast service (GUI)....1177
Configure directed multicast service (CLI)....1178
Verify the directed multicast service configuration....1179
Restrictions for wireless broadcast, non-IP multicast and multicast VLAN....1180
Configure non-IP wireless multicast (CLI)....1180
Configure wireless broadcast (GUI)....1181
Configure wireless broadcast (CLI)....1181
Configure multicast-over-multicast for AP multicast groups (CLI)....1182
Verify wireless multicast....1183
Multicast optimization....1183
Configure IP multicast VLAN for WLAN (GUI)....1184
Configure IP multicast VLAN for WLAN (CLI)....1184
Verify the multicast VLAN configuration....1185
Multicast filtering....1186
Supported L3 multicast report for filtering....1187
Configure multicast filtering (CLI)....1187
Verify multicast filtering....1188

97 Map-Server Per-Site Support....1189
Map server per site support....1190

Configure the default map server (GUI)....1190
Configure the default map server (CLI)....1191
Configure a map server per site (GUI)....1191
Configure a map server per site (CLI)....1192
Create a map server for each VNID (GUI)....1192
Create a map server for each VNID (CLI)....1193
Create a fabric profile and associate a tag and VNID (GUI)....1193
Create a fabric profile and associate a tag and VNID (CLI)....1194
Verify the map server configuration....1194

98 Volume Metering....1197
Volume metering....1198
Configure volume metering (CLI)....1198

99 Enabling Syslog Messages in Access Points and Controller for Syslog Server....1201
Syslog messages in access points and controllers....1202
Syslog support for client state change....1202
Configure syslog support for client state change (CLI)....1203
Sample syslogs....1203
Configure the syslog server for the controller (GUI)....1204
Configure a syslog server for a controller (CLI)....1205
AP logging to the in-memory buffer and flash....1207
AP logging to terminal....1207
Configure AP logging to a syslog server....1208
Configure a syslog server for an AP profile (CLI)....1209
Configure the AP syslog settings (GUI)....1210
Verify syslog server configurations....1211

100 Login Banner....1215
Login banners....1216
Configure a login banner (GUI)....1216
Configure a login banner (CLI)....1216

101 Wi-Fi Alliance Agile Multiband 1219
Wi-Fi alliance agile multiband....1220
Components of a Wi-Fi Agile Multiband topology....1220
Supported MBO components....1221
Limitations of MBO....1222
Configure MBO on a WLAN (CLI)....1222
Verify MBO configuration....1224

102 SNMP Traps....1227
SNMP trap configuration....1228
Configure SNMP Traps (GUI)....1228
Enable AP Traps (CLI)....1229
Enable wireless client traps (CLI)....1230
Enable Mesh Traps (CLI)....1230
Enable RF Traps (CLI)....1231
Enable rogue, mobility, RRM, and general traps (CLI)....1232
Verify SNMP wireless traps....1233

103 Disabling Clients with Random MAC Address....1235
Random MAC addresses....1236
Configure random MAC address denial (CLI)....1236
Verify denial of clients with a random MAC address....1237

104 Dataplane Packet Logging....1239
Dataplane packet logs....1240
Enable or disable debug level (CLI)....1240
Enable packet logging in the global and filtered buffers in the ingress-path (CLI)....1241
Enable packet logging in global and filtered buffers for the punt-inject path (CLI)....1242
Verify dataplane packet logging....1243
Clear logs and conditions in global and filtered trace buffers....1244

105 Streaming Telemetry....1245
Streaming telemetry....1246
Gather points....1246
Subscription....1247
Transport....1248
Scale considerations....1248
Session....1248
Configure telemetry on controllers....1250
On-Change Telemetry Support....1259
Supported XPaths for on-change subscription....1259
Troubleshooting Telemetry Support....1263
Client event and SSID telemetry filters....1266
How the Client Event and SSID Telemetry Filter Works....1266
Restrictions for Cisco Catalyst Center client event and SSID telemetry filter....1268
Configure Client Event and SSID Telemetry Filters....1268
Enable iCAP filter on AP (CLI)....1268
Disable client telemetry data for a WLAN (YANG)....1269
Verify client telemetry data for WLAN....1269

106 Application Performance Monitoring....1271
APM....1272
Restrictions....1272
Configure APM....1273
Create a flow monitor....1273
Create a wireless WLAN profile policy....1274
Create a policy tag....1276
Attach the policy profile to an AP....1276
Verify APM....1277

107 Wireless Clients Threshold Warning....1279
Wireless client threshold warnings....1280
Configure a warning period (CLI)....1280
Configure the client threshold (CLI)....1281

108 Intelligent Capture Hardening....1283
Cisco Intelligent Capture (iCAP) feature and enhancements....1284
Anomaly detection....1284
RF statistics....1285
Configure anomaly detection in AP profile (CLI)....1285
Configure anomaly detection in an AP (CLI)....1286
Verify anomaly detection and RF statistics....1287

109 Amazon S3 Support....1289
Amazon S3 support....1290
Restrictions and guidelines....1290
Configure Amazon S3 support (CLI)....1290
Verify Amazon S3 support....1292

110 Amazon Web Services CloudWatch....1295
Amazon Web Services CloudWatch Support....1296
Benefits of Amazon web services CloudWatch service....1296
Configure an AWS CloudWatch Profile (CLI)....1297
Verify AWS CloudWatch configuration....1299

111 Kernel Minidump and Trustzone Upgrade....1301
Kernel minidump and trust zone upgrade....1302
Configure the Minidump from the AP (CLI)....1303
Configure the minidump from the controller (CLI)....1304
Verify minidump configuration....1305

Part IX: Cloud Services....1307

112 Cisco Spaces....1309

Cisco Spaces....1310

Configure Cisco Spaces....1310

Verify Cisco Spaces configuration....1311

113 Sensor Connect backup and restore....1313

Sensor Connect backup and restore....1314

Feature history for Sensor Connect backup and restore....1314

Guidelines for Sensor Connect backup and restore....1315

Restrictions for Sensor Connect backup and restore....1315

Backup the Sensor Connect application....1315

Restore the Sensor Connect application....1316

Sensor Connect launch and Day-0 configuration....1316

114 AP and Sensor Connect support for full filtering....1319

AP and Sensor Connect support for full filtering....1320

Feature history for AP and Sensor Connect support for full filtering....1320

Guidelines for BLE full filtering....1320

Launch Sensor Connect from the Catalyst 9800 controller....1321

115 Using Cloud Monitoring as a Solution for Network Monitoring....1323

Cloud monitoring....1324

When to use cloud monitoring....1324

Cloud monitoring features....1324

Prerequisites for cloud monitoring....1325

Enable cloud monitoring (GUI)....1325

Enable cloud monitoring (CLI)....1325

Onboard the controller using Cisco Meraki dashboard....1326

Verify cloud monitoring....1326

Troubleshoot cloud monitoring....1327

116 Wireless Active Testing with ThousandEyes....1329

Wireless active testing (WAT)....1330

Feature history for wireless active testing with ThousandEyes....1330

Supported controller and AP platforms....1330

Supported security types....1331

Guidelines of wireless active testing....1332

Configure wireless active testing (GUI)....1333

Configure wireless active testing (CLI)....1333

Configure HTTP(S) proxy for WAT (GUI)....1334
Configure HTTP(S) proxy for WAT (CLI)....1334
Configure WAT radio selection and tag the RF profile (GUI)....1335
Configure wireless active testing radio selection (CLI)....1336
Enable WAT management for an individual AP (GUI)....1337
Enable WAT management for an individual AP (CLI)....1337
Enable WAT management for multiple APs using location (GUI)....1338
Enable WAT management for multiple APs using location (CLI)....1339
Enable WAT management for multiple APs using filter (GUI)....1340
Enable WAT management for multiple APs using filter (CLI)....1340
Verify wireless active testing....1341

117 Wireless Active Testing SCEP support....1343
Wireless Active Testing SCEP support....1344
Feature history for Wireless Active Testing SCEP support....1344
SCEP enrollment for Wireless Active Testing....1344
Guidelines for Wireless Active Testing SCEP....1345

Part X: Security....1347

118 MAC Filtering....1351
MAC Filtering....1352
Configuring MAC Filtering for Local Authentication (CLI)....1353
Configuring MAC Filtering (GUI)....1355
Configuring MAB for External Authentication (CLI)....1355

119 Web-Based Authentication....1359
Local web authentication....1360
How local web authentication works....1361
Restrictions....1362
Roles of devices in local web authentication....1362
Banner messages for local web authentication....1363
Authentication states in customized local web authentication....1365
How to configure local web authentication....1366
Configuration examples for local web authentication....1391
External Web Authentication....1397
Configure EWA with single WebAuth server address and default ports (80/443) (CLI)....1397
Configure EWA with multiple web servers and/or ports different than default (80/443)....1399
Configure wired guest EWA with multiple web servers, ports different than default (80/443)....1402
Authenticating sleeping clients....1403
Guidelines for mobility scenarios....1404
Restrictions on authenticating sleeping clients....1404
Configure authentication for sleeping clients (GUI)....1405

Configure authentication for sleeping clients (CLI)....1405
Sleeping Clients With Multiple Authentication....1406
Mobility support for sleeping clients....1406
Supported combinations of multiple authentications....1407
Configure WLAN for 802.1X and local web authentication....1407
Configure a WLAN for MAC authentication bypass and LWA....1408
Configure a WLAN for local web authentication and MAC filtering....1410
Configure a PSK + LWA in a WLAN....1411
Configure a sleeping client....1413
Verify a sleeping client configuration....1414
Multiple Authentication Combinations with 802.1X and LWA....1414
Multiauthentication combinations with 802.1X authentication and local web authentication....1414
Limitations for multi authentication combination of 802.1X and local web authentication....1415
Enable the multiauthentication combinations of 802.1X authentication and local web authentication (CLI)....1415
Verify multiauthentication combination with 802.1X authentication and local web authentication...1416
Title....1416

120 Central Web Authentication....1417
Central web authentication....1418
How to configure ISE....1419
How to configure central web authentication on the controller....1422

121 Private Shared Key....1433
Private preshared keys....1434
Limitations....1434
How identity PSK authentication works....1434
Configure a PSK in a WLAN....1435
Configure a PSK in a WLAN using GUI....1437
Apply a policy profile to a WLAN (GUI)....1438
Apply a policy profile to a WLAN using CLI....1438
Verify a private PSK....1439

122 Multi-Preshared Key....1443
Multi-preshared key....1444
Restrictions....1445
Procedures....1446
Configure a multi-preshared key (GUI)....1446
Configure a multi-preshared key (CLI)....1448
Verify multi-PSK configurations....1450

123 Wireless AAA Authentication Survivability Cache....1453
Wireless AAA authentication survivability cache....1454

Feature history for wireless AAA authentication survivability cache....1454
Configure wireless AAA authentication survivability cache 1455
RADIUS configuration for MSCHAPv2....1462
Configure EAP-TLS on ISE....1463
Verify wireless AAA authentication survivability cache....1463

124 Wi-Fi Protected Access 3....1465
Overview of WPA3....1466
Simultaneous Authentication of Equals....1466
Opportunistic Wireless Encryption....1467
Hash-to-Element (H2E)....1468
YANG (RPC model)....1468
Transition Disable....1470
WPA3 SAE iPSK....1471
WPA3 Configuration Tasks....1471
Configure SAE (WPA3+WPA2 mixed mode)....1471
Configure WPA3 enterprise (GUI)....1474
Configure WPA3 enterprise....1474
Configure WPA3 OWE....1476
Configure WPA3 OWE transition mode (GUI)....1478
Configure WPA3 OWE transition mode....1478
Configure WPA3 SAE (GUI)....1480
Configure WPA3 SAE....1481
Configure a WPA3 SAE iPSK WLAN profile (CLI)....1483
Configure a policy profile with AAA override and VLAN assignment (CLI)....1485
Configure a passphrase in a client authorization policy in the RADIUS server(GUI) 1486
Configure WPA3 SAE H2E (GUI)....1486
Configure WPA3 SAE H2E (CLI)....1487
Configure WPA3 WLAN for Transition Disable....1489
Configure anti-clogging and SAE retransmission (GUI)....1490
Configure anti-clogging and SAE retransmission (CLI)....1491
Verify WPA3 SAE and OWE....1492
Verify WPA3 SAE H2E support in WLAN....1494
Verify WPA3 transition disable in WLAN....1500

125 WPA3 Security Enhancements for Access Points....1505
WPA3 security enhancements for APs....1506
Restrictions....1510
GCMP-256 Cipher and SuiteB-192-1X AKM....1510
SAE-EXT-KEY support....1513
AP beacon protection....1521
Multiple cipher support per WLAN....1523
OWE support with GCMP-256 cipher....1526

126 IP Source Guard....1529
IP source guard....1530
Configure IP source guard (GUI)....1530
Configure IP source guard (CLI)....1530

127 Traffic Filtering on AP by Source IP ACL....1533
Traffic filtering on AP by source IP ACL....1534
Feature History for Traffic Filtering on AP by Source IP ACL....1534
Configure AP management ACL on controller....1534
Verify mapped ACLs....1537

128 802.11w....1539
Protected management frames with 802.11w....1540
Integrity Group Temporal Key (IGTK)....1541
Broadcast or multicast integrity protocol (BIP)....1542
SA teardown protection....1542
Prerequisites for 802.11w....1543
Restrictions for 802.11w....1543
Configure 802.11w....1543
Configure 802.11w (GUI)....1543
Configure 802.11w (CLI)....1544
Disable 802.11w....1545
Monitor 802.11w....1546

129 Management Frame Protection....1549
Information About Management Frame Protection....1550
Restrictions for Management Frame Protection....1551
Configure Management Frame Protection (CLI)....1551
Verifying Management Frame Protection Settings....1552

130 IPv4 ACLs....1555
Network security with ACLs....1556
ACL overview....1556
Supported ACLs....1557
ACEs and fragmented and unfragmented traffic....1559
Standard and extended IPv4 ACLs....1560
Hardware and software treatment of IP ACLs....1564
IPv4 ACL interface considerations....1565
Restrictions for configuring IPv4 access control lists....1566
How to configure ACLs....1567
Configure IPv4 ACLs (GUI)....1567

Configure IPv4 ACLs....1567
Create a numbered standard ACL (GUI)....1567
Create a numbered standard ACL (CLI)....1568
Create a numbered extended ACL (GUI)....1569
Create a numbered extended ACL (CLI)....1570
Create named standard ACLs (GUI)....1573
Create named standard ACLs (CLI)....1573
Create extended named ACLs (GUI)....1574
Create extended named ACLs (CLI)....1575
Apply an IPv4 ACL to an interface (GUI)....1576
Apply an IPv4 ACL to an interface (CLI)....1577
Apply ACL to policy profile (GUI)....1578
Apply ACL to policy profile (CLI)....1578
Configuration examples for ACLs....1579
Examples: including comments in ACLs....1579
Examples: apply an IPv4 ACL to a policy profile in a wireless environment....1579
IPv4 ACL Configuration Examples....1580
Monitor IPv4 ACLs....1583

131 Downloadable ACL....1585

Downloadable ACLs....1586
Feature history for downloadable ACL....1586
Scale considerations for downloadable ACL....1587
Restrictions for downloadable ACL....1587
Configuring dACL name and definition in Cisco ISE....1588
Configure dACL in a controller (CLI)....1588
Configure explicit authorization server list (CLI)....1589
Verify dACL configuration....1590

132 DNS-Based Access Control Lists....1593

DNS-based access control lists....1594
Restrictions on DNS-based access control lists....1595
Define ACLs....1596
Methods for applying ACLs....1596
Configure ISE for central web authentication (GUI)....1597
View DNS-based access control lists....1598
Configuration examples for DNS-based access control lists....1598
Verify DNS snoop agent (DSA)....1599
URL filters....1601
FlexConnect mode....1602
Local mode....1604
Configure ISE for central web authentication....1606
Create authorization profiles (GUI)....1607
Map authorization profiles to authentication rule (GUI)....1607

Map authorization profiles to authorization rule (GUI)....1608
FlexConnect client IPv6 support with WebAuth pre and post ACL....1608
FlexConnect client IPv6 support with WebAuth pre and post ACL....1608
Enable pre-authentication ACL for LWA and EWA (GUI)....1609
Enable pre-authentication ACL for LWA and EWA (CLI)....1610
Enable post-authentication ACL for LWA and EWA (GUI)....1611
Enable post-authentication ACL for LWA and EWA (CLI)....1611
Enable DNS ACL for LWA and EWA (GUI)....1612
Enable DNS ACL for LWA and EWA (CLI)....1612
Verify FlexConnect client IPv6 support with WebAuth pre and post ACL....1613

133 Allowed List of Specific URLs....1615
Allowed lists of specific URLs....1616
Add URL to allowed list....1616
Portal resolving to multiple IP addresses....1618
Verify URLs on the allowed list....1618

134 Cisco Umbrella WLAN....1621
Cisco Umbrella WLAN....1622
Register Controller to Cisco Umbrella account....1622
Restrictions for Umbrella WLAN....1623
Configure Cisco Umbrella WLAN....1624

135 RADIUS Server Load Balancing....1639
RADIUS server load balancing....1640
Prerequisites for RADIUS server load balancing....1642
RADIUS server load balancing restrictions....1642
Enable load balancing for a named RADIUS server group (CLI)....1643

136 AAA Dead-Server Detection....1645
AAA dead-server detection....1646
Prerequisites for AAA dead-server detection....1646
Restrictions for AAA dead-server detection....1647
Configure AAA dead-server detection (CLI)....1647

137 ISE Simplification and Enhancements....1651
Security utilities configuration options....1652
Configure multiple RADIUS servers....1653
Verify AAA and Radius server configurations....1653
Captive bypassing....1654
Configure captive bypassing for WLAN in LWA and CWA (GUI)....1655
Configure captive bypassing for WLAN in LWA and CWA....1656

DHCP option 55 and 77 parameters....1657
Configure the DHCP options 55 and 77 to ISE using the GUI....1657
Configure the device to send DHCP options 55 and 77 to ISE (CLI)....1658
Configure the EAP request timeout using the GUI....1659
Configure EAP request timeout....1659
Configure the EAP request timeout in wireless security....1660
Captive portals....1661
Configure captive portal (GUI)....1661
Configure captive portal....1662
Example: Captive portal configuration....1665

138 RADIUS DTLS....1667
RADIUS DTLS....1668
Requirement: Port RADIUS DTLS support to BINOS platforms....1669
Configure RADIUS DTLS server (CLI)....1669
Configure DTLS Dynamic Author (CLI)....1677
Enable DTLS for client (CLI)....1678
Verify RADIUS DTLS server configuration....1682
Clear RADIUS DTLS specific statistics....1682

139 Policy Enforcement and Usage Monitoring....1685
Policy enforcement and usage monitoring....1686
Configure policy enforcement and enable CoA (CLI)....1686
Example: Configure policy enforcement and usage monitoring....1687
Verify policy usage and enforcement....1688

140 Local Extensible Authentication Protocol....1689
Local EAP....1690
Restrictions for local EAP....1690
Configure local EAP profile (CLI)....1690
Configure local EAP profile (GUI)....1691
Configure AAA authentication (GUI)....1692
Configure AAA authorization method (GUI)....1692
Configure AAA authorization method (CLI)....1693
Configure local advanced methods (GUI)....1694
Configure WLAN (GUI)....1694
Configure WLAN (CLI)....1695
Create a user account (CLI)....1696
Attach a policy profile to a WLAN interface (GUI)....1697
Deploy policy tag to access points (GUI)....1698

141 Local EAP Ciphersuite....1699
Local EAP Ciphersuite....1700

Restrictions for Local EAP Ciphersuite....1701
Configure Local EAP Ciphersuite (CLI)....1701

142 Authentication and Authorization Between Multiple RADIUS Servers....1703
Authentication and authorization between multiple RADIUS servers....1704
Configure explicit authentication and authorization server list (GUI)....1704
Configure explicit authentication server list (GUI)....1705
Configure explicit authentication server list (CLI)....1705
Configure explicit authorization server list (GUI)....1707
Configure explicit authorization server list (CLI)....1707
Configure authentication and authorization list for 802.1X security (GUI)....1710
Configure authentication and authorization list for 802.1X security (CLI)....1710
Configure web authentication for WLAN with split authentication and authorization servers....1711
Configure authentication and authorization list for web authentication (GUI)....1711
Configure authentication and authorization list for web authentication (CLI)....1712
Verify split authentication and authorization configuration....1713
Configuration examples....1714
RADIUS Message-Authenticator support for non-EAP requests....1715
Configure RADIUS Message-Authenticator support for non-EAP requests (CLI)....1716

143 CUI Information in RADIUS Accounting....1717
CUI information in RADIUS accounting request....1718
Prerequisites and restrictions....1718
Add CUI information in a RADIUS accounting request....1718
Verify CUI information in a RADIUS accounting request....1719

144 Secure LDAP....1721
Lightweight Directory Access Protocol (LDAP)....1722
SLDAP prerequisite....1723
SLDAP restrictions....1723
Configure SLDAP (CLI)....1723
Configure a AAA server group (GUI)....1725
Configure a AAA server group (CLI)....1726
Configure search and bind operations for authentication requests (CLI)....1727
Configure a dynamic attribute map on a SLDAP server (CLI)....1728
Verify the SLDAP configuration....1729

145 Network Access Server Identifier....1731
Network Access Server Identifier (NAS-ID)....1732
Create a NAS ID policy (GUI)....1732
Create a NAS ID policy (CLI)....1733
Attach a policy to a tag (GUI)....1734
Attach a policy to a tag (CLI)....1734

Verify the NAS ID configuration....1735

146 Locally Significant Certificates....1737

Locally significant certificates....1738
Certificate provisioning in controllers....1738
How device certificate enrollment works....1739
How certificate provisioning works on lightweight APs....1740
Restrictions for locally significant certificates....1740
Provisioning locally significant certificates....1741
Configure RSA key for PKI trustpoint....1741
Configure PKI trustpoint parameters....1742
Authenticate and enroll a Public Key Infrastructure (PKI) trustpoint (GUI)....1744
Authenticate and enroll the PKI trustpoint with CA server (CLI)....1744
Configure AP join attempts with LSC certificate....1747
Configure AP join attempts with LSC certificate (CLI)....1747
Configure subject-name parameters in LSC certificate....1748
Configure key size for LSC certificate....1748
Configure trustpoint for LSC provisioning on an AP....1749
Configure an AP LSC provision list (GUI)....1750
Configure an AP LSC provision list (CLI)....1750
Configure LSC provisioning for all the APs (GUI)....1751
Configure LSC provisioning for all APs (CLI)....1752
Configure LSC provisioning for the APs in the provision list....1753
Import a CA certificate to the trustpool (GUI)....1753
Import a CA certificate to the trustpool....1754
Clean the CA certificates imported in trustpool (GUI)....1755
Clean CA certificates imported in trustpool (CLI)....1755
Create a new trustpoint dedicated to a single CA certificate....1756
Verify LSC configuration 1757
Configure management trustpoint to LSC....1758
Configure management trustpoint to LSC (CLI)....1758
AP certificate renewal (LSC)....1759
AP certificate renewal (LSC)....1759
Benefits of AP certificate renewal (LSC)....1759
Requirements for AP certificate renewal (LSC)....1759
Restrictions for AP certificate renewal (LSC)....1760
Configure AP certificate auto-renewal (LSC) (GUI)....1760
Configure AP certificate renewal (LSC) (CLI)....1761
Configure forced renewal of AP certificate (LSC) for individual or multiple APs (GUI)....1762
Configure forced renewal of AP Certificate (LSC) based on site tag (GUI)....1763
Configure forced renewal of AP Certificate (LSC) based on site tag (CLI)....1763
Configure forced renewal of all APs (GUI)....1764
Verify AP certificate renewal (LSC)....1764
MIC and LSC APs joining the controller....1765

Support for MIC and LSC APs joining the controller....1765
Recommendations and limitations....1766
MIC and LSC access point authentication....1766
Configure LSC on the controller (CLI)....1767
Enable the AP certificate policy on the APs (CLI)....1767
Configure the AP policy certificate (GUI)....1769
Configure the allowed list of APs to join the controller (CLI)....1769
Verifying the configuration status....1770
LSC fallback APs....1771
LSC fallback APs....1771
Troubleshoot LSC fallback state....1771
Recover from LSC fallback....1773
Configuring controller self-signed certificate for wireless AP join....1773
Use cases for controller self-signed certificate for wireless AP join....1774
Prerequisites for controller self-signed certificate for wireless AP join....1774
Configure clock calendar (CLI)....1775
Enable HTTP server using CLI....1775
Configure CA server (CLI)....1776
Configure trustpoint (CLI)....1779
Authenticate and enroll the PKI trustpoint with CA server....1781
Tag wireless management trustpoint name....1782
Disable PKI server....1782
Verify controller certificates for wireless AP join....1783

147 Public Key Infrastructure Management....1785
About PKI management (GUI)....1786
Restriction: CSR generation on the GUI may hang indefinitely....1786
Configuration Tasks....1787
Authenticate and enroll a Public Key Infrastructure (PKI) trustpoint (GUI)....1787
Generate an AP self-signed certificate (GUI)....1787
Add the CA server (GUI)....1788
Add an RSA or EC Key for PKI trustpoint (GUI)....1788
Add and manage certificates 1789

148 Controller Self-Signed Certificate for Wireless AP Join....1793
Controller self-signed certificates for wireless AP Join....1794
Use cases for controller self-signed certificate for wireless AP join....1794
Requirement: Use minimum RSA key size for DTLS connections....1794
Prerequisites for controller self-signed certificate for wireless AP join....1795
Configure clock calendar (CLI)....1795
Enable HTTP server using CLI....1796
Configure CA server (CLI)....1797
Configure trustpoint (CLI)....1799
Authenticate and enroll the PKI trustpoint with CA server....1801

Tag wireless management trustpoint name....1802
Disable PKI server....1803
Verify controller certificates for wireless AP join....1804

149 Managing Rogue Devices....1807
Rogue devices and detection....1808
Restrictions on rogue detection....1808
Rogue AP containment and attack signatures....1808
Cisco Prime Infrastructure interaction and rogue detection....1809
Rogue containment (Protected Management Frames (PMF) enabled)....1810
Off-channel PMF rogue containment....1810
AP impersonation detection....1811
Rogue detection security level....1812
Configuring rogue detection (GUI)....1814
Configure rogue detection (CLI)....1815
Configure RSSI deviation notification threshold for rogue APs (CLI)....1816
Configure management frame protection (GUI)....1817
Configure Management Frame Protection (CLI)....1817
Enable AP authentication....1818
Configure off-channel PMF rogue containment (GUI)....1819
Configure off-channel PMF rogue containment (CLI)....1819
Verify management frame protection....1820
Verify rogue events....1821
Verify rogue detection....1821
Verify off-channel PMF rogue containment....1823
Examples: rogue detection onfiguration....1823
Configure rogue policies (GUI)....1824
Configure rogue policies (CLI)....1825
Wireless Service Assurance (WSA) rogue events....1827
Wireless service assurance rogue events....1827
Monitor wireless service assurance rogue events....1828
Rogue AP scale modes per class....1829
Feature history for rogue full scale quotas and priorities....1830
Advantages and disadvantages of role-scale modes....1830
Configure rogue AP scale (CLI)....1831
Verify rogue scale details....1832
Rogue PMF containment....1833
Enable rogue PMF containment....1833
Verify PMF containment....1834

150 Classifying Rogue Access Points....1837
Rogue AP classifications....1838
Rogue containment methods....1839
Guidelines and restrictions for classifying Rogue APs....1840

How to classify rogue APs....1841
Monitor rogue classification rules....1846
Examples: classifying rogue AP....1847

151 Advanced WIPS....1849

Advanced WIPS....1850
Advanced WIPS signatures and definitions....1850
Guidelines and Restrictions....1852
Enable advanced WIPS....1853
Syslog messages for advanced WIPS....1853
Advanced WIPS solution components and capabilities....1854
Supported modes and platforms....1855
Enable advanced WIPS (GUI)....1855
Enable advanced WIPS (CLI)....1855
Configure syslog threshold for advanced WIPS (CLI)....1857
View advanced WIPS alarms (GUI)....1857
Verify advanced WIPS....1858
Verify syslog configuration for advanced WIPS....1859

152 Cisco TrustSec....1861

Cisco TrustSec....1862
Cisco TrustSec features....1863
Configure a device SGT manually (CLI)....1864
Configure ISE for TrustSec (CLI)....1865
Verify Cisco TrustSec configuration....1866
Security Group Access Control List (SGACL)....1868
Guidelines and restrictions of SGACL....1869
Enable SGACL on the AP (GUI)....1870
Enable SGACL on the AP (CLI)....1870
Policy enforcement....1871
Inline tagging....1874
SGACL and environment data download over REST....1876
Cisco TrustSec - ISE communication over IPv6 1889
Limitations of IPv4....1889
Configure ISE for TrustSec (CLI)....1889
Configure TrustSec SXP with IPv6 (GUI)....1891
Configure TrustSec SXP with IPv6....1892
Verify Cisco TrustSec PAC over IPv6 RADIUS....1893
TrustSec IP-SGT High Availability binding....1894
Verify TrustSec IP-SGT High Availability binding support....1895

153 SGT Inline Tagging and SXPv4....1899

SGT inline tagging on AP and SXPv4....1900

Create an SXP profile....1900
Configure SGT inline tagging on APs....1901
Configure an SXP connection (GUI)....1901
Configure an SXP connection....1902
Verify SGT push to APs....1903

154 Ciphersuites for CAPWAP-DTLS....1907
Default ciphersuites supported for CAPWAP-DTLS....1908
Configure multiple ciphersuites (CLI)....1908
Set server preference (CLI)....1909
Verify operational ciphersuites and priority....1910

155 Configure Secure Shell 1911
SSH....1912
SSH and device access....1912
SSH servers and integrated clients....1912
SSH configuration guidelines....1913
SCP....1913
Secure copy protocol....1914
SFTP support....1914
Prerequisites for configuring SSH....1914
Restrictions for configuring SSH....1915
Set up the device to run SSH....1915
Configure the SSH server....1917
Monitor the SSH configuration and status....1919

156 MACsec support on APs....1921
MACsec support on AP 1922
Common MACsec terminology....1923
Supported AP....1923
MACsec modes, authentication options, and supported ciphers....1923
Restrictions....1924
MACsec failover mechanism....1924
Configuring MACsec with PSK....1925
Configuring MACsec with IEEE 802.1X....1932
Auto MACsec for wireless APs....1937
Benefits of auto MACsec....1938
Enable auto MACsec (GUI)....1938
Enable auto MACsec in AP join profile (CLI)....1939
Verify auto MACsec....1939
Recovery for Auto MACsec....1940

157 FIPS....1943

Federal Information Processing Standard (FIPS)....1944
Guidelines and restrictions for FIPS....1944
FIPS self-tests....1945
Configure FIPS (CLI)....1946
How FIPS configuration works in high availability setups....1947
Verify FIPS configuration....1949
Common criteria....1949
Configure the common criteria (CLI)....1950
Verify CC configuration....1951
Check points for CC mode operation....1951
TLS signature algorithm enhancements for NDcPP v3.0e....1953
Signature algorithms and ciphersuites in TLS and DTLS applications....1954
Configure syslog TLS profile ciphersuites (CLI)....1957
Configure HTTP secure ciphersuites....1958
Verify Syslog TLS profile....1958
Verify HTTP client and server applications....1958
HTTPS server security considerations with NDcPP 4.0....1959
Best practices for HTTPS server security....1960
Configure HTTP server security settings (CLI)....1960
Configure HTTP client security settings (CLI)....1961

158 Internet Protocol Security....1963
Internet Protocol Security (IPsec)....1964
IKEv1 transform sets....1964
Configure IPsec using IKEv1....1965
IKEv2 transform sets....1968
Configure IPsec using IKEv2....1968
IPsec transforms and lifetimes....1971
Use of X.509 with IKE version....1972
IPsec session interruption and recovery....1973
Example: Configure IPsec using ISAKMP....1973
Verify IPSec traffic....1973
Example: Configure IPsec using IKEv2....1975
Verify IPSec with IKEv2 traffic....1975

159 Transport Layer Security Tunnel Support....1979
Transport layer security tunnels....1980
Configure a transport layer security tunnel....1980
Transport layer security tunnel verification....1983

160 Configuring RFC 5580 Location Attributes....1987
RFC 5580 location attributes....1988
Location-capable attribute....1989

Restriction for configuring RFC 5580 location attributes....1990
Configure location delivery based on out-of-band agreement (CLI)....1990
Configure the location-capable attribute (CLI)....1990
Configure a civic profile (CLI)....1991
Configure a geo profile (CLI)....1993
Configure an operator name (CLI)....1994
Associate a location attribute with a user location (CLI)....1996
Associate the NAS location with location attributes (CLI)....1997
Verify RFC 5580 location attribute configuration....1998

161 IP MAC Binding....2001
IP MAC binding....2002
Use cases for no IP MAC binding....2002
Disable IP MAC binding (CLI)....2003
Verify IP MAC binding....2004

162 Disabling IP Learning in FlexConnect Mode....2005
Disabling IP learning in FlexConnect mode....2006
Restrictions for disabling IP learning in FlexConnect mode....2006
Disable IP learning in FlexConnect mode....2006
MAC entry verification from database....2007

163 Disabling Device Tracking to Support NAC Devices....2009
Disabling device tracking to support NAC devices....2010
Feature history for disable device tracking support for NAC devices....2010
Restrictions for disabling device tracking to support NAC devices....2010
Disable device tracking for wireless clients (CLI)....2011
Verify ARP broadcast....2012

164 Disabling IP Learning in Local Mode....2013
Disabling IP learning in local mode....2014
Restrictions for disabling IP learning in local mode....2014
Disable IP learning in local mode (CLI)....2015
Verify MAC entries database....2016
ARP broadcast verification....2016

165 Security-Enhanced Linux....2017
Security-Enhanced Linux....2018
Configure SELinux in the EXEC mode....2018
Configure SELinux in the global configuration mode....2019
Examples for SELinux....2020
SELINUX syslog message reference....2020

Verify count of denials....2021
Verify SELinux enablement....2022

Part XI: Mobility....2023

166 Mobility....2025
Mobility....2026
Intracontroller roaming....2026
Intercontroller roaming....2027
Intersubnet roaming....2029
SDA roaming....2031
Mobility groups....2033
Supported platforms....2034
Configure the mobility feature (GUI)....2038
Configure the mobility group (CLI)....2039
Configure inter-release controller mobility (GUI)....2041
Configure inter-release controller mobility (CLI)....2041
Verify mobility....2043

167 NAT Support on Mobility Groups....2049
NAT support for mobility groups....2050
Restrictions for NAT support on mobility groups....2051
Functionalities supported on mobility NAT....2051
Configure a mobility peer (CLI)....2052
Verify NAT support on mobility groups....2053

168 Static IP Client Mobility....2055
Static IP client mobility....2056
Restrictions for static IP client mobility....2056
Configure a static IP client mobility profile (GUI)....2056
Configure a static IP client mobility (CLI)....2057
Verify static IP client mobility....2058

169 Mobility Domain ID - 802.11i Roaming....2061
Mobility domain ID–802.11i roaming....2062
Verify mobility domain ID - 802.11i roaming....2063

170 802.11r Support for Flex Local Authentication....2065
802.11r support for FlexConnect local authentication....2066
802.11r support verification for flex local authentication....2066

171 Opportunistic Key Caching....2069
Opportunistic key caching....2070
Enable opportunistic key caching....2070
Enable opportunistic key caching (GUI)....2071
Verify opportunistic key caching....2071

Part XII: High Availability....2073

172 High Availability....2075
High Availability....2076
Prerequisites for High Availability....2078
High Availability restrictions....2079
Best practices for RP port configuration....2081
Configure High Availability (CLI)....2081
Disable High Availability....2083
Copy a WebAuth tar bundle to the standby controller (GUI)....2084
System and network fault handling....2085
Handling recovery mechanism....2104
Verify high availability configurations....2105
Verify AP or client SSO statistics....2105
Verify high availability....2107
ACI network controllers....2111
Configure a switchover....2115
Auto-upgrade feature....2115
Auto-upgrade use cases....2116
Configuration workflow....2116
Configure auto-upgrade (CLI)....2117
Link layer discovery protocol use cases....2117
Enable LLDP (CLI)....2118
Enable LLDP timers (CLI)....2118
Enable LLDP TLV-Select (CLI)....2119
LLDP verification....2120
Reload reason history....2122
Feature history....2123
Request reload reason history using YANG....2123
Reload reason history verification....2125

173 Redundancy Management Interface....2129
Redundancy management interfaces....2130
Gateway monitoring....2130
Gateway statistics monitoring....2132
Gateway reachability detection....2133

Standby monitoring....2135
Standby controller health monitoring using programmatic interfaces....2140
Standby controller health monitoring (CLI)....2140
Gateway-monitoring configuration verification....2143
RMI IPv4 configuration verification....2144
RMI IPv6 configuration verification....2145
Redundancy port interface configuration verification....2146
Redundancy management interfaces....2148
Controller operations with Redundant Management Interface....2151
Dual stack configurations on management vlans with RMI....2151
RMI-based high-availability pairings....2152
HA pairing without previous configuration....2153
Paired controllers....2154
Upgrade from Cisco IOS XE 16.1.x to a later release....2155
Configure redundancy management interface (GUI)....2156
Configure redundancy management interface (CLI)....2157

Part XIII: Quality of Service....2161

174 Quality of Service....2163
Wireless Quality of Service....2164
Wireless QoS targets....2164
Wireless QoS mobility....2166
Precious metal policies for wireless QoS....2166
Prerequisites for wireless QoS....2167
Restrictions for QoS on wireless targets....2167
Bidirectional rate limiting....2169
Apply metal policy with bidirectional rate limiting....2170
Requirements for bidirectional rate limiting configuration....2171
Configure the metal policy on the SSID....2171
Configure a metal policy on your client device....2172
Configure bidirectional rate limiting for all traffic....2173
Configure Bi-Directional Rate Limiting Based on Traffic Classification....2173
Apply bidirectional rate limiting policy map to policy profile....2175
Apply metal policy with bidirectional rate limiting....2176
Per-client bidirectional rate limiting....2178
Use case for per-client bidirectional rate limiting....2178
Process for per-client bidirectional rate limiting....2179
Prerequisites for per-client bidirectional rate limiting....2179
Restrictions on per-client bidirectional rate limiting....2179
Configure per-client bidirectional rate limiting (GUI)....2179
Verify per client bi-directional rate limiting....2180
Configure bi-directional rate limiting using AAA override (CLI)....2180
Verify bi-directional rate-limit....2181

How to configure wireless QoS....2183
Configure a policy map with class map (GUI)....2183
Configure a class map (CLI)....2184
Configuring Policy Profile to Apply QoS Policy (GUI)....2185
Configure policy profile to apply QoS policy (CLI)....2186
Apply policy profile to policy tag (GUI)....2187
Apply policy profile to policy tag (CLI)....2187
Attach policy tag to an AP....2188
Configure custom QoS mapping (CLI)....2189
Configure a DSCP-to-user priority mapping exception....2191
Configure trust upstream DSCP value....2192

175 Wireless Auto-QoS....2195
Auto QoS information....2196
Configure wireless auto-QoS on profile policy (CLI)....2196
Disable wireless autoqos....2197
Rollback auto-QoS configuration (GUI)....2198
Rollback AutoQoS configuration (CLI)....2198
Clear wireless autoqos policy profile (GUI)....2199
Clear wireless autoqos policy profile (CLI)....2199
View Auto-QoS on policy profile (CLI)....2200

176 Native Profiling....2203
Native profiling....2204
Create a class map (GUI)....2205
Create a class map (CLI)....2206
Create a service template (GUI)....2208
Create a service template (CLI)....2208
Create a parameter map (CLI)....2210
Create a policy map (GUI)....2210
Create a policy map (CLI)....2211
Configure native profiling in local mode (CLI)....2214
Verify native profile configuration....2214

177 Air Time Fairness....2217
Air time fairness....2218
Restrictions on Cisco air time fairness....2219
Cisco air time fairness use cases....2219
Configure Cisco air time fairness....2220
Create a Cisco ATF profile (GUI)....2220
Create Cisco ATF profile (CLI)....2221
Attach Cisco ATF profile to a policy profile (GUI)....2222
Attach Cisco ATF profile to a policy profile (CLI)....2222

Enable ATF in the RF profile (GUI)....2223
Enable ATF in the RF profile (CLI)....2224
Cisco ATF configuration verification....2225
Cisco ATF statistics verification....2225

178 IPv6 Non-AVC QoS Support....2227
IPv6 non-AVC QoS support....2228
Mark DSCP values in an IPv6 packet (CLI)....2228
Drop an IPv6 packet with DSCP values (CLI)....2229
Configure policing for IPv6 traffic (CLI)....2230
IPv6 non-AVC QoS verification....2231

179 QoS Basic Service Set Load....2233
QoS basic set service load....2234
Configure Wi-Fi multimedia (CLI)....2234
Enable QoS basic set service load (CLI)....2236
QoS basic set service load verification....2237

Part XIV: IPv6....2239

180 IPv6 Client IP Address Learning....2241
IPv6 client address learning....2242
Prerequisites for IPv6 client address learning....2242
Configure IPv6 interface on a switch (GUI)....2242
Configure IPv6 on Interface (CLI)....2243
Address assignment using SLAAC....2244
Stateful DHCPv6 address assignment....2245
Router solicitation....2251
Router advertisement....2251
Neighbor discovery....2254
Neighbor discovery suppression....2255
IPv6 address tracking for wireless clients....2255
Configure unknown address multicast neighbor solicitation forwarding (CLI)....2256
Native IPv6....2257
IPv6....2257
General guidelines and unsupported features of IPv6....2258
Configure IPv6 addressing (CLI)....2258
Create an AP join profile (GUI)....2260
Create an AP join profile (CLI)....2260
Configure the primary and backup Controller (GUI)....2261
Configure primary and backup controller (CLI)....2261
Verify IPv6 configuration....2262

181 IPv6 ACL....2263
IPv6 ACL....2264
Understand IPv6 ACLs....2264
Types of ACL....2264
Prerequisites for configuring IPv6 ACL....2265
Restrictions for configuring IPv6 ACL....2265
Configure IPv6 ACLs....2265
Default IPv6 ACL configuration....2265
Interact with other features and switches....2266
Create an IPv6 ACL (GUI)....2266
Create an IPv6 ACL (CLI)....2267
Create WLAN IPv6 ACL (GUI)....2269
Create WLAN IPv6 ACL (CLI)....2270
Verify IPv6 ACL....2270
Display IPv6 ACLs (CLI)....2271
Example: Create an IPv6 ACL....2271
Example: Apply an IPv6 ACL to a policy profile in a wireless environment....2272
Display IPv6 ACLs (CLI)....2272
Example: Display IPv6 ACLs....2273
Example: Configure RA throttling....2273

182 Configuring IPv6 Mobility....2275
IPv6 client mobility....2276
Using router advertisement....2277
Router advertisement throttling....2278
IPv6 address learning....2278
Handling multiple IP addresses....2278
IPv6 configuration....2279
Prerequisites for IPv6 client mobility....2279
Monitor IPv6 client mobility....2279

183 IPv6 Support on Flex and Mesh....2281
IPv6 support on FlexConnect + mesh deployment....2282
Configure IPv6 support for FlexConnect + Mesh (CLI)....2282
Configure preferred IP address as IPv6 (GUI)....2283
Configure preferred IP address as IPv6 (CLI)....2283
Verify IPv6 on FlexConnect + Mesh....2284

184 IPv6 CAPWAP UDP Lite Support....2287
UDP Lite....2288
Enable UDP lite support (CLI)....2288
Verify UDP lite support configuration....2289

185 Neighbor Discovery Proxy....2291
Neighbor discovery....2292
Neighbor discovery proxy....2292
Duplicate address detection proxy....2293

186 Address Resolution Protocol Proxy....2295
Address resolution protocol....2296
Address resolution protocol proxy....2296

187 IPv6 Ready Certification....2299
IPv6 ready certification....2300
Configure IPv6 route information (CLI)....2300
Verify IPv6 route information....2301

Part XV: CleanAir....2303

188 Cisco CleanAir....2305
Cisco CleanAir....2306
Cisco CleanAir-related terms....2307
Cisco CleanAir components....2307
Prerequisites for CleanAir....2308
Difference between Spectrum Intelligence and Cisco CleanAir....2309
EDRRM and AQR update mode....2311
Configure CleanAir....2311
Enable CleanAir for the 2.4-GHz band (GUI)....2311
Enable CleanAir for the 2.4-GHz band (CLI)....2312
Configure a CleanAir alarm for 2.4-GHz air-quality and devices (CLI)....2312
Configure interference reporting for a 2.4-GHz device (GUI)....2313
Configure interference reporting for a 2.4-GHz device (CLI)....2314
Enable CleanAir for the 5-GHz band (GUI)....2315
Enable CleanAir for the 5-GHz band (CLI)....2315
Configure interference reporting for a 5-GHz device (GUI)....2315
Configure interference reporting for a 5-GHz device (CLI)....2316
Configure event driven RRM for a CleanAir event (GUI)....2318
Configure EDRRM for a CleanAir event (CLI)....2318
Verify CleanAir support on connected APs....2319
CleanAir Pro scanning....2320
Configuration examples for CleanAir....2324
CleanAir Pro scanning....2324
Enable CleanAir Pro scan features (CLI)....2326
Monitor CleanAir Pro statistics (GUI)....2327

Verify CleanAir Pro scanning details....2328
Verify CleanAir parameters....2329
Configure air quality traps (CLI)....2330
Monitor with CleanAir device clusters....2330
CleanAir FAQs....2331

189 Bluetooth Low Energy....2333
Bluetooth low energy....2334
Enable Bluetooth low energy beacon (GUI)....2334
Enable Bluetooth low energy beacon (CLI)....2335

190 Persistent Device Avoidance....2337
Cisco persistent device avoidance....2338
Configure persistent device avoidance (GUI)....2338
Configure persistent device avoidance (CLI)....2339
Verify persistent device avoidance....2339

191 Spectrum Intelligence....2343
Spectrum Intelligence....2344
Configure spectrum intelligence (CLI)....2344
Verify spectrum intelligence information....2345
Debugg spectrum intelligence on supported APs (CLI)....2346

192 Spectrum Analysis....2349
Spectrum analysis....2350
Live spectrum analysis....2350
Perform AP spectrum analysis (GUI)....2350
Configure spectrum analysis (CLI)....2351
Verify spectrum analysis....2352

Part XVI: Mesh Access Points....2353

193 Mesh Access Points....2355
Mesh network....2356
Mesh RRM feature support and evolution across AP generations....2357
Restrictions for mesh access points....2357
Country codes....2358
Mesh interoperability between controllers....2358
Enable wireless mesh profile (CLI)....2359
Associate wireless mesh to an AP profile (CLI)....2360
Verify mesh configuration....2360
Mesh Authorization and Security....2369

MAC authorization....2369
EAP authentication....2372
Preshared key provisioning....2374
Provision LSC for bridge-mode and mesh APs (GUI)....2376
Provision LSC for bridge-mode and mesh APs....2377
Intrusion detection system....2378
Bridge group names....2379
Configure a bridge group name (GUI)....2380
Configure a bridge group name (CLI)....2380
Mesh background scanning....2381
Background scanning and MAP fast ancestor find mode....2382
Configure background scanning (GUI)....2382
Configure background scanning....2383
Configure AP fast ancestor find mode (GUI)....2383
Configure background scanning and MAP fast ancestor find mode (Task)....2384
Verify background scanning and MAP fast ancestor find....2385
Mesh serial backhaul....2385
Enable serial backhaul in radio profile (GUI)....2386
Enable mesh configurations in radio profile (CLI)....2387
Enable serial backhaul (CLI)....2388
Verify mesh serial backhaul....2389
Mesh backhaul RRM....2390
Radio Resource Management on mesh....2391
Configure RRM on mesh backhaul (CLI)....2392
Configure RRM channel assignment for root access points globally....2392
Configure RRM channel assignment for an AP....2393
RRM DCA status verification....2394
Dynamic frequency selection....2395
Spectrum Intelligence for mesh....2396
Mesh backhaul at 2.4 GHz and 5 GHz 2396
Configure mesh backhaul (CLI)....2397
Specify the backhaul slot for the root AP (GUI)....2397
Specify the backhaul slot for the root AP (CLI)....2398
Mesh backhaul verification....2398
Backhaul client access....2399
Configure dot11ax rates on mesh backhaul per AP (GUI)....2401
Configure Dot11ax rates on mesh backhaul in mesh profile (GUI)....2401
Configure wireless backhaul data rate (CLI)....2402
Configure data rate per AP (CLI)....2403
Configure data rate using mesh profile (CLI)....2404
Verify dot11ax rates on mesh backhaul....2405
Ethernet bridging....2405
Configure Ethernet bridging (GUI)....2406
Configure Ethernet bridging (CLI)....2406
Multicast over mesh Ethernet bridging network....2408

Configure multicast modes over mesh....2409
Workgroup bridge....2410
DHCP and NAT functionality on root AP (RAP)....2410
How DHCP and NAT functionalities work on root AP - IPv4 scenario....2411
Configure DHCP server on root AP (CLI)....2411
DHCP server for root AP configuration verification....2412
Mesh convergence....2412
Noise-tolerant fast detection....2413
Configure mesh convergence (CLI)....2413
Mesh convergence verification....2414
Mesh network recovery mechanisms....2414
Mesh daisy chaining....2415
Prerequisites for mesh Ethernet daisy chaining....2415
Restrictions for mesh Ethernet daisy chaining....2415
Configure mesh Ethernet daisy chaining (CLI)....2416
Enable mesh Ethernet daisy chaining....2416
Mesh Ethernet daisy chaining verification....2417
FlexConnect + bridge mode....2417
Flex resilient with Flex and bridge modes....2418
Mesh AP Roles and Topology....2425
Select a preferred parent (GUI)....2426
Select a preferred parent (CLI)....2426
Change the role of an AP (GUI)....2427
Change the role of an AP (CLI)....2427
Mesh leaf node....2428
Configure subset channel synchronization....2429
Configure battery state for mesh AP (GUI)....2430
Configure battery state for mesh AP (CLI)....2430
Link test....2431
Use a link test on mesh backhaul (GUI)....2431
Use a link test on mesh backhaul....2432
Air time fairness on mesh....2433
Configure ATF on mesh (GUI)....2434
Configure ATF on mesh....2434
Create an ATF policy for a MAP....2435
Create an ATF policy (GUI)....2435
Add an ATF to a policy profile (GUI)....2436
Enable ATF mode in an RF profile (GUI)....2436
Verify ATF configuration on mesh....2437
Mesh call admission control....2438
Configure mesh CAC (CLI)....2438
Fast teardown for a mesh deployment....2439
Configure fast teardown for a mesh AP profile (GUI)....2439
Configure fast teardown for a mesh AP profile (CLI)....2440
Fast teardown verification with default mesh profile....2441

194 Redundant Root Access Point (RAP) Ethernet Daisy Chaining.....2443
Redundant RAP Ethernet daisy chaining.....2444
Prerequisites for redundant RAP Ethernet daisy chaining support.....2444
Configure redundant RAP Ethernet daisy chaining support (CLI).....2444
Verify daisy chain redundancy.....2445

Part XVII: URWB.....2447

195 URWB.....2449
Ultra-Reliable Wireless Backhaul Background.....2450
Topologies.....2450
Benefits.....2450
Limitations and restrictions.....2451
Configure URWB using the controller.....2451
Create URWB network profile (GUI).....2451
Configure URWB network profile (CLI).....2453
Limitations and guidelines for secondary Ethernet ports 2460
Configure secondary Ethernet port (GUI) 2461
Configure secondary Ethernet port (CLI).....2462
Create URWB radio profile (GUI).....2463
Create URWB radio profile (CLI).....2465
Configure URWB parameters in the privileged EXEC mode.....2471
Verify URWB details.....2473
URWB telemetry protocol.....2474
Configure URWB telemetry protocol using AP CLI.....2474
Monitor URWB network information using Controller (GUI).....2476
Verify AP URWB information 2477
URWB recovery 2478
Prerequisites.....2478
Configure the URWB recovery using AP CLI.....2478
Country code behavior on AP models.....2480
URWB recovery configuration example.....2481
Dynamically configure URWB parameters.....2482
Hot-configurable URWB parameters.....2482
Parameters requiring a device reload.....2483
Configure URWB parameters using CLI.....2483
Configure URWB parameters using GUI.....2484
View configuration update summary.....2485
Controller-directed URWB traffic.....2485
URWB clients.....2485
URWB traffic egress point.....2486
URWB client WLANs.....2486

URWB mobility across subnets....2486
Limitations of Controller-directed URWB traffic....2487
How URWB client mobility works....2487
Create VLAN for vehicle AP and wired client (GUI)....2487
Create VLAN for vehicle AP and wired client....2488
Create URWB profile for mobility role base and client (GUI)....2489
Create URWB profile for mobility role base and client....2491
Create policy tag profile (GUI)....2493
Create policy tag profile....2494
Bind policy tag to coordinator APs (GUI)....2495
Bind the policy tag to coordinator APs....2495
Configure VLAN tagging for vehicle AP (GUI)....2496
Configure VLAN tagging for the vehicle AP....2497
View the client type information....2497
Support for 6 GHz Low Power Indoor mode....2498
Restrictions....2499
How 6 GHz mobility clients operate....2499
Configure the 6-GHz LPI mode....2500
Verify 6 GHz Low Power Indoor on AP in mobility client role....2505
NAT/PAT for URWB....2506
Guidelines for NAT/PAT configuration....2507
Configure NAT/PAT for URWB (GUI)....2507
Configure NAT/PAT for URWB....2508
Digital I/O ports....2509
Control digital I/O ports....2510
Digital I/O device mapping....2511
Power saving modes....2512
Manage power saving modes....2512
IOx frameworks....2513
IOx resource specifications for IW9165E....2513
Enable IOx using controller....2514
Enable IOx using AP....2514
URWB topology awareness....2515
Configure URWB topology awareness (CLI)....2515
Configure URWB topology awareness (GUI)....2516
Verify URWB network information on the coordinator AP....2517
Ethernet port authentication....2518
Guidelines and limitations....2519
Configure a RADIUS server and group....2519
Enable 802.1x authentication on Ethernet ports....2521
Troubleshoot RADIUS authentication with debug commands....2522
Verify RADIUS authentication status on an AP....2523

Part XVIII: VideoStream....2525

196 VideoStream....2527
Media stream....2528
Prerequisites for media stream....2528
How to configure media stream....2528
Create and attach policy-based QoS profile....2542
Monitor media streams....2549

Part XIX: Software-Defined Access Wireless....2553

197 Software-Defined Access Wireless....2555
Software-defined access wireless....2556
SD-Access wireless mesh inter fabric edge switch roaming protection....2559
Configure SD-access wireless....2561
Configure default map server (GUI)....2561
Configure default map server (CLI)....2562
Configure SD-Access Wireless Profile (GUI)....2563
Configure SD-Access wireless profile (CLI)....2563
Configuring Map Server in Site Tag (GUI)....2564
Configure map server in site tag (CLI)....2565
Configure map server per L2-VNID using GUI....2565
Configure map server per L2-VNID (CLI)....2566
SD-Access wireless verification commands....2567

198 Passive Client....2569
Passive clients....2570
Enable passive client for FlexConnect mode (CLI)....2571
DHCP broadcast support for workgroup bridges....2572
Enable passive client on WLAN policy profile (GUI)....2572
Enable passive client on WLAN policy profile (CLI)....2573
Enable ARP broadcast on VLAN (GUI)....2573
Enable ARP broadcast on VLAN (CLI)....2574
Passive client in fabric deployment....2575
Enable broadcast underlay on VLAN....2575
Enable ARP flooding....2577
Verify passive client configuration....2580

199 Fabric in a Box with External Fabric Edge....2583
Fabric in a Box with external Fabric Edge....2584
Configure a Fabric profile (CLI)....2584
Configure a policy profile (CLI)....2585
Configure a site tag (CLI)....2586

Configure a WLAN (CLI)....2587
Configure a policy tag (CLI)....2587
Configure an AP profile (CLI)....2588
Configure map server and AP subnet (CLI)....2589
Configure Fabric on FiaB node (CLI)....2590
Configure a Fabric Edge Node (CLI)....2594
Verify Fabric configuration....2599

Part XX: VLAN....2605

200 VLANs....2607
Virtual Local Area Network....2608
Supported VLANs....2608
VLAN port membership modes....2608
VLAN configuration files....2609
Normal-range VLAN configuration guidelines....2610
Extended-range VLAN configuration guidelines....2610
Prerequisites for VLANs....2610
Restrictions for VLANs....2611
Configuring normal-range VLANs....2612
Create or modify an Ethernet VLAN (CLI)....2612
Assign static-access ports to a VLAN (GUI)....2613
Assign static-access ports to a VLAN (CLI)....2613
Extended-range VLANs....2614
Create an extended-range VLAN (GUI)....2615
Create an extended-range VLAN (CLI)....2615
Monitor VLANs....2616

201 VLAN Groups....2617
VLAN groups....2618
Prerequisites for VLAN groups....2618
Restrictions for VLAN groups....2618
Create a VLAN Group (GUI)....2619
Create a VLAN Group (CLI)....2619
Add a VLAN group to policy profile (GUI)....2620
Add a VLAN group to a policy profile (CLI)....2620
View the VLANs in a VLAN group....2621
VLAN group support for DHCP and static IP clients....2621
Prerequisites and restrictions of VLAN Group support for DHCP and static IP clients....2622
Supported features....2622

Part XXI: WLAN....2625

202 WLANs.....2627
WLANs.....2628
Prerequisites for WLANs.....2629
Restrictions for WLANs.....2629
Band selection.....2630
Off-channel scanning deferral.....2630
DTIM period.....2631
WLAN radio policies.....2632
Restrictions for WLAN radio policy.....2632
Prerequisites for configuring Cisco client extensions.....2632
Peer-to-peer blocking.....2633
Diagnostic channels.....2633
AP names in beacon.....2633
How to configure WLANs.....2634
WLAN wizard.....2634
Create WLANs (GUI).....2646
Create WLANs (CLI).....2647
Delete WLANs (GUI).....2648
Delete WLANs (CLI).....2648
WLAN search methods (CLI).....2649
Enable WLANs (GUI).....2650
Enable WLANs (CLI).....2650
Disable WLANs (GUI).....2651
Disable WLANs (CLI).....2651
Configure general WLAN properties (CLI).....2652
Configure advanced WLAN properties (CLI).....2654
Configure advanced WLAN properties (GUI).....2656
Configure WLAN radio policy (GUI).....2658
Configure a WLAN radio policy (CLI).....2659
Configure AP name advertisement in beacons (GUI).....2661
Configure AP name advertisement in beacons.....2661
WLAN properties verification (CLI).....2662
Verify WLAN-VLAN information of an AP.....2663
WLAN radio policy verification.....2663
Verify AP name advertisement.....2664

203 WLAN Security.....2665
WPA1 and WPA2 security protocols.....2666
Restrictions for WPA2 and WPA3.....2667
Prerequisites for Layer 2 Security.....2667
AAA override.....2667
Configure AAA override (CLI).....2667
VLAN override.....2668

Configure override VLAN for central switching (CLI)....2669
Configure override VLAN for local switching (CLI)....2670
VLAN override on Layer 3 web authentication....2671
VLAN override verification on Layer 3 web authentication....2672
Fallback for AAA-overridden VLAN....2672
Feature history for fallback for AAA-overridden VLAN....2672
Central switching and FlexConnect mode scenarios....2673
Configure fallback for AAA-overridden VLAN (CLI)....2674
Verify fallback for AAA-overridden VLAN....2675
How to configure WLAN security....2675
Configure static WEP Layer 2 security parameters (GUI)....2675
Configure static WEP layer 2 security parameters (CLI)....2676
Configure WPA + WPA2 Layer 2 security parameters (GUI)....2678
Configure WPA + WPA2 layer 2 security parameters (CLI)....2678

204 Remote LANs....2683
Remote LANs....2684
Limitations of RLAN....2685
Configure remote LANs (RLANs)....2686
RLAN authentication fallback....2702
Configure RLAN authentication fallback (CLI)....2703
Modify 802.1X EAP timers for RLAN clients....2704
Verify RLAN authentication fallback....2705

205 RLAN External Module....2707
External module....2708
Requirement: Prerequisites for configuring external module....2708
Configure external module (GUI)....2708
Configure external module (CLI)....2708
External module verification....2709

206 802.11ax Per WLAN....2711
802.11ax mode per WLAN....2712
Configure 802.11ax mode per WLAN (GUI)....2712
Configure 802.11ax mode per WLAN (CLI)....2713
Verify 802.11ax mode per WLAN....2713

207 BSS Coloring....2715
BSS coloring....2716
BSS coloring....2716
OBSS-PD and spatial reuse....2717
Configure BSS color on AP (GUI)....2717
Configure BSS color in the privileged EXEC mode....2717

Configure BSS color globally....2718
Configure BSS color in the configuration mode....2718
Configure overlapping BSS packet detect (GUI)....2719
Configure OBSS-PD spatial reuse globally (CLI)....2719
Configure OBSS PD in an RF profile (GUI)....2720
Configure OBSS-PD spatial reuse in the RF profile mode (CLI)....2721
BSS color and OBSS-PD verification....2722

208 DHCP for WLANs....2725
Dynamic Host Configuration Protocol....2726
Internal DHCP servers....2726
External DHCP servers....2727
DHCP assignments....2727
DHCP option 82....2728
Restrictions for configuring DHCP for WLANs....2729
Guidelines for DHCP relay configuration....2729
Configure DHCP for WLANs....2730
Configure DHCP scopes (GUI)....2730
Configure DHCP scopes (CLI)....2731
Configure internal DHCP server....2732
Configure the internal DHCP server under client VLAN SVI (GUI)....2732
Configure the internal DHCP server under client VLAN SVI (CLI)....2733
Configure the internal DHCP server under a wireless policy profile (GUI)....2737
Configure the internal DHCP server under a wireless policy profile (CLI)....2738
Configure the internal DHCP server globally (GUI)....2742
Configure the internal DHCP server globally (CLI)....2742
Configure IP reservations in the internal DHCP server (CLI)....2746
Verify internal DHCP configuration....2747
Configure DHCP-Required for FlexConnect....2749
FlexConnect DHCP-Required....2749
Restrictions for FlexConnect DHCP-Required....2749
Configure FlexConnect DHCP-Required (GUI)....2750
Configure FlexConnect DHCP-required (CLI)....2751
Verify FlexConnect DHCP-required....2752

209 Aironet Extensions IE (CCX IE) 2753
Cisco Aironet Extensions Information Element....2754
Configure Aironet Extensions IE using GUI....2754
Configure Aironet Extensions IE (CLI)....2754
AP name addition verification....2755

210 Device Analytics....2757
Device analytics....2758

Restrictions for device analytics....2758
Configure device analytics (GUI)....2758
Configure device analytics (CLI)....2759
Verify device analytics....2761
Verify device analytics configuration....2761
Adaptive 802.11r....2763
Configure adaptive 802.11r (GUI)....2764
Adaptive 802.11r verification....2764

211 Device Classifier Dynamic XML Support....2767
Device classifier dynamic XML support....2768
Feature history for device classifier dynamic XML support....2771
Enable device classifier (CLI)....2772
Update dynamic XML file....2773
Verify TLV value....2774
Old classification cache clearing....2774

212 BSSID Counters....2777
BSSID counters....2778
Enable BSSID statistics and BSSID neighbor statistics (CLI)....2778
BSSID statistics verification on the controller....2779

213 Fastlane+....2783
Fastlane+....2784
Configure Fastlane+ on a WLAN (CLI)....2784
Configure Fastlane+ on a WLAN (GUI)....2785
Monitor Fastlane+ (GUI)....2785
Fastlane+ verification....2786

214 Workgroup Bridges....2789
Cisco Workgroup Bridges....2790
Restrictions for WGB....2792
Configure a WGB on a WLAN (CLI)....2793
Verify WGB status on the controller....2795
Configure AP as WGB....2795
Configure Cisco Wave 2 APs or 11AX APs in WGB or CAPWAP AP mode (CLI)....2795
Configure an SSID profile for Cisco Wave 2 and 11AX APs (CLI)....2796
Configure the authentication server (CLI)....2798
Configure dot1X credential (CLI)....2799
Configure an EAP profile (CLI)....2800
Configure manual-enrollment of a Trustpoint for WGB (CLI)....2801
Configure auto-enrollment of a Trustpoint for workgroup bridge (CLI)....2803
Configure manual certificate enrollment using TFTP server (CLI)....2805

Import the PKCS12 format certificates from the TFTP server (CLI)....2807
Configure radio interface for workgroup bridges (CLI)....2807
Configure workgroup bridge timeouts (CLI)....2811
Configure bridge forwarding for workgroup bridge (CLI)....2813
Simplified WGB configuration....2814
Configure multiple WGBs (CLI)....2815
Verify WGB configuration....2816

215 Peer-to-Peer Client Support....2819
Peer-to-peer client support....2820
Configure peer-to-peer client support....2820

216 Deny Wireless Client Session Establishment Using Calendar Profiles....2823
Denial of wireless client session establishment....2824
Guidelines and limitations....2825
Configure daily calendar profile (CLI)....2825
Configure weekly calendar profile (CLI)....2826
Configure monthly calendar profile (CLI)....2828
Map daily calendar profile to policy profile (CLI)....2829
Map weekly calendar profile to policy profile (CLI)....2831
Map monthly calendar profile to policy profile (CLI)....2832
Verify calendar profile configuration....2834
Verify policy profile configuration....2834

217 Ethernet over GRE....2837
Ethernet over GRE....2838
EoGRE configuration....2839
Create a tunnel gateway (CLI)....2840
Configure the tunnel gateway (GUI)....2841
Configure a tunnel domain (CLI)....2841
Configure tunnel domain (GUI)....2842
Configure EoGRE global parameters CLI)....2843
Configure EoGRE global parameters (GUI)....2844
Configure a tunnel profile (CLI)....2845
Configure the tunnel profile (GUI)....2847
Associate WLAN to a wireless policy profile (CLI)....2848
Attach a policy tag and a site tag to an AP....2849
Verify EoGRE tunnel configuration....2850

218 Wireless Guest Access....2859
Wireless guest access....2860
Foreign map....2862
Wireless Guest Access: Use Cases....2863

Guidelines for wireless guest access....2863
Load balancing among multiple guest controllers....2864
Configure mobility tunnel for guest access (GUI)....2864
Configure mobility tunnel for guest access (CLI)....2864
Configure guest access policy (GUI)....2865
Configure guest access policy (CLI)....2866
View guest access debug information (CLI)....2868
Verify wireless guest access enablement....2869
Configure guest access using different security methods....2869
Open authentication....2869
Local web authentication....2873
Central web authentication....2878
Configure 802.1x with local web authentication....2883
Configure local web authentication with PSK protocol (CLI)....2884
Central web authentication with PSK protocol....2886
Central web authentication with iPSK protocol....2887
Configure web authentication on MAC address bypass failure (GUI)....2888
Configure web authentication on MAC address bypass failure (CLI)....2889
Configure WLAN for web authentication on MAC authentication bypass failure with pre-shared key (CLI)....2891
Configure WLAN with web authentication for MAB failures using a pre-shared key (CLI)....2893
Configure WLAN for web authentication on MAB failure with SAE (CLI)....2895
Configure WLAN for web authentication on MAC authentication failure with Dot1x (CLI)....2896

219 Wired Guest Access....2899
Wired guest access....2900
Restrictions for wired guest access....2902
Configure access switch for wired guest client (CLI)....2902
Configure access switch for foreign controller (CLI)....2904
Configure foreign controller with open authentication (GUI)....2905
Configure foreign controller with open authentication (CLI)....2906
Configure foreign controller with local web authentication (GUI)....2908
Configure foreign controller with local WEB authentication (CLI)....2909
Configure anchor controller with open authentication (GUI)....2911
Configure anchor controller with open authentication (CLI)....2912
Configure anchor controller with local web authentication (GUI)....2914
Configure anchor controller with local web authentication (CLI)....2914
Configure session timeout for a profile policy (CLI)....2917
Configure global settings (GUI)....2918
Verify wired guest configurations....2918
Wired guest access use cases....2922

220 Data Rate Selection at SSID Level....2923
Legacy data rate selection at SSID level....2924

Feature history for legacy data rate selection at SSID level....2924
Guidelines for legacy data rate selection....2925
Configure the minimum data rate (GUI)....2925
Configure the minimum data rate (CLI)....2925
Verify RF profile and radio slot status....2927
Verify RF profile configuration....2927

221 User Defined Network....2929
User defined networks....2930
Restrictions for User Defined Network....2931
Configure a user defined network (CLI)....2931
Configure a user defined network (GUI)....2933
Verify user defined network configuration....2933

222 Hotspot 2.0....2937
Hotspot 2.0....2938
Open roaming....2940
Configure Hotspot 2.0....2942

223 Client Roaming Across Policy Profile....2975
Client roaming policy profile....2976
Configure client roaming across policy profile....2977
Verify client roaming across policy profiles....2978

224 Assisted Roaming....2983
802.11k neighbor list and assisted roaming....2984
Restrictions for assisted roaming....2984
Configure assisted roaming (GUI)....2984
Configure assisted roaming (CLI)....2985
Verify assisted roaming....2987
Configuration examples for assisted roaming....2987

225 802.11r BSS Fast Transition....2989
802.11r Fast Transition....2990
Restrictions for 802.11r Fast Transition....2991
Monitor 802.11r fast transition....2992
Configure 802.11r Fast Transition in an open WLAN (CLI)....2993
Configure 802.11r BSS Fast Transition on a Dot1x security enabled WLAN (CLI)....2995
Configure 802.11r Fast Transition on a PSK security-enabled WLAN (CLI)....2996
Disable 802.11r fast transition (GUI)....2998
Disable 802.11r Fast Transition (CLI)....2999
802.11r Fast Transition for SAE authenticated clients....3000

Feature history for 802.11r Fast Transition....3001
Configure 802.11r Fast Transition on a SAE security-enabled WLAN (GUI)....3001
Configure 802.11r Fast Transition on an SAE security-enabled WLAN (CLI)....3002
Verify 802.11r Fast Transition SAE....3004

226 802.11v....3007
802.11v amendment....3008
Prerequisites for configuring 802.11v....3009
Restrictions for 802.11v....3009
802.11v BSS transition management....3009
Configure 802.11v BSS transition management (GUI)....3009
Configure 802.11v BSS transition management (CLI)....3010

227 Virtual Routing and Forwarding....3013
VRF support....3014
Guidelines and restrictions for VRF support....3015
Create a VRF instance....3015
Map VRF to SVI....3016
Add VRF name through Option 82 for DHCP relay....3017
Add VRF name to DHCP server for DHCP relay....3018
VRF support verification....3020

228 Automated Frequency Coordination....3025
Automated frequency coordination....3026
Feature history for automated frequency coordination....3029
Prerequisites for Automated Frequency Coordination....3029
Restrictions for Automated Frequency Coordination....3030
Onboard the cloud controller (GUI)....3030
Configure DNA services (GUI)....3031
Configure power mode per RF profile (CLI)....3032
Configure power mode per RF profile (GUI)....3033
Configure AP parameters (GUI)....3034
Configure AP parameters (CLI)....3035
AFC details verification....3036
AP height configuration through priming profile....3041
Guidelines for AP height configuration through priming profile....3042
Configure AP height through priming profile (GUI)....3042
Configure AP height through priming profile (CLI)....3043
Apply priming profile using filters (GUI)....3044
Apply priming profile using filter (CLI)....3045
Apply priming profile statically (GUI)....3046
Apply priming profile statically (CLI)....3047
AP priming profile verification....3047

Part XXII: Cisco DNA Service for Bonjour.....3049

229 Cisco DNA Service for Bonjour Solution Overview.....3051

Cisco DNA Service for Bonjour Solution.....3052
Solution components.....3053
Supported platforms.....3054
Supported network designs.....3055

230 Configuring Local and Wide Area Bonjour Domains.....3065

Cisco DNA Service for Bonjour Solution Overview.....3066
Restrictions for Cisco DNA service for Bonjour solution.....3066
How Cisco Wide Area Bonjour services work.....3066
Cisco wide area Bonjour supported network design.....3067
Traditional wired and wireless networks.....3067
Cisco SD Access wired and wireless networks.....3068
Local and wide area Bonjour policies.....3069
Local area Bonjour policies.....3070
Wide Area Bonjour policies.....3073
Default mDNS service configurations.....3074
HSRP-aware mDNS service-routing.....3075
mDNS service-gateway SSO support.....3076
Configure local and wide area Bonjour domains.....3077
How to configure multicast DNS mode for LAN and wired networks.....3077
Enable mDNS gateway on the device.....3077
Create custom service definition (GUI).....3079
Create custom service definition.....3079
Create a service list (GUI).....3080
Create service list.....3081
Create service policy (GUI).....3082
Create service policy.....3083
Associate service policy to an interface.....3084
Local area Bonjour in multicast DNS mode for wireless networks.....3086
Associate service policy with wireless profile policy.....3088
Configure wide area Bonjour domain.....3089
Associate service policy with the controller in wide area Bonjour domain.....3089
Configure HSRP-aware mDNS service-routing on SDG.....3091
Configure HSRP-aware mDNS service-routing on service-peer.....3091
Verify local area bonjour in multicast DNS mode for LAN and wireless networks.....3093
Verify SDG-agent status.....3093
Verify wide area Bonjour controller status.....3094
Verify mDNS cache configurations.....3095
Verify additional mDNS cache configurations.....3097

Verify local area Bonjour configuration for LAN and wireless networks....3098
Additional references for DNA Service for Bonjour....3098
Feature history for Cisco DNA service for Bonjour....3099

231 Configuring Local Area Bonjour for Wireless Local Mode....3103
Local area Bonjour for wireless local mode....3104
Prerequisites for local area Bonjour for wireless local mode....3104
Restrictions for local area Bonjour for wireless local mode....3104
Local area Bonjour for wireless local mode....3104
Wireless AP multicast....3106
Configure wireless AP multicast (GUI)....3106
Configure wireless AP multicast (CLI)....3107
Configure multicast in IP network (CLI)....3108
Configure local area Bonjour for wireless local mode....3109
Configure mDNS service policy (GUI)....3109
Configure mDNS service policy using CLI....3110
Configuring custom service definition (GUI)....3113
Configure custom service definition (CLI)....3113
Configure mDNS gateway on WLAN (GUI)....3114
Configure mDNS gateway on WLAN (CLI)....3115
Configure service-routing on service-peer....3116
Configure location-based mDNS on service-peer (GUI)....3117
Configure location-based mDNS on service-peer (CLI)....3119
Verify mDNS gateway configuration....3121
Reference....3123

232 Configuring Local Area Bonjour for Wireless FlexConnect Mode....3125
Local area Bonjour for wireless FlexConnect mode....3126
Restrictions for local area Bonjour for wireless FlexConnect mode....3126
Prerequisites for local area Bonjour for wireless FlexConnect mode....3126
mDNS gateway alternatives for wireless FlexConnect mode....3126
Local area Bonjour for wireless FlexConnect mode....3127
Local area Bonjour for wireless FlexConnect mode....3129
Configure mDNS gateway mode (CLI)....3130
Configure mDNS service policy (CLI)....3131
Configure mDNS Location-Filter (CLI)....3135
Configure custom service definition (CLI)....3138
Configure service-routing on service-peer using CLI....3139
Location-based mDNS....3142
Configure service-routing on SDG agent (CLI)....3143
Verify local area Bonjour in service-peer mode....3145
Verify local area Bonjour in SDG agent mode....3147
Reference....3149

233 Configuration Example for Local Mode - Wireless and Wired.....3151
Overview.....3152
Configuring Wireless AP Multicast Mode.....3152
Configuration Example for Default Service List and Policy in Wide Area Bonjour Between Multilayer Wired and Wireless Endpoints.....3154
Configuration Example for Customized Service List and Policy in Wide Area Bonjour Between Multilayer Wired and Wireless Endpoints.....3156
Cisco Catalyst Center Traditional Multilayer Wired and Wireless Configuration.....3159
Verifying Wide Area Bonjour Between Multilayer Wired and Wireless Local Mode.....3161
Reference.....3168

234 Configuration Example for FlexConnect Mode - Wireless and Wired.....3169
Overview.....3170
Configuration Example for Default Service List and Policy in FlexConnect Mode - Wireless and Wired.....3171
Configuration Example for Customized Service List and Policy in FlexConnect Mode - Wireless and Wired.....3175
Verifying Configuration Example for FlexConnect Mode - Wireless and Wired.....3181
Reference.....3184

Part XXIII: Multicast Domain Name System.....3187

235 Multicast Domain Name System.....3189
mDNS Gateway.....3190
Guidelines and restrictions for configuring mDNS AP.....3190
Enable mDNS Gateway (GUI).....3192
Enable or disable mDNS Gateway (GUI).....3192
Enable or disable mDNS Gateway (CLI).....3193
Create default service policy.....3195
Create custom service definition (GUI).....3195
Create custom service definition (CLI).....3195
Create service list (GUI).....3197
Create service list (CLI).....3197
Create service policy (GUI).....3199
Create service policy (CLI).....3199
Configure a local or native profile for an mDNS policy (CLI).....3201
Configure an mDNS FlexConnect profile (GUI).....3202
Configure an mDNS FlexConnect Profile (CLI).....3202
Apply an mDNS FlexConnect profile to a wireless FlexConnect profile (GUI).....3203
Apply an mDNS FlexConnect profile to a Wireless FlexConnect profile (CLI).....3203
Enable the mDNS gateway on the VLAN interface (CLI).....3204
Configure mDNS AP (CLI).....3205

Enable mDNS Gateway on the RLAN interface (CLI)....3206
Enable mDNS Gateway on guest LAN interface (CLI)....3209
Associate mDNS service policy with wireless profile policy (GUI)....3210
Associate mDNS service policy with wireless profile policy (CLI)....3210
Enable or disable mDNS Gateway for WLAN (GUI)....3213
Enable or disable mDNS Gateway for WLAN (CLI)....3213
mDNS Gateway with guest anchor support and mDNS bridging....3214
Configure mDNS Gateway on guest anchor (CLI)....3215
Configure mDNS Gateway on guest foreign (Guest LAN) (CLI)....3215
Configure mDNS Gateway on guest anchor (CLI)....3216
Configure mDNS Gateway on guest foreign (Guest WLAN) (CLI)....3217
Verify mDNS Gateway configurations....3217
Location based service filtering....3224
Prerequisite for location-based service filtering....3224
Configure mDNS location-based filtering using SSID (CLI)....3224
Configure mDNS location-based filtering using AP name (CLI)....3225
Configure mDNS location-based filtering using AP location (CLI)....3226
Configure mDNS location-based filtering using regular expression (CLI)....3227
Nearest mDNS based wired service filtering....3233
Nearest mDNS-based wired service filtering....3233
Custom wired service policy support for FlexConnect mode....3236
VLAN and MAC based wired service filtering....3236
Use cases....3237
Configure wired service policy support in FlexConnect profile....3237

1 Overview of the Controller

Topics:

- Cisco 9800 Series Wireless Controllers
- Configuration data model
- Initial controller setup methods
- Interactive help
- Cisco Support Assistant
- Product analytics

Cisco 9800 Series Wireless Controllers

Introduces the Cisco Catalyst 9800 Series Wireless Controller, describing its architecture, key functionalities, and core benefits for next-generation wireless networks.

Cisco Catalyst 9800 Series Wireless Controllers are the next generation wireless controllers that

- are IOS XE based and integrate the Radio Frequency (RF) performance excellence from Aironet systems,
- supports intent-based networking that are easy to adapt and scale, and
- are available in multiple physical and virtual form factors for flexible management options.

Available form factors and deployment options

You can deploy the controllers as physical appliances or as virtual devices in private and public cloud environments.

The controllers cater to these deployment options:

- Cisco Catalyst 9800 Series Wireless Controller Appliance
- Cisco Catalyst 9800 Series Wireless Controller for Cloud
- Cisco Catalyst 9800 Embedded Wireless for Switch

Management options

You can managed them the Cisco Catalyst 9800 Series Wireless Controllers with:

- Cisco Catalyst Center
- NETCONF or YANG, or
- GUI or CLI.

Cisco Prime Infrastructure is also supported.

Configuration data model

Explains the configuration data model structure for the controller, detailing how configuration information is organized, managed, and implemented within the system.

A configuration data model is a framework that

- enables reusability of network settings and components
- supports simplified provisioning for scalable network management, and
- provides enhanced flexibility and modularization to adapt to changing business and IT requirements.

For more information, see Information About New Configuration Model.

Initial controller setup methods

Outlines methods for initial setup of the Cisco Catalyst 9800 Series Wireless Controller using both GUI and CLI, and provides instructions for deploying controllers in environments of various sizes to support AP connectivity and wireless services.

The initial configuration wizard in Cisco Catalyst 9800 Series Wireless Controller is a simplified, out-of-the-box installation and configuration interface for the controller. This section provides instructions to set up a controller to operate in a small, medium, or large network wireless environment, where access points can join and together as a simple solution provide various services, such as corporate employee or guest wireless access on the network.

Setting up the controller using GUI

To set up the controller using the GUI, see the `Configuring Wireless Controller section` in Cisco Catalyst 9800 Wireless Controller Series Web UI Deployment Guide.

Setting up the controller using CLI

To set up the controller using CLIs, see the `Performing the Initial Configuration on the Controller` section of your respective controller installation guides.

- Cisco Catalyst 9800-80 Wireless Controller Hardware Installation Guide
- Cisco Catalyst 9800-40 Wireless Controller Hardware Installation Guide
- Cisco Catalyst 9800-L Wireless Controller Hardware Installation Guide
- Cisco Catalyst 9800-CL Cloud Wireless Controller Installation Guide
- Cisco Catalyst CW9800M Wireless Controller Hardware Installation Guide
- Cisco Catalyst CW9800H1 and CW9800H2 Wireless Controllers Hardware Installation Guide

Restrictions for initial controller setup

Network Reset Commands for EWC Devices

Do not use the **erase startup-config** command if you intend to reset EWC, APs, and EWC-APs in the network to Day0 state. We recommend that you use the **wireless ewc-ap factory-reset** command.

This restriction applies when resetting an EWC device and all associated APs and EWC-APs in a wireless network to their factory default (Day0) state.

The **erase startup-config** command only removes the configuration from the local device and does not synchronize the reset across other devices in the network. In contrast, the **wireless ewc-ap factory-reset** command resets the EWC device and all connected APs and EWC-APs to Day0 state, ensuring network-wide consistency.

Setting up the controller Using GUI

- To identify remote users in the GUI, the remote authentication server (RADIUS or TACACS) must send the user-type attribute with value 6.
- If you make configuration changes in the CLI and in the GUI simultaneously, you must click the **Refresh** button in the GUI to synch both the changes. You should always click the **Refresh** button in the GUI to update the changes done through the CLI.
- When you go to the login page, the controller sends a request to the RADIUS server for the banner text. You can view this request on the RADIUS server.
- When you log in to the Cisco Catalyst 9800-L Wireless Controller GUI, you receive many CPU alerts because of the CPU utilization spikes during log in.
- Header size refers to the cookies sent from the browser through the controller GUI to the device WebSocket server in a WebSocket header. If the size of the cookie exceeds 4 KB, then any request that is made to the WebSocket server (query or configuration) via the GUI would fail. In such a case, ensure that you clear the browser cookies and cache, periodically.

  The banner text displayed on the browser is fetched from the controller. You can view this request on the RADIUS server.

Synchronize changes across CLI and GUI using Refresh

- If you make configuration changes in the CLI and in the GUI simultaneously, you must click the **Refresh** button in the GUI to synchronize the changes.

Interactive help

Details the interactive help features, guiding users through supported workflows and providing contextual assistance for configuring and managing the controller.

Interactive help is a user interface feature that

- provides step-by-step guidance within the application
- adapts instructions and walk-throughs to the user's context, and
- assists users in completing complex configurations or navigating the system.

Modes of starting the interactive Help

You can start the interactive help in the these ways.

- Hover over the blue flap at the right-hand corner of a window in the GUI and clicking **Interactive Help**.
- Click **Walk-me Thru** in the left pane of a window in the GUI.
- Click **Show me How** whenever displayed in the GUI. Clicking **Show me How** triggers a specific interactive help that is relevant to the context you are in.

  For example, **Show me How** in **Configure** > **AAA** walks you through the various steps for configuring a RADIUS server. Choose **Configuration> Wireless Setup** > **Advanced** and click **Show me How** to trigger the interactive help that walks you through the steps relating to various kinds of authentication.

Additional troubleshooting information

If the WalkMe launcher is unavailable on Safari, modify the browser settings.

1. Choose **Preferences** > **Privacy**.
2. In the **Website tracking** section, uncheck the **Prevent cross-site tracking** check box to disable this action.
3. In the **Cookies and website data** section, uncheck the **Block all cookies** check box to disable this action.

Cisco Support Assistant

Introduces the Cisco Support Assistant, explaining its integrated support capabilities and how it assists with troubleshooting, diagnostics, and streamlining case resolution.

Cisco Support Assistant (CSA) is an in-product support tool that

- provides built-in integration of Cisco Support Assistant Extension (CSAE) into the controller GUI without the need for installation of a Chrome browser extension (CSA Embed)
- integrates Cisco Technical Assistance Center (TAC) support functions directly into the GUI, and
- enables easy and quick access to Cisco support without leaving the GUI.

CSA Embed is the built-in integration of the Cisco Support Assistant Extension (CSAE) within the controller, that does not require installation of the Chrome browser extension.

Feature History

Feature Name	Release	Description
Cisco support assistant integration to WLC GUI	Cisco IOS XE 17.18.2	This feature integrates the Cisco Support Assistant (CSA) directly within the controller GUI. You can now open TAC cases, upload logs, record screens, and access contextual help directly from within the controller interface, and without installing the Chrome browser extension. You can find a feature demonstration here.

Supported features

1. **Case Open**: Provides an intuitive, context-aware interface for customers to open TAC cases directly from the GUI. You can attach relevant logs and files easily, ensuring faster case resolution with minimal manual steps.
2. **Screen Recording**: Allows your to record on-screen activities to demonstrate issues visually during case submission. These recordings help TAC engineers quickly understand the problem context, leading to more efficient troubleshooting.
3. **File Upload**: Simplifies the process of uploading diagnostic files and configuration logs to Cisco TAC. Secure file transfer is integrated directly within the GUI. This makes it easy to share your files while maintaining compliance and privacy standards.
4. **Smart Tooltips**: Provides contextual guidance throughout the GUI by displaying helpful hints, quick links to Cisco documentation, and troubleshooting insights. This increases your self-service capabilities and reduces your need to contact support.
5. **Wireless Compatibility Assistant**: Integrates the wireless compatibility assistant to the **Access Point Monitoring and Configuration** controller GUI window to enable self-service for wireless configuration, compatibility, best practices, and feature-related questions.

For a demonstration of the supported features, see here.

> **Note**
>
> Use CSA Embed in English for the best experience. Some CSA features may not work properly in local languages or on certain pages.

Enable CSA Embed

You can enable or disable the **CSA Embed** feature from the settings menu (**gear icon**) on the top right corner of the GUI by toggling the **Cisco Support Assistant** option.

Product analytics

Describes the product analytic capabilities, highlighting features that monitor controller operations, analyze performance trends, and support data-driven wireless network management.

Product analytics is a network device function that

- collects non-personal usage data from Cisco devices

- enables ongoing improvement of Cisco products and customer experience by analyzing that data, and
- can be configured or disabled by administrators using specific device commands.

Supported platforms

This feature is supported on the Cisco Catalyst 9800 Series Wireless Controllers (9800-80, 9800-40, 9800-L, 9800-CL, CW9800M, and CW9800H1/H2).

Management commands

You can use the the **pae** command to enable or disable this feature.To modify your organization's preferences for device and licensing systems information, use the **pae** command.

The commands are introduced as part of this feature.

- **pae**
- **show product-analytics kpi**
- **show product-analytics report**
- **show product-analytics stats**

For more information:

- Cisco Catalyst 9800 Series Wireless Controller Command Reference
- Wireless Product Analytics FAQ

Network connectivity requirements

When Product Analytics is enabled, the controller may send DNS queries to these public DNS resolver addresses, even if DNS servers are configured on the controller.

- IPv4: 208.67.222.222, 208.67.220.220, 8.8.8.8, 8.8.4.4
- IPv6: 2620:119:35::35, 2620:119:53::53, 2001:4860:4860::8888, 2001:4860:4860::8844

Privacy and data handling

Cisco collects device and licensing systems information through Cisco Smart Software Manager (CSSM) and other channels for these purposes.

- Analytics and reporting
- Product improvement
- Enhancing the customer experience

Cisco processes this data in accordance with

- Cisco Privacy Statement
- General Terms and Conditions, and
- any other applicable agreements with Cisco.

Note

Turning off Smart Licensing Device Systems Information does not impact other Systems Information collection including from Cisco Catalyst Center or vManage.

Part I

System Configuration

Topics:

- New Configuration Model
- Wireless Management Interface
- Configure Network Address Translation for Wireless Management IP
- Hardware Features
- Smart Licensing Using Policy
- Management over Wireless
- Boot Integrity Visibility
- SUDI99 Certificate Support
- Link Aggregation Group
- Reload Reason History

System Configuration

2 New Configuration Model

Topics:

- New configuration model, tags, and profile
- Tags and their associated profiles
- Configure a wireless profile policy (GUI)
- Configure a site tag (GUI)
- Enhanced site tag-based load balancing
- Wireless radio profiles
- AP filters
- Location configuration

Introduces the tag-based configuration model for wireless networks and guides through creating and associating various profile types including wireless, flex, access point, and radio frequency profiles with network infrastructure.

New configuration model, tags, and profile

Introduces the foundational elements of the tag-based configuration model and outlines the complete configuration workflow for implementing wireless network infrastructure.

The new configuration model is a wireless network configuration approach that

- uses profiles to define feature-specific parameters
- uses tags to bundle and apply profiles to APs, and
- separates policy, site, RF, and WLAN definitions for modular deployment.

Profiles define attributes such as WLAN policy, RF behavior, and AP join characteristics. Tags are logical containers that map these profiles to APs, ensuring flexible and scalable configurations.

A tag is a logical container that

- is defined by the property of policies you associate with it
- maps multiple profiles—policy, site, and RF—to an AP or client, and determines that device's configuration and behavior, and
- enables modular and flexible deployment across sites.

Every tag has a default that is created when the system boots up.

A profile is a logical container that

- include feature-specific attributes and parameters applied to tags
- represent multiple attributes related to policy, site, and RF
- determine the configuration and behavior of applied APs and associated clients, and
- is a reusable entities that can be used across tags.

Configuration workflow

Configuring wireless networks involves establishing profiles and tags that link configurations to APs for effective wireless management.

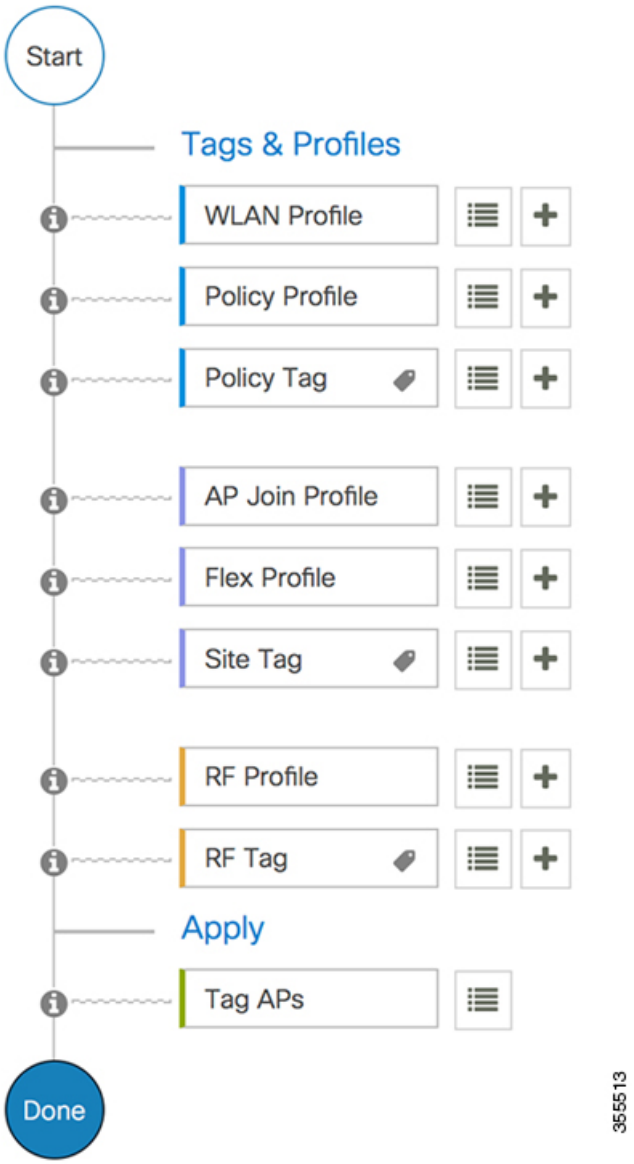
Summary

The key components involved in the process are:

- **Profiles**: Logical groupings of settings (such as WLAN, Policy, AP Join, Flex, and RF) used to configure APs.
- **Tags**: Logical containers or labels (such as Policy, Site, Tag) that help link profiles with APs.
- **APs**: Devices that implement the configurations provided by profiles and tags to deliver wireless connectivity.

Workflow

Figure 1: Configuration workflow



These are the stages of configuration:

1. **Profile creation** creates specific profiles:
   - WLAN profile for managing wireless network settings.
   - Policy profile for enforcing network rules.
   - AP Join profile for controlling AP associations.
   - Flex profile for handling local switching.
   - RF profile for optimizing radio frequency settings.
2. **Tag creation**: generates tags to complement profiles:
   - Policy tag to align with the Policy profile.
   - Site tag for specific location identification.
   - RF tag corresponding to the RF profile.

3. **Tag association**: associates the created tags with an AP to apply the configured settings and policies.

Result

The configuration workflow enables APs to be set up correctly with designated profiles and tags, resulting in efficient network operation and management.

Tags and their associated profiles

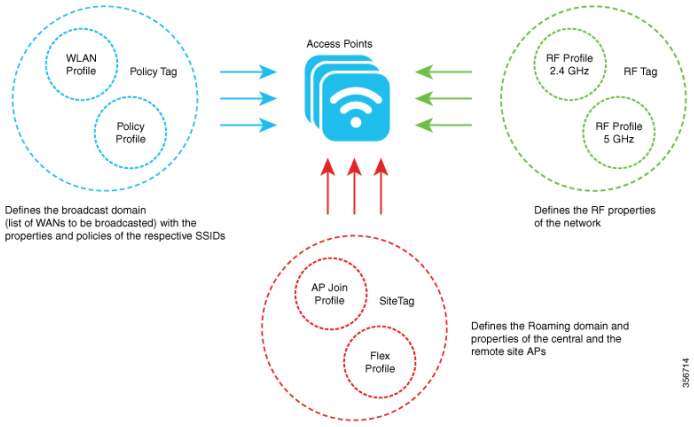
Details RF tags, RF profiles, site tags, flex profiles, AP join profiles, policy tags, WLAN profiles, policy profiles, and methods for associating APs with tags.

The new configuration model has different types of tags that contain various types of profiles, depending on the functionality they represent.

Table 1: Various types of tags and the contained profiles

Tag	Contains	Function
RF Tag	Radio profile	RF tags are used to group and manage radio frequency profiles for wireless networks.
Site Tag	Flex profile and AP join profile	Site tags define the properties of a site and maps the flex profile to the the AP join profile.
Policy Tag	WLAN profile and policy profile	Policy tags map WLAN profile to the policy profile

The figure depicts the various types of tags and their contained profiles. The figure also depicts how the tag is thus applied to access points easily.



RF Tags

RF Tag fundamentals and configuration, detailing their role in wireless settings and capabilities for network optimization.

An RF tag is a configuration object that

- contains profiles for different radio frequency bands
- are used to group and manage radio frequency profiles for wireless networks

- defines global or band-specific parameters for wireless devices, and
- provides default settings for each RF profile to ensure consistent operation across radios.

The RF tag contains the 2.4 GHz, 5 GHz, and 6 GHz RF profiles. The default RF tag contains the global configuration for 2.4 and 5 GHz bands and default RF profile for 6 GHz band. All these profiles contain the same default values for global or RF profile parameters for the respective radios.

Example of RF tag

For a device operating in multiple frequency bands, applying an RF tag ensures that each radio (such as 2.4 GHz or 5 GHz) uses the network-approved configuration for coverage, performance, and reliability.

RF Profiles

Introduces RF Profiles, highlighting their significance in managing radio parameters and optimizing wireless network performance.

An RF profile is a configuration set that

- centralizes radio settings for APs
- applies a uniform set of radio parameters to all APs within an AP group, and
- ensures consistent wireless performance and feature deployment across the group.

Some RF profile options are specific to the 6-GHz band and include features such as Unsolicited Broadcast Probe Response, FILS Discovery, Multi-BSSID, and Preferred Scanning Channels. These features address 802.11ax requirements and help optimize management traffic or channel selection for reliable wireless performance..

Preferred Scanning Channels is a feature that helps RRM choose PSC channels to 6-GHz radios.

Site Tags

Describes Site Tags and their application in logically grouping access points for streamlined wireless management and policy enforcement.

A site tag is a configuration object that

- defines the properties of a wireless site
- maps a flex profile to an AP join profile, and
- specifies site-specific settings, such as a list of primary APs for upgrades.

Additional reference information

- Attributes unique to a flex or remote site are part of the flex profile, while attributes that belong to the actual physical site—like the primary APs list—are directly included in the site tag.
- The site tag includes settings that are not suited for shared, reusable profiles.
- If the flex profile name or AP profile name is changed within the site tag, associated APs are required to rejoin the controller by disconnecting their Datagram Transport Layer Security (DTLS) session.
- When a site tag is created, the AP and flex profiles default to preset values (default-ap-profile and default-flex-profile).

Example of site tag

Assigning a set of primary APs for efficient software upgrades is done through the site tag, not the flex profile, since this information applies specifically to the physical site.

Flex profiles

A Flex profile is a configuration profile that

- contains policy attributes for network management
- specifies remote site-specific parameters, and
- supports custom mappings such as VLAN-to-ACL and VLAN name-to-ID assignments.

The FlexConnect configuration helps the central controller to manage sites that are geo-distributed, for example, retail, campus, and so on.

Example of flex profile

A Flex profile can include EAP profiles used when an AP acts as an authentication server for local RADIUS information, as well as mappings for VLANs and associated ACLs or VLAN names and IDs

Analogy: A Flex profile is like a master checklist for a remote office: just as a checklist ensures every necessary step, tool, and contact is available for smooth operation at the branch, a Flex profile gathers all the required policies and site-specific settings so each access point at a remote site works according to the organization's requirements without repeated manual configuration.

AP join profiles

An AP join profile is a configuration set that

- specifies global and AP group-specific parameters for APs
- defines network and communication settings specific to an AP, such as CAPWAP, IPv4, IPv6, and protocol configurations, and
- centralizes control of AP-specfic settings—including retransmit configuration, UDP Lite, high availability, Global AP failover, Hyperlocation config parameters, Telnet and SSH, and 11u parameters—to streamline access point management across a network.

Policy Tag

A policy tag is a network configuration object that

- maps each WLAN profile to a specific policy profile
- determines how network and switching policies are assigned to wireless clients, and
- controls the deployment of WLAN and policy profiles to access points (APs) based on their enabled state.

Policy tag mapping

A policy tag maps the WLAN profile to the policy profile.

Network Configuration Object	Function
WLAN profile	defines the wireless characteristics of the WLAN.
Policy profile	defines the network policies and the switching policies for the client (Quality of Service [QoS] is an exception, which constitutes AP policies as well).

Network Configuration Object	Function
Policy Tag	maps the WLAN profile to the policy profile. There are 16 such WLAN-to-policy profile pairs. The policy tag directs how wireless services and policies are applied across the network.

Note

Quality of Service (QoS) policies are an exception, as they are managed as part of AP policies instead of policy profiles.

Example of policy tag

If a policy tag includes WLAN1 mapped to Policy1, and both profiles are enabled, their definitions are pushed to APs assigned that policy tag. If either the WLAN profile or the policy profile is disabled, that mapping is not pushed to the AP. You can also remove a WLAN profile from an AP by deleting its mapping in the policy tag configuration.

WLAN profiles

A WLAN profile is a configuration entity that

- defines the wireless network by specifying the service set identifier (SSID)
- associates Layer 2 security policies with WLANs, and
- groups related settings required for controllers to manage wireless local area networks.

Create WLANs with the same SSID you to assign different Layer 2 security policies within a single wireless LAN.

Distinguish WLANs that use the same SSID by assigning a unique WLAN profile name to each. Each WLAN with a shared SSID must have a unique Layer 2 security policy so that clients can select the appropriate WLAN based on the security information advertised in beacon and probe responses.

Note

Switching and network policies are not included in the WLAN profile definition; they are configured separately.

Analogy: A WLAN profile is like a membership card for a club. The card (profile) shows which club you belong to (SSID), lists the rules you must follow to enter (Layer 2 security policies), and includes the necessary information for the club staff (controller) to manage your membership. Just as two people can have cards for the same club but with different access levels, multiple WLAN profiles can share the same SSID but have different security policies to distinguish them.

Policy profiles

A policy profile is a network configuration entity that

- allows you to group and manage multiple policies
- is reusable across tags and deployments, and applied to APs or controllers for client access, and

- improves consistency and efficiency in large-scale wireless environments.

Example of policy profile

Policy profile centralizes policy parameters such as VLAN, access control list (ACL), Quality of Service (QoS), session timeout, idle timeout, AVC profile, Bonjour profile, local profiling, device classification, and BSSID QoS.

Table 2: Comparison between policy profiles and WLAN profiles

Policy profiles	WLAN profiles
Contain network and switching policies	Contain wireless-related security attributes, and features such as authentication and encryption
Reusable across tags and deployments, and applied to APs or controllers for client access	Configured per WLAN (SSID)

Table 3: Comparison between policy profiles and WLAN profiles

Policy profiles	WLAN profiles
Contain network and switching policies	Contain wireless-related security attributes, and features such as authentication and encryption
Reusable across tags and deployments, and applied to APs or controllers for client access	Configured per WLAN (SSID)

Analogy: A policy profile is like a preset rulebook for network access, grouping the common rules that apply to all players (clients), while the WLAN profile acts like a gatekeeper, focusing on who can enter and under what security conditions.

Table 4: Comparison between policy profiles and WLAN profiles

Policy profiles	WLAN profiles
Contain network and switching policies	Contain wireless-related security attributes, and features such as authentication and encryption
Reusable across tags and deployments, and applied to APs or controllers for client access	Configured per WLAN (SSID)

Methods for associating APs with tags

APs can be associated with tags in several ways, supporting flexible and scalable network configuration. The main association methods are:

- **Ethernet MAC address association**: The default option where an AP's Ethernet MAC address is directly mapped to a policy-tag, site tag, and RF tag.
- **Filter-based association**: Uses regular expressions (regex) to match AP Ethernet MAC addresses. Any AP matching the pattern receives the assigned tags (policy-tag, site tag, and RF tag) configured through an AP filter.
- **AP-based association**: Tag names are pre-configured at the Plug and Play (PnP) server. The AP stores these and submits the tag name during the discovery process.
- **Location-based association**: Tags are mapped to specific locations. Any AP mapped to a location receives the corresponding tags.

Note

Before downgrading from Cisco IOS XE 26.1.1 or later to Cisco IOS XE 17.18.x or earlier, explicitly assign the required policy, site, and RF tags to each access point that has a statically configured name. Otherwise, an AP whose tags are assigned through an AP filter or location configuration receives the default tags after the downgrade.

Effect of AP tag modification

Modifying an AP tag results in DTLS connection reset, forcing the AP to rejoin the controller. If only one tag is specified in the configuration, default tags are used for other types, for example, if only policy tag is specified, the default-site-tag and default-rf-tag will be used for site tag and RF tag.

Modifying an AP tag has these effects:

- It resets the DTLS connection, forcing the AP to rejoin the controller.
- If only one tag is specified (for example, only a policy tag), the system assigns default tags for the other types:
  - The default-site-tag is used for the site tag.
  - The default-rf-tag is used for the RF tag.

Configure a wireless profile policy (GUI)

Provides comprehensive configuration procedures for wireless profile policies, flex profiles, AP profiles, user management, password encryption, and RF profiles through GUI and CLI interfaces.

Define and apply a wireless profile policy to manage network behavior and access for wireless clients.

Use this task to create or modify a policy profile, ensuring wireless network policies suit your organization's needs.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. On the **Policy Profile** page, click **Add**.
3. In the **Add Policy Profile** window, in **General** tab, enter a name and description for the policy profile. The name can be ASCII characters from 32 to 126, without leading and trailing spaces. Do not use spaces as it causes system instability.
4. To enable the policy profile, set **Status** as **Enabled**.
5. Use the slider to enable or disable **Passive Client** and **Encrypted Traffic Analytics**.
6. In the **CTS Policy** section, choose the appropriate status for the following:
   - Inline Tagging–a transport mechanism using which a controller or access point understands the source SGT.
   - SGACL Enforcement
7. Specify a default **SGT**. The valid range is from 2 to 65519.
8. In the WLAN Switching Policy section, choose the following, as required:
   - Central Switching: Tunnels both the wireless user traffic and all control traffic via CAPWAP to the centralized controller where the user traffic is mapped to a dynamic interface/VLAN on the controller. This is the normal CAPWAP mode of operation.

- Central Authentication: Tunnels client data to the controller, as the controller handles client authentication.
- Central DHCP: The DHCP packets received from AP are centrally switched to the controller and then forwarded to the corresponding VLAN based on the AP and the SSID.
- Central Association Enable: When central association is enabled, all switching is done on the controller.
- Flex NAT/PAT: Enables Network Address Translation(NAT) and Port Address Translation (PAT) mode.

9. Click **Save & Apply to Device**.

The wireless profile policy is successfully created and applied. Devices use the updated policy settings for wireless client management.

Configure a wireless policy profile (CLI)

Configure a Policy Tag using commands.

Define and apply a wireless policy profile on your device using command-line interface commands.

Use this task to configure wireless profile policies on Cisco wireless controllers. Policy profiles specify settings such as VLAN mapping, idle timeouts, and accounting lists for wireless networks.

> **Note**
>
> When a client moves from an old controller to a new controller (managed by Cisco Prime Infrastructure), the old IP address of the client is retained, if the IP address is learned by ARP or data gleaning. To avoid this scenario, ensure that you enable **ipv4 dhcp required** command in the policy profile. Otherwise, the IP address gets refreshed only after a period of 24 hours.

Follow the procedure given to configure a wireless profile policy:

Before you begin

- Ensure you have administrator privileges to access and configure the device.
- Have the required VLAN ID, idle timeout value, and accounting list details available, if applicable.

Follow these steps to configure a wireless profile policy:

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile and enters wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy rr-xyz-policy-1
   ```

3. Optional: Configure the duration of idle timeout in seconds.

   Example

   ```
   Device(config-wireless-policy)# idle-timeout 1000
   ```

4. Configure the VLAN name or VLAN ID.

   Example

   ```
   Device(config-wireless-policy)# vlan 24
   ```

5. Set the accounting list for IEEE 802.1x.

   Example

   ```
   Device(config-wireless-policy)# accounting-list user1-list
   ```

6. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

7. Optional: View detailed information about a policy profile, using the **show wireless profile policy detailed** *policy-profile-name* command.

   Example

   ```
   Device# show wireless profile policy summary
   ```

The wireless policy profile is configured and enabled on the device. All specified settings are applied to the selected policy.

What to do next

- Associate the policy profile with a WLAN as needed.
- Review and validate the applied settings by connecting a client device and confirming expected behavior.

Configure a flex profile (GUI)

Configure a Flex Profile using GUI.

Create or modify a flex profile to customize wireless network behavior for specific sites or device groups.

Use flex profiles within your network management system to define site-specific configurations such as VLANs, SSIDs, or access policies.

Before you begin

Identify the devices or sites to apply the flex profile.

Procedure

1. Choose **Configuration > Tags & Profiles** > **Flex**.
2. Click **Add**.
3. Enter the **Name** of the flex profile. Use ASCII characters from 32 to 126. Do not include leading and trailing spaces.
4. In the **Description** field, enter a description for the flex profile.
5. Click **Apply to Device**.

The flex profile is created or updated and available for assignment to devices or sites.

What to do next

Assign the flex profile to the relevant site, device, or group as needed.

Configure a flex profile

Configure a flex profile using commands.

Create or modify a flex profile to customize wireless network behavior for specific sites or device groups.

Use flex profiles within your network management system to define site-specific configurations such as VLANs, SSIDs, or access policies.

Before you begin

Identify the devices or sites to apply the flex profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a flex profile and enter flex profile configuration mode.

   Example

   ```
   Device(config)# wireless profile flex rr-xyz-flex-profile
   ```

3. Optional: Enable default parameters for the flex profile.

   Example

   ```
   Device(config-wireless-flex-profile)# description xyz-default-flex-profile
   ```

4. Optional: Enable ARP caching.

   Example

   ```
   Device(config-wireless-flex-profile)# arp-caching
   ```

5. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

Example

```
Device(config-wireless-flex-profile)# end
```

6. Optional: View detailed parameters about the flex profile, use the **show wireless profile flex detailed** *flex-profile-name* command.

Example

```
Device# show wireless profile flex summary
```

The flex profile is created or updated and available for assignment to devices or sites.

What to do next

Assign the flex profile to the relevant site, device, or group as needed.

Configure an AP profile (GUI)

Configure an AP Profile using GUI.

Configure and customize AP join profiles for your wireless deployment.

Use this task to define, modify, or apply AP profile parameters such as country code, LED state, timers, VLAN tagging, security settings, management options, and advanced features, using the device's graphical interface.

Before you begin

- Review the default AP join profile to update parameters for your environment (For example, Control and Provisioning of Wireless Access Points (CAPWAP), IPv4 or IPv6, UDP Lite, High Availability, retransmit configuration parameters, global AP failover, Hyperlocation configuration parameters, Telnet or SSH, 11u parameters, and so on.)
- Obtain required information, such as network-specific settings, controller addresses, credentials, and profile names.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. On the **AP Join Profile** window, click **Add**.

   The **Add AP Join Profile** window is displayed.

   > **Note**
   >
   > DHCP fallback is enabled by default. If an AP is assigned a static IP address and unable to reach the controller, the AP falls back to the DHCP. To prevent an AP from switching the static IP to DHCP, you must disable the DHCP fallback configuration in an AP join profile.

3. In the **General** tab, enter a name and description for the AP join profile. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.

4. Check the **LED State** check box to set the LED state of all APs connected to the device to blink, making them easier to locate. The LED state is enabled by default.
5. In the **Client** tab and **Statistics Timer** section, enter the time in seconds that the AP sends its 802.11 statistics to the controller.
6. In the **TCP MSS Configuration** section, check the **Adjust MSS Enable** check box to enter value for Adjust MSS. You can enter or update the maximum segment size (MSS) for transient packets that traverse a router. TCP MSS adjustment enables the configuration of the maximum segment size (MSS) for transient packets that traverse a router, specifically TCP segments with the SYN bit set.

   In a CAPWAP environment, a lightweight AP discovers a device by using CAPWAP discovery mechanisms, and then sends a CAPWAP join request to the device. The device sends a CAPWAP join response to the AP that allows the AP to join the device.

   When the AP joins the device, the device manages its configuration, firmware, control transactions, and data transactions.

7. In the **CAPWAP** tab, you can configure these options:

   - High Availability

     You can configure primary and secondary backup controllers for all APs (which are used if primary, secondary, or tertiary controllers are not responsive) in this order: primary, secondary, tertiary , primary backup, and secondary backup.In addition, you can configure various timers, including heartbeat timers and discovery request timers. To reduce the controller failure detection time, you can configure the fast heartbeat interval (between the controller and the AP) with a smaller timeout value. When the fast heartbeat timer expires (at every heartbeat interval), the AP determines if any data packets have been received from the controller within the last interval. If no packets have been received, the AP sends a fast echo request to the controller.

   a) In the **High Availability** tab, enter the time (in seconds) in the **Fast Heartbeat Timeout** field to configure the heartbeat timer for all APs. Specifying a small heartbeat interval reduces the amount of time it takes to detect device failure.

   > **Note**
   >
   > Configure **Fast Heartbeat Timeout** to assist AP in sending primary discovery request periodically to the configured backup controllers along with the primary, secondary, and tertiary-base controllers.

   b) In the **Heartbeat Timeout** field, enter the time in seconds to configure the heartbeat timer for all APs. Specifying a small heartbeat interval reduces the amount of time it takes to detect device failure.
   c) In the **Discovery Timeout** field, enter a value between one and 10 seconds (inclusive) to configure the AP discovery request timer.
   d) In the **Primary Discovery Timeout** field, enter a value between 30 and 3000 seconds (inclusive) to configure the AP primary discovery request timer.
   e) In the **Primed Join Timeout** field, enter a value between 120 and 43200 seconds (inclusive) to configure the AP primed join timeout.
   f) In the **Retransmit Timers Count** field, enter the number of times that you want the AP to retransmit the request to the device and vice versa. Valid range is between three and eight.
   g) In the **Retransmit TimersInterval** field, enter the time duration between retransmission of requests. Valid range is between two and five.
   h) Check the **Enable Fallback** check box to enable fallback.
   i) Enter the **Primary Controller** name and IP address.
   j) Enter the **Secondary Controller** name and IP address.
   k) Click **Save & Apply to Device**.

> **Note**
>
> The primary and secondary settings in the AP join profile are not used for AP fallback. This means that the AP will not actively probe for those controllers (which are a part of the AP join profile), when it has joined one of them.
>
> This setting is used only when the AP loses its connection with the controller, and then prioritizes which other controller it should join. These controllers have a priority of four and five, following APs in the **High Availability** tab of the AP page.
>
> The APs that are added as the primary, secondary, and tertiary APs in the **High Availability** tab of the AP configuration page, are actively probed and are used for the AP fallback option.

- Advanced

a) In the **Advanced** tab, check the **Enable VLAN Tagging** check box to enable VLAN tagging.
b) Check the **Enable Data Encryption** check box to enable Datagram Transport Layer Security (DTLS) data encryption.
c) Check the **Enable Jumbo MTU** to enable large maximum transmission unit (MTU). The MTU is the largest physical packet size, measured in bytes, that a network can transmit. Any messages larger than the MTU are divided into smaller packets before transmission. Jumbo frames exceed the standard Ethernet frame size, which is 1518 bytes (including Layer 2 (L2) header and FCS). Because vendors may have different frame size definitions, jumbo frames are not standardized by IEEE.
d) Use the **Link Latency** drop-down list to select the link latency. Link latency monitors the round-trip time of the CAPWAP heartbeat packets (echo request and response) from the AP to the controller and back.
e) From the **Preferred Mode** drop-down list, choose the mode.
f) Click **Save & Apply to Device**.

8. In the **AP** tab, you can configure these options:

- General

a) In the **General** tab, check the **Switch Flag** check box to enable switches.
b) Check the **Power Injector State** check box if power injector is being used. Use power injectors to provide flexible powering options for APs, such as local power, multiport switches with inline power, or multiport power patch panels.

   Power Injector Selection parameter enables you to protect your switch port from an accidental overload if the power injector is inadvertently bypassed.

c) From the **Power Injector Type** drop-down list, choose power injector type from these options:

   - Installed–This option examines and remembers the MAC address of the currently connected switch port and assumes that a power injector is connected. Choose this option if your network contains older Cisco 6-Watt switches and you want to avoid possible overloads by forcing a double-check of any relocated APs.

     If you want to configure the switch MAC address, enter the MAC address in theInjector Switch MAC Address text box. If you want the AP to find the switch MAC address, leave theInjector Switch MAC Address text box blank.

> **Note**
>
> Each time an AP is relocated, the MAC address of the new switch port fails to match the remembered MAC address, and the AP remains in low-power mode. You must then physically verify the existence of a power injector and reselect this option to cause the new MAC address to be remembered.

- Override–This option allows the AP to operate in high-power mode without first verifying a matching MAC address. You can use this option if your network does not contain any older Cisco 6-W switches that could be overloaded if connected directly to a 12-W AP. The advantage of this option is that if you relocate the AP, it continues to operate in high-power mode without any further configuration. The disadvantage of this option is that if the AP is connected directly to a 6-W switch, an overload occurs.

d) In the**Injector Switch MAC** field, enter the MAC address of the switch either in xx:xx:xx:xx:xx:xx, xx-xx-xx-xx-xx-xx, or xxxx.xxxx.xxxx format.
e) From the **EAP Type** drop-down list, choose the EAP type as *EAP-FAST* , *EAP-TLS* , or *EAP-PEAP*.
f) From the **AP Authorization Type** drop-down list, choose the type as either *CAPWAP DTLS +* or *CAPWAP DTLS*.
g) In the **Client Statistics ReportingInterval** section, enter the interval for 5 GHz and 2.4 GHz radios in seconds.
h) Check the **Enable** check box to enable extended module.
i) From the **Profile Name** drop-down list, choose a profile name for mesh.
j) Click **Save & Apply to Device**.

- Hyperlocation: Cisco Hyperlocation is a location solution that allows to track the location of wireless clients with the accuracy of one meter. Selecting this option disables all other fields in the screen, except NTP Server.

a) In the **Hyperlocation** tab, check the **Enable Hyperlocation** check box.
b) Enter the **Detection Threshold** value to filter out packets with low RSSI. The valid range is -100 dBm to -50 dBm.
c) Enter the **Trigger Threshold** value to set the number of scan cycles before sending a BAR to clients. The valid range is zero to 99.
d) Enter the **Reset Threshold** value to reset value in scan cycles after trigger. The valid range is zero to 99.
e) Enter the **NTP Server** IP address.
f) Click **Save & Apply to Device**.

- BLE: If your APs are Bluetooth Low Energy (BLE) enabled, they can transmit beacon messages that are packets of data or attributes transmitted over a low energy link. These BLE beacons are frequently used for health monitoring, proximity detection, asset tracking, and in-store navigation. For each AP, you can customize BLE Beacon settings configured globally for all APs.

a) In the **BLE** tab, enter a value in the **BeaconInterval** field to indicate how often you want your APs to send out beacon advertisements to nearby devices. The range is from one to 10, with a default value of one.
b) In the **Advertised Attenuation Level** field, enter the attenuation level. The range is from 40 to 100, with a default of 59.
c) Click **Save & Apply to Device**.

- Packet Capture: Packet Capture feature allows to capture the packets on the AP for the wireless client troubleshooting. The packet capture operation is performed on the AP by the radio drivers on the current channel on which it is operational, based on the specified packet capture filter.

a) In the **Packet Capture** tab, choose an **AP Packet Capture Profile** from the drop-down list.
b) You can also create a new profile by clicking the **+** sign.
c) Enter a name and description for the AP packet capture profile.

d) Enter the **Buffer Size**.
e) Enter the **Duration**.
f) Enter the **Truncate Length** information.
g) In the **Server IP** field, enter the IP address of the TFTP server.
h) In the **File Path** field, enter the directory path.
i) Enter the username and password details.
j) From the **Password Type** drop-down list, choose the type.
k) In the **Packet Classifiers** section, use the option to select or enter the packets to be captured.
l) Click **Save**.
m) Click **Save & Apply to Device**.

9. In the **Management** tab, you can configure these options:

- Device

a) In the **Device** tab, enter the **IPv4/IPv6 Address** of the TFTP server, **TFTP Downgrade** section.
b) In the **Image File Name** field, enter the name of the software image file.
c) From the **Facility Value** drop-down list, choose the appropriate facility.
d) Enter the IPv4 or IPv6 address of the host.
e) Choose the appropriate **Log Trap Value**.
f) Enable Telnet, SSH or both configurations, if required.
g) Enable core dump, if required.
h) Click **Save & Apply to Device**.

- User

a) In the **User** tab, enter username and password details.
b) Choose the appropriate password type.
c) In the **Secret** field, enter a custom secret code.
d) Choose the appropriate secret type.
e) Choose the appropriate encryption type.
f) Click **Save & Apply to Device**.

- Credentials

a) In the **Credentials** tab, enter local username and password details.
b) Choose the appropriate local password type.
c) Enter 802.1x username and password details.
d) Choose the appropriate 802.1x password type.
e) Enter the time in seconds after which the session should expire.
f) Enable local credentials, 802.1x credentials, or both as required.
g) Click **Save & Apply to Device**.

- CDPInterface

a) In the **CDPInterface** tab, enable the CDP state, if required.
b) Click **Save & Apply to Device**.

10. In the **Rogue AP** tab, check the **Rogue Detection** check box to enable rogue detection.

11. In the **Rogue Detection Minimum RSSI** field, enter the RSSI value.

This field specifies the minimum RSSI value for which a Rogue AP should be reported. All Rogue APs with RSSI lower than what is configured will not be reported to controller.

12. In the **Rogue Detection TransientInterval** field, enter the transient interval value.

This field indicates how long the Rogue AP should be seen before reporting the controller.

13. In the **Rogue Detection ReportInterval** field, enter the report interval value.

    This field indicates the frequency (in seconds) of Rogue reports sent from AP to controller.

14. Check the **Rogue Containment Automatic Rate Selection** check box to enable rogue containment automatic rate selection.

    The AP selects the best rate for the target Rogue, based on its RSSI.

15. Check the **Auto Containment on FlexConnect Standalone** check box to enable the feature.

    The AP continues containment if it moves to FlexConnect standalone mode.

16. Click **Save & Apply to Device**.

The AP join profile is created or updated. Devices assigned to this profile use its configuration for network operation and management.

What to do next

- Verify that APs have successfully joined and received the new settings by reviewing AP status.
- Adjust profile settings as necessary for site-specific needs or to resolve configuration issues.

Configure an AP profile (CLI)

Configure an AP Profile using commands.

Set up and customize an AP profile using CLI commands to apply network-wide AP configurations.

Use this procedure when you need to define or modify an AP profile for your wireless controller, typically to standardize AP settings or apply feature-specific options.

Before you begin

Identify the name and required settings for your AP profile.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile xyz-ap-profile
   ```

> **Note**
>
> - In an AP profile, the **EAP-FAST** is the default EAP type.
> - When you delete a named profile, the APs associated with that profile does not revert to the default profile.

3. Add a description for the AP profile.

   Example

   ```
   Device(config-ap-profile)# description "xyz ap profile"
   ```

4. Configure DHCP fallback.

   Example

   ```
   Device(config-ap-profile)# ip dhcp fallback
   ```

   > **Note**
   >
   > DHCP fallback is enabled by default. If an AP is assigned a static IP address and cannot reach the controller, the AP falls back to the DHCP. To prevent an AP from switching from a static IP to DHCP, disable the DHCP fallback configuration in an AP join profile.

5. Enable CDP for all Cisco APs.

   Example

   ```
   Device(config-ap-profile)# cdp
   ```

6. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-profile)# end
   ```

7. Optional: Display detailed information about an AP join profile.

   Example

   ```
   Device# show ap profile name xyz-ap-profile detailed
   ```

The AP profile is created or updated with specified settings, and changes are active on the controller.

What to do next

Assign APs to the new profile as required, and verify AP behavior matches the intended configuration.

Configure user for AP management (CLI)

Configure the AP Management using commands.

Set up a management user account to centrally control APs through CLI.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile, and then enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile default-ap-profile
   ```

3. Specify the AP management username and password for managing all of the APs configured to the controller.

   Example

   ```
   Device(config-ap-profile)# mgmtuser username myusername password 0 12345678
   ```

   - **0**: Specifies an UNENCRYPTED password.
   - **8**: Specifies an AES encrypted password.

   > **Note**
   >
   > While configuring a username, ensure that you do not use special characters. Using special characters may cause a configuration error.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(configure-ap-profile)# end
   ```

The AP management user is configured for all APs that are managed by the controller.

Set a private configuration key for password encryption

Configure a Private Configuration key for Password Encryption.

Configure a private configuration key to enable password encryption on the device.

Use this task to set or change a private key used for encrypting passwords. You must do this before you enable AES password encryption.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Set the password encryption keyword.

   Example

   ```
   Device(config)# key config-key password-encrypt 12345678
   ```

   Here, *config-key* refers to any key value with at least eight characters.

   Note

   The *config-key* value must not begin with these special characters:

   - !
   - #
   - ;

3. Enable the encrypted pre-shared key.

   Example

   ```
   Device(config)# password encryption aes
   ```

4. Return to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

You have now enabled password encryption using your specified private configuration key.

What to do next

You can now enable AES password encryption, if needed.

Configure an RF profile (GUI)

Configure an RF profile using GUI.

Create and enable an RF profile to optimize radio frequency settings for your wireless network.

Use this task to define an RF profile that sets parameters for radio bands and device operation. This ensures consistent performance across your wireless deployment.

Before you begin

Use the same RF profile name when configuring the wireless RF tag. If the RF tag contains an RF profile that does not exist, the radios do not operate.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF**.
2. On the **RF Profile** window, click **Add**.
3. In the **General** tab, enter a name for the RF profile. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.
4. Choose the appropriate **Radio Band**.
5. To enable the profile, set the status as **Enable**.
6. Enter a **Description** for the RF profile.
7. Click **Save & Apply to Device**.

The new RF profile is created, enabled, and applied to the device.

What to do next

- Verify that the RF profile settings are visible and active on your device.
- Assign the profile to RF tags or APs, if required, to complete your configuration.

Configure an RF profile (CLI)

Configure an RF profile using commands.

Create and enable an RF profile to optimize radio frequency settings for your wireless network.

Use this task to define an RF profile that sets parameters for radio bands and device operation. This ensures consistent performance across your wireless deployment.

Before you begin

Use the same RF profile name when configuring the wireless RF tag. If the RF tag contains an RF profile that does not exist, the radios do not operate.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an RF profile, and enter RF profile configuration mode.

   Example

   ```
   Device(config)# ap dot11 24ghz rf-profile rfprof24_1
   ```

   Note

   Use the **24ghz** command to configure the 802.11b parameters. Use the **5ghz** command to configure the 802.11a parameters. Use the **6ghz** command to configure the 802.11 6-GHz parameters.

3. Optional: Enable default parameters for the RF profile.

   Example

   ```
   Device(config-rf-profile)# default
   ```

4. Enable the RF profile on the device.

   Example

   ```
   Device(config-rf-profile)# no shutdown
   ```

5. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-rf-profile)# end
   ```

6. Optional: Display a summary of available RF profiles.

   Example

   ```
   Device# show ap rf-profile summary
   ```

7. Optional: Display detailed information about a particular RF profile.

   Example

   ```
   Device# show ap rf-profile name rfprof24_1 detail
   ```

You have created and enabled a new RF profile, and applied it to the device.

What to do next

- Verify that the RF profile settings are visible and active on your device.
- Assign the profile to RF tags or APs as required to complete your configuration.

Configure a site tag (GUI)

Guides through configuring site tags, policy tags, wireless RF tags via GUI and CLI, and demonstrates attaching policy and site tags to access points.

Create and apply a site tag to organize and manage APs within your wireless network.

Use site tags to assign AP join profiles and control plane names to APs, supporting flexible network segmentation and management.

Before you begin

Confirm required AP join profiles and control plane names are already set up.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. On the **Manage Tags** page, click the **Site** tab.
3. Click **Add** to view the **Add Site Tag** window.
4. Enter a name and description for the site tag. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.
5. Choose the required **AP Join Profile** to be attached to the site tag.
6. Choose the required **Control Plane Name**.
7. If required, enable the **Local Site**.

   If you disable Local Site, the site is remote and the deployment uses FlexConnect mode.
8. Click **Save & Apply to Device** .

The site tag is created and applied to the device. The selected AP join profile and control plane name are now associated with the site tag.

What to do next

Verify that your APs are assigned to the correct site tag and check their operational status in the device list.

Configure a site tag for FlexConnect APs (CLI)

Configure a Site Tag using CLI.

Set up a site tag for wireless deployments to enable site-specific configurations using the CLI.

Before you begin

Gather required information, such as the desired site tag name and flex profile name.

Procedure

1. Enter global configuration mode.

   Example

```
Device# configure terminal
```

2. Configure a site tag and enter site tag configuration mode.

   Example

   ```
   Device(config)# wireless tag site rr-xyz-site
   ```

3. Configure a flex profile.

   Example

   ```
   Device(config-site-tag)# flex-profile rr-xyz-flex-profile
   ```

   - You cannot remove flex profile configurations from a site tag that is configured as a local site.

     Use the **no local-site** command to remove local site configurations before applying flex profile configurations.

4. Add a description for the site tag.

   Example

   ```
   Device(config-site-tag)# description "default site tag"
   ```

5. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

   Example

   ```
   Device(config-site-tag)# end
   ```

6. Optional: Display the number of site tags.

   Example

   ```
   Device# show wireless tag site summary
   ```

The site tag is configured on the device, and site-specific settings are applied.

Configure policy tag (GUI)

Configure a Policy Tag using the GUI.

Create and apply a policy tag to group wireless local area network (WLAN) and policy profiles for your network configuration.

Use this task when you need to define or update policy tags for your wireless network devices using the GUI.

Before you begin

- Prepare unique names for policy tags using ASCII characters (32 to 126, no leading or trailing spaces).
- Identify the WLAN and policy profiles you plan to map.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Tags** > **Policy**.

2. Click **Add** to view the **Add Policy Tag** window.
3. Enter a name and description for the policy tag. You can use ASCII characters from 32 to 126, but do not include any leading or trailing spaces.
4. Click **Add** to map WLAN and policy.
5. Choose the WLAN profile to map with the appropriate policy profile, and click the tick icon.
6. Click **Save & Apply to Device**.

Your device has the new policy tag applied. The mapped WLAN and policy profiles are now active based on your configuration.

What to do next

Verify that connected devices use the updated policy tag and that expected network policies are enforced.

Configure a policy tag (CLI)

Configure a policy tag using commands.

Create and apply a policy tag to group wireless local area network (WLAN) and policy profiles for your network configuration.

Use this task when you need to define or update policy tags for your wireless network devices using the CLI.

Before you begin

- Prepare unique names for policy tags using ASCII characters (32 to 126, no leading or trailing spaces).
- Identify the WLAN and policy profiles you plan to map.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure policy tag and enter policy tag configuration mode.

   Example

   ```
   Device(config-policy-tag)# wireless tag policy default-policy-tag
   ```

   Note

   When performing local web authentication, the clients connected to a controller get disconnected intermittently before session timeout.

3. Add a description to a policy tag.

Example

```
Device(config-policy-tag)# description "default-policy-tag"
```

4. Map a remote-LAN profile to a policy profile.

   Example

   ```
   Device(config-policy-tag)# remote-lan remote-lan-name policy
   profile-policy-name port-id port-id-number
   ```

5. Map a policy profile to a WLAN profile.

   Example

   ```
   Device(config-policy-tag)# wlan wlan-name policy profile-policy-name
   ```

   > **Note**
   >
   > Ensure that the WLAN profile is not used by any other profiles. If the AP uses the default profile, ensure that the **no central switching** command is configured on other profiles.

6. Exit policy tag configuration mode, and return to privileged EXEC mode.

   Example

   ```
   Device(config-policy-tag)# end
   ```

7. Optional: Display the configured policy tags.

   Example

   ```
   Device# show wireless tag policy summary
   ```

   > **Note**
   >
   > To view detailed information about a policy tag, use the **show wireless tag policy detailed** *policy-tag-name* command.

Your device has the new policy tag applied. The mapped WLAN and policy profiles are now active based on your configuration.

What to do next

Verify that connected devices use the updated policy tag and the expected network policies are applied.

Configure wireless RF tag (GUI)

Configure an RF (Radio frequency) Tag using GUI.

Create and apply an RF tag that defines radio frequency profiles for device groups.

Procedure

1. a) Choose **Configuration** > **Tags & Profiles** > **Tags** > **RF**.
2. Click **Add** to view the **Add RF Tag** window.
3. Enter a name and description for the RF tag. The name can be ASCII characters from 32 to 126 and must not include leading or trailing spaces.
4. Choose the required **5 GHz Band RF Profile**, **5 GHz Band RF Profile**, and **2.4 GHz Band RF Profile** to be associated with the RF tag.
5. Click **Update & Apply to Device**.

The RF tag is created with the specified RF profiles and applied to the device.

Configure wireless RF tag (CLI)

Configure a RF (Radio frequency) Tag using commands.

Assign and configure an radio frequency (RF) tag on your wireless device using the CLI, to manage radio profile settings per tag.

Perform this task to create or modify a wireless RF tag. This allows you to apply specific 2.4 GHz, 5 GHz, or 6 GHz RF profiles to associated APs.

Before you begin

- You can use only two profiles (2.4-GHz and 5-GHz band RF profiles) in an RF tag.
- You can use only three profiles (2.4-GHz, 5-GHz and 6-GHz band RF profiles) in an RF tag.
- Ensure you use the same AP tag name you created during the AP configuration task.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create an RF tag to enter wireless RF tag configuration mode.

   Example

   ```
   Device(config)# wireless tag rf rftag1
   ```

3. Attach an IEEE 802.11b RF policy to the RF tag.

Example

```
Device(config-wireless-rf-tag)# 24ghz-rf-policy rfprof24_1
```

To configure a dot11a policy, use the **5ghz-rf-policy** command. To configure a 6GHz radio dot11 policy, use the **6ghz-rf-policy** command.

4. Add a description for the RF tag.

   Example

   ```
   Device(config-wireless-rf-tag)# description Test
   ```

5. Exit configuration mode to return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-rf-tag)# end
   ```

6. Display the available RF tags.

   Example

   ```
   Device# show wireless tag rf summary
   ```

7. Display detailed information of a particular RF tag.

   Example

   ```
   Device# show wireless tag rf detailed rftag1
   ```

You have configured the RF tag and it is now available for assignment.

What to do next

Assign the RF tag to the appropriate AP tag or APs if you have not already done so.

Attach a policy tag and site tag to an AP (GUI)

Attach a Policy Tag and Site Tag to an AP using GUI.

Assign a policy tag and site tag to an AP using the GUI.

Use this procedure to associate specific network policies and locations with an AP in your Cisco wireless deployment.

Before you begin

Make sure you have the wired MAC address of the AP.

Procedure

---

1. Choose **Configuration** > **Wireless** > **Access Points**.

   The **All Access Points** section displays details of all the APs on your network.

2. To edit the configuration details of an AP, select the row for that AP.

   The **Edit AP** window is displayed.

3. In the **General** tab and **Tags** section, specify the appropriate policy, site, RF tags, and radio profile that you created on the **Configuration** > **Tags & Profiles** > **Tags** window.
4. Click **Update & Apply to Device**.

   Note

   To see the policy tag, site tag, or the RF tag that is applied to the AP through the GUI, refresh the GUI window.

The AP is now associated with the specified policy, site, and optionally RF tags you selected.

Attach policy tag and site tag to an AP (CLI)

Attach Policy Tag and Site Tag to an AP using commands.

Assign a policy tag and site tag to an AP using the CLI.

Use this procedure to associate specific network policies and locations with an AP in your Cisco wireless deployment.

Before you begin

Make sure you have the wired MAC address of the AP.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a Cisco AP and enters AP profile configuration mode.

   Example

   ```
   Device(config)# ap F866.F267.7DFB
   ```

   Note

   The *mac-address* should be a wired mac address.

3. Map a policy tag to the AP.

   Example

   ```
   Device(config-ap-tag)# policy-tag rr-xyz-policy-tag
   ```

4. Map a site tag to the AP.

   Example

   ```
   Device(config-ap-tag)# site-tag rr-xyz-site
   ```

5. Associate the RF tag.

   Example

   ```
   Device(config-ap-tag)# rf-tag rf-tag1
   ```

6. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-tag)# end
   ```

7. Optional: Display AP details and the tags associated to it.

   Example

   ```
   Device# show ap tag summary
   ```

8. Display the AP name with tag information.

   Example

   ```
   Device# show ap name "ap-name" tag info
   ```

9. Optional: Display the AP name with tag details.

   Example

   ```
   Device# show ap name ap-name tag detail
   ```

The AP is now associated with the specified policy, site, and optionally, RF tags you selected

Enhanced site tag-based load balancing

Explains enhanced site tag-based load balancing concepts, prerequisites, use cases, configuration procedures, and verification commands for optimizing wireless network performance.

A site tag-based load balancing is a controller mechanism that

- distributes APs across Wireless Network Control Daemon (WNCD) processes based on site tags,
- allows configuration of a site load for each site tag to optimize AP assignment considering site size, and
- automatically retains site tags in persistent memory and balances them during controller bootup in descending order of the configured site load.

The controller takes the load balancing decision for a site tag when the first AP from that site tag joins.

Table 5: Feature History

Feature	Release	Feature Information
Enhanced Site Tag-Based Load Balancing	Cisco IOS XE Dublin 17.10.1	When the first AP from a site joins the controller, it takes the decision to load balance the entire site. However, this is done without knowing the site load.

How controllers functioned before load balancing

Previously, controllers did not consider the actual size of sites when making load balancing decisions. The system functioned optimally only when all sites were of approximately equal size. Unequal distribution caused some WNCDs to be overloaded while others remained underused.

The enhanced feature enables administrators to specify a site load for each site tag, ensuring the controller takes site size into account when distributing APs and thus resulting in improved load distribution among WNCDs.

Behavior during controller reboot

- After you configure site load balancing in one or more site tags and reboot the controller, The feature retains actively used site tags and balances them at bootup by prioritizing those with higher configured site load before any APs join.
- If load balancing is configured for a site tag with APs already joined, load balancing assignments remain unchanged unless all APs, including those not in the site tag, disconnect or the controller is rebooted.

assigning a higher site load to one site over the other

Suppose Site Tag X is assigned a higher site load than Site Tag Y. Upon controller bootup, APs from Site Tag X are distributed among WNCDs before those from Site Tag Y, helping prevent excessive load on any single WNCD even if Site Tag X contains many more APs than Site Tag Y.

Analogy: assigning classes to school buses

Think of enhanced site tag-based load balancing like assigning students from different-sized classes (site tags) to school buses (WNCDs). If you ignore class size, you might put all students from the largest class on a single bus, overloading it while leaving other buses half empty. By taking class sizes into account when assigning students, you distribute them more evenly and efficiently among the buses, preventing overcrowding and making better use of all resources. Similarly, by setting site load values, the controller assigns APs more fairly to WNCDs, balancing the work and avoiding overload.

Prerequisites for Enhanced Site Tag-Based Load Balancing

Enhanced Site Tag-Based load balancing feature prerequisites.

- Ensure that you configure the site load.
- We recommended that you configure all the named sites with a load value.

> **Note**
>
> The configured load is only an estimate. It will only be used for site load balancing. Specifically, it does not prevent APs, or clients from joining or associating.

Use cases

The two Use Cases for Site Load Balancing.

The site load configuration uses a load factor instead of an absolute number to address different use cases. Although the load factor does not have to match the number of APs in a site, using the AP count often provides a practical approximation of the site load.

Here are the use cases:

- Sites with normal client density and roaming load can use AP count as a good approximation of site load. Examples include cubicle areas in offices and hospitals.
- For sites with high client density and roaming load, configure a load factor greater than the number of APs. For example, when a site has 200 APs, use a load factor of 300 or 400 to accommodate the increased client load. Examples include stadiums, cafeterias, and conference floors.

Configure site load (CLI)

Configure Site Load in the Controller Console using commands.

Assign a relative load value to a site tag using CLI. This helps the system optimize resource allocation and network performance for wireless deployments.

Use this task when you need to set or adjust the load recommendation for a specific site on your wireless controller. The load value influences how resources are distributed across sites.

Before you begin

Identify the name of the site tag you want to configure.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure site tag and enter site tag configuration mode.

   Example

   ```
   Device(config)# wireless tag site area1
   ```

3. Configure the site load.

Example

```
Device(config-site-tag)# load 200
```

The *load* is the estimate of the relative load reserved for the site. Values range between zero to 1000. The default value zero means no load recommendation for the site.

4. Return to privileged EXEC mode.

Example

```
Device(config-site-tag)# end
```

The controller uses the configured load value for the specified site tag to optimize network resource allocation for that site

What to do next

Optionally, verify your changes by displaying the site tag configuration and confirming the new load value.

Verify enhanced site tag-based load balancing configuration

Verify Site Load Balancing Configuration in the Controller Console.

To view detailed information about a site, use the command:

```
Device# show wireless tag site detailed area1
Site Tag Name        : area1
Description          :
---------------------------------------
AP Profile           : default-ap-profile
Local-site           : Yes
Image Download Profile: default
Fabric AP DHCP Broadcast : Disabled
Fabric Multicast Group IPv4 Address : 239.1.1.1
Site Load            : 200
```

To view the default site tag type for Wireless Network Controller daemon (WNCD) instances, use the command:

```
Device# show wireless loadbalance tag affinity
Tag           Tag type    No of AP's Joined   Wncd Instance
-----------------------------------------------------------------------
area1         SITE TAG                  50         0
area2         SITE TAG                  50         0
area3         SITE TAG                  100        1
area4         SITE TAG                  150        2
```

Wireless radio profiles

Describes wireless radio profile functionality, restrictions, configuration procedures for beam selection, antenna count settings, per-radio per-slot RF tag profiles, and verification methods.

A wireless radio profile is a configuration profile that

- defines radio parameters and beam steering settings for APs with specific antennae

- enables administrators to select antenna count, beam selection mode, and beam direction for supported APs, and
- allows application of customized radio settings to AP slots through RF tag configurations.

Table 6: Feature history Table

Feature Name	Release	Description
Cisco IOS XE 17.6.1	C-ANT9104 antenna support along with beam width and antenna count configuration on the controller	With this feature • you can configure radio profiles for beam-selection APs with the C-ANT9104 antenna. • you can configure antenna count for Cisco Catalyst 9124AXI/D outdoor APs • you can configure the antenna beam-selection for the 5-GHz slots–slot 1 and slot 2, and • you must explicitly configure the beam selection mode for APs with the C-ANT9104 antenna, as there is no default value for the beam-selection configuration.

Beam steering modes

The C-ANT9104 antenna-enabled Cisco Catalyst 9130AX Series APs have precise control over the antennae pattern. You can thus configure the beam-steering direction for the antennae on the controller. The C-ANT9104 antenna-enabled Cisco Catalyst 9130AX Series APs can operate on these beam-steering modes:

- Wide beam
- Narrow beam
- Narrow beam with 10 degrees tilt
- Narrow beam with 20 degrees tilt

After you create the radio profile, link the radio profile under the radio frequency (RF) tag configuration, so that the radio profile is applied to the APs.

Analogy: smart lighting presets

A wireless radio profile is like a preset scene on a smart lighting system. It lets you preconfigure a group of lights (APs) for specific conditions–such as brightness, color, and focus–so applying the scene guarantees the desired ambiance every time

Restrictions

- Radio Resource Management (RRM) configuration is not supported for Cisco ANT9104 antennae.

  RRM features such as Dynamic Channel Assignment (DCA), Radio Transmit Power Control (TPC), Flexible Radio Assignment (FRA), and so on, are disabled on C-ANT9104 antenna-enabled Cisco Catalyst 9130 Series APs.
- If the AP enabled with Cisco ANT9104 antenna uses a software version earlier than Cisco IOS XE 17.6.1, the AP can still join the controller. However, the AP does not function as the operation status of the radios is down.

Configure a wireless radio profile (GUI)

Configure a Wireless Radio Profile using GUI.

Create and apply a wireless radio profile using the GUI

Use this task when you need to define or update radio settings on Cisco wireless devices through the GUI.

Before you begin

Confirm the devices support the intended profile settings.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF/Radio**.
2. On the **Radio Profile** window, click **Add** .
3. Enter a name and description for the radio profile.
4. Choose the appropriate **Antenna Beam** selection. Configure this option for APs connected with the C-ANT9104 antenna.

   Note

   The antenna beam selection is set to **Not Configured** if no settings are detected.

5. Enter the number in the **Number of antenna to be enabled** field.

   Note

   The option is available for the Cisco Catalyst 9124AXE Outdoor Access Points.

6. Click **Save & Apply to Device**.

The new wireless radio profile is saved and applied to selected devices.

Configure a radio profile and beam selection

Configure a Wireless Radio Profile using commands.

Set up a wireless radio profile and define antenna beam selection parameters to optimize wireless performance.

Use this task to create or update a wireless radio profile and specify how the antenna selects its transmission beam using CLI commands.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the radio profile to enter the wireless radio profile configuration mode.

Example

```
Device(config)# wireless profile radio wireless-radio-profile
```

3. Configure the beam selection of the antenna under the new radio profile.

Example

```
Device(config-wireless-radio-profile)# antenna beam-selection narrow tilt 10
```

The wireless radio profile is applied with the specified antenna beam selection settings.

Configure the number of antennas in a wireless radio profile

Configure the Antenna Count in a Wireless Radio Profile using commands.

Set the number of antennas to be enabled for each slot in a wireless radio profile.

Perform this task to ensure the correct number of antennas are active on your wireless device, optimizing radio performance according to deployment needs.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the radio profile to enter the wireless radio profile configuration mode.

Example

```
Device(config)# wireless profile radio wireless-radio-profile
```

3. Configure the number of antennas to be enabled under the new radio profile.

Example

```
Device(config-wireless-radio-profile)# antenna count 4
```

The specified number of antennas are enabled under the radio profile, ensuring the wireless device operates with the intended antenna configuration.

Configure a slot for each radio in the RF tag profile

Configure a Radio Profile for each slot in an RF tag using commands.

Assign a radio profile to each slot in an RF tag, ensuring that specific profile configurations are applied to each radio in a wireless network device.

Each radio (such as 2.4 GHz or 5 GHz) in a device slot must be linked to a radio profile that is under an RF tag. This is achieved using the CLI. Applying the correct radio profile for each slot ensures optimal RF behavior and feature settings.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the RF tag to enter the wireless RF tag configuration mode.

   Example

   ```
   Device(config)# wireless tag rf wireless-rf-tagname
   ```

3. Configure the 802.11a or 802.11b radio profile.

   Example

   ```
   Device(config-wireless-rf-tag)# dot11 5ghz slot1 radio-profile
   wireless-radio-profile
   ```

You have configured each radio slot under the RF tag with the designated radio profile. The device applies the profile settings to the correct radio.

Verify a radio profile

Verify the summary of all configured radio profiles using commands.

To view the summary of all the configured radio profiles, use the command:

```
Device# show wireless profile radio summary

Number of radio-profiles: 3

Antenna Profile Name             Description
------------------------------------------------------------------------
radio-profile-1                  Custom profile for Slot1

antenna-ewlc                     Add description
default_radio_profile            Preconfigured default radio profile
```

To view detailed information about the parameters configured for a radio profile, use the command:

```
Device# show wireless profile radio detailed radio-profile-name
Radio Profile name                              : radio-profile-1
Description                                     : Custom profile for slot1
Beam-Selection                                  : Wide beam
```

To view radio profile and RF tag information, use the command:

```
Device # show ap name Cisco-AP tag info
AP Name            : Cisco-AP
AP Mac             : 04xx.40xx.XXXX

Applied Tags :
-----------------------------------------
  Tag Type             Tag Name
  -----------------------------------------
  RF Tag               test-rf
  Site Tag             default-site-tag
  Policy Tag           default-policy-tag

Tag/Profile Type Misconfigured
-----------------------------------------
RF Tag No
Policy Tag No
Site Tag No
Flex profile No
AP join profile No
2.4GHz Rf Profile No
5 GHz Rf Profile No
5 GHz Slot1 Radio Profile NO
5 GHz Slot2 Radio Profile Yes

Resolved Tags :
-----------------------------------------
Tag Source         : Static

  Tag Type             Tag Name
  -----------------------------------------
  RF Tag               test-rf
  Site Tag             default-site-tag
  Policy Tag           default-policy-tag
```

To display beam selection and the number of antennas, run the commands:

```
Device# show wireless profile radio detailed radio-profile-1
Radio Profile name : radio-profile-1
Description        : Custom profile for slot1
Beam-Selection     : Wide beam
```

```
Device# show ap name cisco-ap config slot 1 | section 11n
802.11n Antennas
      Number of Antennas selected               : 2
      Supported Antenna  modes                  : 1x1 2x2 4x4
      Antenna port mapping                      : AB
      SIA Status                                : Not Present
```

```
Device# show ap name cisco-ap config slot 1 | include beam
Beam Selection : Narrow from centre 20
```

AP filters

Introduces AP filter concepts and provides procedures for setting tag priority, managing tag precedence, creating AP filters, configuring filter priorities, and verifying configurations.

An AP filter is a global tagging mechanism that

- associates APs with specific tag sources based on defined filter criteria
- organizes and applies tag sources in priority order according to configuration, and
- enables flexible control over which tags are assigned to each AP throughout the device discovery and onboarding process.

Additional reference information

AP filters function similarly to access control lists (ACLs) on the controller and operate at the global level. You can create AP filters based on AP names or other attributes, and incorporate filter criteria into discovery requests. Tag sources for APs can originate from static configuration, the AP filter engine, per-AP Plug and Play (PnP), or default configuration. The AP filter feature determines tag precedence, ensuring the correct tags are assigned according to organizational policies.

You cannot disable the AP filter feature. However, you can configure the priority of tag sources using the **ap filter-priority** <**priority**> <**filter-name**> command.

You can specify tag names at the PnP server (similarly to Flex groups or AP groups). During AP discovery and join requests, the AP stores and sends the allocated tag name as part of the request, allowing for automated and dynamic tag assignment.

Analogy: ticket sorting gate

An AP filter is like a ticket sorting gate at an event venue. Each attendee (AP) presents information at the entrance, and the sorting gate (filter) checks their details to assign them to the right group or area (tag source) based on predefined criteria and priorities. Just as event organizers use sorting gates to efficiently direct attendees, network administrators use AP filters to efficiently assign tags and manage APs in large wireless environments.

Set tag priority (GUI)

Configure Tag Priority using GUI.

Set the order of priority for tag sources in AP configuration using the GUI

Use this task whenever you want to control which tag source is applied first to your AP configurations in the system. This helps ensure the desired tag assignment behavior.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags** > **AP** > **Tag Source**.
2. Drag and drop the tag sources to change priorities.

The system saves and applies the new tag source priority for AP configuration.

Set tag priority

Configure Set Tag Priority using commands.

Resolve ambiguity in tag assignment by defining the priority of tag sources for access points.

When an AP joins the wireless controller, it picks up tags from multiple sources. Setting tag source priorities ensures APs use tags from the preferred source. If you do not configure precedence, default priorities are used.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Optional: Configure AP tag source priority. Default priorities for Static, Filter, and PnP.

   Example

   ```
   Device(config)# ap tag-source-priority 2 source pnp
   ```

3. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

4. Revalidate AP tag sources. The priorities become active only after this command is run.

   Example

   ```
   Device# ap tag-sources revalidate
   ```

   Note

   If you change the priorities for Filter and PnP, and need to verify the changes, run the **revalidate** command.

The controller applies the new tag source priorities, and APs assign tags based on your configured precedence.

Create an AP filter (GUI)

Create an AP filter using GUI.

Define an AP filter to automatically apply tags to APs that match specific naming patterns.

Use this task to organize and manage access points by creating filter rules based on AP names and assigning tags for policy, site, or RF profiles.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags** > **AP** > **Filter** .
2. Click **Add** .

3. In the **Associate Tags to AP** dialog box that is displayed, enter the **Rule Name** , the **AP name regex** , and the **Priority** .
4. Optional: You can also choose the policy tag from the **Policy Tag Name** drop-down list, the site tag from the **Site Tag Name** drop-down list and the RF tag from the **RF Tag Name** drop-down list.
5. Click **Apply to Device** .

The AP filter is created. Any AP that matches the specified pattern is automatically assigned the designated tags.

Create an AP filter (CLI)

Create an AP filter using CLI.

Configure an AP filter using the CLI to apply specific policy, RF, and site tags to APs matching a name pattern.

- Identify the regular expression pattern for the AP names you want to filter.
- Determine the appropriate policy, RF, and site tag names to use.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP filter.

   Example

   ```
   Device(config)# ap filter filter-1
   ```

3. Configure the AP filter based on regular expression.

   Example

   ```
   Device(config-ap-filter)# ap name-regex testany
   ```

   For example, if you have named an AP as **ap-lab-12**, then you can configure the filter with a regular expression, such as **ap-lab-\d+** , to match the AP name.

4. Configure a policy tag for this filter.

   Example

   ```
   Device(config-ap-filter)# tag policy pol-tag1
   ```

5. Configure an RF tag for this filter.

   Example

   ```
   Device(config-ap-filter)# tag rf rf-tag1
   ```

6. Configure a site tag for this filter.

Example

```
Device(config-ap-filter)# tag site site1
```

7. Exit configuration mode to return to privileged EXEC mode.

Example

```
Device(config-ap-filter)# end
```

The AP filter is configured with the specified criteria and tag associations. APs with names matching your pattern will have the corresponding tags applied.

Configure filter priority (GUI)

Configure and set up filter priority using GUI.

Configure and manage the priority of AP filters to control tag associations on APs.

Perform this task when you need to add a new AP filter or change the priority of an existing AP filter using the GUI.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Tags** > **AP** > **Filter**.
2. Choose:
   a) To set up a new AP filter, click **Add**. In the **Associate Tags to AP** dialog box that is displayed, enter the **Rule Name**, the **AP name regex** and the **Priority**. Optionally, you can also select the **Policy Tag Name**, the **Site Tag Name** and the **RF Tag Name**. Click **Apply to Device**.
   b) To update the priority of an existing AP filter, click on the filter and in the **Edit Tags** dialog box and change the **Priority**. In case the filter is inactive, you cannot configure a priority. Click **Update and Apply to Device**.

The AP filter priority is configured or updated and applied to the specified devices.

Configure filter priority (CLI)

Configure and set up filter priority using CLI.

Configure and manage the priority of AP filters to control tag associations on APs.

Perform this task when you need to add a new AP filter or change the priority of an existing AP filter using the GUI.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure AP filter priority. Valid values range from zero to 1023. The value zero represents the highest priority.

   Example

   ```
   Device(config)# ap filter priority 10 filter-name test1
   ```

   This step is necessary to activate the filter.

3. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-ap)# end
   ```

The AP filter priority is configured or updated and applied to the specified devices.

Verify AP filter configuration

Verify AP Filter Configuration using commands.

These **show** commands are used to display tag sources and filters, and their priorities.

To view the tag source priorities, use the command:

```
Device# show ap tag sources

Priority Tag source
--------------------------------
0 Static
1 Filter
2 AP
3 Default
```

To view the available filters, use the command:

```
Device# show ap filter all

Filter Name             regex             Policy Tag               RF Tag
          Site Tag
-------------------------------------------------------------------------------------------
first                   abcd              pol-tag1                 rf-tag1
          site-tag1
test1                   testany
          site1
filter1                 testany
```

To view the list of active filters, use the command:

```
Device# show ap filters active

Priority   Filter Name       regex          Policy Tag               RF Tag
           Site Tag
-------------------------------------------------------------------------------------------
10         test1             testany
           site1
```

To view the source of an AP tag, use the command:

```
Device# show ap tag summary

Number of APs: 4

AP Name           AP Mac          Site Tag Name    Policy Tag Name    RF Tag
Name    Misconfigured Tag Source
————————————————————————————————————————————————————————————————————————
AP002A.1034.CA78 002a.1034.ca78 named-site-tag   named-policy-tag   named-rf-tag
   No Filter
AP00A2.891C.2480 00a2.891c.2480 named-site-tag   named-policy-tag   named-rf-tag
   No Filter
AP58AC.78DE.9946 58ac.78de.9946 default-site-tag default-policy-tag
default-rf-tag No AP
AP0081.C4F4.1F34 0081.c4f4.1f34 default-site-tag default-policy-tag
default-rf-tag No Default
```

Location configuration

Explains location configuration for access points, covering restrictions, configuration procedures, location assignment methods, SNMP MIB integration, and verification of location settings and statistics.

This feature works in conjunction with the existing tag resolution scheme. The location is considered as a new tag source to the existing system.

A location is a network entity that

- represents a site or physical area where one or more APs are deployed
- associates a specific set of tags—policy, RF, and site tags—with those APs, and
- links each tag set to a collection of Ethernet MAC addresses corresponding to APs at that location.

Location features and usage

Locations serve as a new tag source within the tag resolution scheme, functioning similarly to static tag sources. During location configuration, you can

- configure a site or location for an AP
- assign a set of tags for the location, and
- add APs to the location.

The combination of unique tags and associated MAC addresses distinguishes each location and enables organized network management.

Restriction for location configuration

Location Configuration prerequisites.

If you configure an AP in one location, you cannot configure the same AP in another location.

Configure a location for an AP (GUI)

Configure a Location for an AP using GUI.

Create and classify a new location for an AP. Apply the appropriate type and client density settings.

Use this task when you need to add a location for AP deployment and associate location-specific policies or tags.

Before you begin

When you create local and remote sites in the basic setup workflow, the corresponding policies and tags are created automatically. Tags and policies created in the basic setup cannot be modified using the advanced workflow. Similarly, items created with the advanced workflow cannot be changed in the basic setup.

Procedure

1. Choose **Configuration** > **Wireless Setup** > **Basic**.
2. On the **Basic Wireless Setup** window, click **Add**.
3. In the **General** tab, enter a name and description for the location.
4. Set the **Location Type** as either *Local* or *Flex*.
5. Use the slider to set **Client Density** as *Low*, *Typical*, or *High*.
6. Click **Apply**.

The new location is saved and ready for assignment and policy application.

Configure a location for an AP (CLI)

Configure a Location for an AP using CLI.

Assign and configure a location for an AP using CLI commands.

Use this task to logically organize APs by location for better device management and policy assignment.

Before you begin

Confirm the AP is registered and operational.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a location for an AP.

   Example

   ```
   Device(config)# ap location name location1
   ```

   Run the **no** form of this command to remove location for an AP.

3. Configure tags for the location.

   Example

   ```
   Device(config-ap-location)# tag policy policy_tag
   ```

```
Device(config-ap-location)# tag rf rf_tag
```

```
Device(config-ap-location)# tag site site_tag
```

4. Add description to the location.

   Example

```
Device(config-ap-location)# location description
```

5. Return to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

   Example

```
Device(config-ap-location)# end
```

The AP location is set and the associated tags and a description are configured.

Add an AP to location (GUI)

Add an AP to the location using GUI.

Add an AP to a specific location using the GUI.

Use this task to associate APs with a defined location, which enables accurate AP count and location tagging in the GUI. Ensure the AP tag source is set to **location** for correct display–if needed, run the **no ap** *ap-mac* command on the controller to reset the AP tag source to default.

Before you begin

- Confirm you have the MAC address of the APs to be added.
- Optionally, prepare a CSV file with AP MAC addresses for bulk upload.

Procedure

---

1. Choose **Configuration** > **Wireless Setup** > **Basic**.
2. In the **Basic Wireless Setup** window, click **Add** to configure these options:
   - General
   - Wireless Networks
   - AP Provisioning
3. In the **AP Provisioning** tab and **Add/Select APs** section, enter the AP MAC address and click the right arrow to add the AP to the associated list. The MAC address can be either in *xx:xx:xx:xx:xx:xx*, *xx-xx-xx-xx-xx-xx*, or *xxxx.xxxx.xxxx* format.

   You can also add a CSV file from your system. Ensure that the CSV has the MAC Address column.

4. Use the search option in the **Available AP List** to select the APs from the **Selected AP** list and click the right arrow to add the AP to the associated list.
5. Click **Apply**.

The selected APs are added to the specified location, and location tagging is updated in the GUI.

Adding an Access Point to the Location (CLI)

Add an access point (AP) to a specified location on your network controller using CLI commands for efficient network management.

Use this task if you need to organize APs by site, floor, building, or area to simplify administration and monitoring.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a location for an access point.

   Example

   ```
   Device(config)# ap location name location-name
   ```

   ```
   Device(config)# ap location name location1
   ```

3. Add an access point to the location.

   Example

   ```
   Device(config-ap-location)# ap-eth-mac ap_ethernet_mac
   ```

   ```
   Device(config-ap-location)# ap-eth-mac 188b.9dbe.6eac
   ```

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-location)# end
   ```

   Alternatively, you can also press Ctrl-Z to exit global configuration mode.

> **Note**
>
> After adding an AP to a location, the AP may reset automatically to get the new configuration

The access point is successfully assigned to the specified location. The AP may reset to update its configuration.

SNMP MIB

A set of managed objects that represent logical and physical entities.

The SNMP MIB provides information on a set of managed objects that represent logical and physical entities, and relationships between them.

Table 7: MIB Objects and Notes

MIB Objects	Notes
cLApLocationName	Provides the name of the AP location.
cLApLocationPolicyTag	Provides the policy tag configured on the location.
cLApLocationSitetag	Provides the site tag configured on the location.
cLApLocationRfTag	Provides the RF tag configured on the location.
cLAssociatedApsApMac	Provides the configured APs on the location.

Verify location configuration

Verify Location Configuration using commands.

To view the summary of AP location configuration, use the command:

```
Device# show ap location summary

Location Name    Description     Policy Tag           RF Tag           Site
Tag
-------------------------------------------------------------------------------------------
first            first floor     default-policy-tag   default-rf-tag
default-site-tag
second           second floor    default-policy-tag   default-rf-tag
default-site-tag
```

To view the AP location configuration details for a specific location, use the command:

```
Device# show ap location details first

Location Name.....................: first
Location description..............: first floor
Policy tag........................: default-policy-tag
Site tag..........................: default-site-tag
RF tag............................: default-rf-tag
```

```
Configured list of APs
005b.3400.0af0
005b.3400.0bf0
```

To view the AP tag summary, use the command:

```
Device# show ap tag summary

Number of APs: 4
AP Name      AP Mac           Site Tag Name      Policy Tag Name      RF Tag
Name      Misconfigured  Tag Source
-------------------------------------------------------------------------------
Asim_5-1     005b.3400.02f0   default-site-tag   default-policy-tag
default-rf-tag   Yes            Filter
Asim_5-2     005b.3400.03f0   default-site-tag   default-policy-tag
default-rf-tag   No             Default
Asim_5-9     005b.3400.0af0   default-site-tag   default-policy-tag
default-rf-tag   No             Location
Asim_5-10    005b.3400.0bf0   default-site-tag   default-policy-tag
default-rf-tag   No             Location
```

Verify location statistics

Verify Location Statistics using commands.

To view the AP location statistics, use the command:

```
Device# show ap location stats

Location name    APs joined     Clients joined     Clients on 11a     Clients
 on 11b
-------------------------------------------------------------------------------
first             2             0                   3                  4

second            0             0                   0                  0
```

3 Wireless Management Interface

Topics:

- Wireless management interfaces
- WMI Configuration Tasks

Explains the Wireless Management Interface configuration and provides step-by-step guidance for setting up, migrating, and verifying interface settings on wireless controllers.

Wireless management interfaces

Explains Wireless Management Interface configuration, outlines step-by-step setup and migration processes, and provides verification procedures to ensure successful deployment and management on wireless controllers.

The Wireless Management Interface (WMI) is a mandatory Layer 3 interface on the controller that

- supports all communications between the controller and access points,
- enables Control And Provisioning of Wireless Access Points (CAPWAP) or inter-controller mobility messaging and tunneling traffic, and
- serves as the default interface for in-band management and connectivity to enterprise services such as AAA, syslog, and SNMP.

You can use the WMI IP address to connect to the device using SSH or Telnet. You can also access the GUI through HTTP or HTTPS by entering the WMI IP address in a browser address field.

Supported Protocols

Starting from Cisco IOS XE Release 17.6.1, the controller can use Ethernet Service Port (SP) (Management Interface VRF or GigabitEthernet 0) for specific management or control plane protocols

Supported protocols for the WMI include:

- Simple Network Management Protocol (SNMP)
- RADIUS (both for user authentication to the controller and wireless client authorization)
- TACACS+
- Syslog
- Network Time Protocol (NTP)
- SSH, Network Configuration Protocol (NETCONF), or HTTPS
- NetFlow

Best practices and restrictions for WMI

Best practices for WMI

- **IP Address Configuration**: Configure the Wireless Management Interface (WMI) with a single IP address, either IPv4 or IPv6. Alternatively, use a dual-stack configuration for flexibility in addressing.
- **IPv6 Recommendations**: Assign a static IPv6 address in WMI. Do not configure the IPv6 address using the **ipv6 auto-config** command for the WMI.
- Always use a dedicated wireless management VLAN and configure the WMI as a Switched VLAN Interface (SVI).
- If you configure the uplink port or port channel to the next-hop switch as a dot1q trunk, ensure the wireless management VLAN is allowed or tagged on the trunk.

This recommendation is applicable to all AP deployment modes (local, FlexConnect, or SDA).

The recommendation is not applicable for certain scenarios:

- The WMI is a Layer 3 port on a controller deployed in a public cloud.
- The WMI is a loopback interface for embedded wireless controller in Cisco Catalyst 9000 switches.

Restrictions

- Only one WMI is allowed on a controller.
- Use only a Layer 2 interface or anA SVI to configure the WMI.
- Do not use a Layer 3 interface for guest anchor scenarios.
- Use a Layer 3 interface for public cloud deployments only.
- Use the WMI to terminate CAPWAP traffic.
- The **ipv6 auto-config** command is not supported.

Change the WMI interface when RMI is configured

Occasionally, you may need to change the Wireless Management Interface (WMI). If Redundant Management Interface (RMI) is configured on your network, this task shows you how to change the WMI interface.

Procedure

1. Remove all RMI configurations.
2. Save the changes using the **write memory** command.
3. Reload the controller.
4. Change the WMI interface.
5. Reconfigure the RMI to use the same interface as the WMI.
6. Save the changes using the **write memory** command.
7. Reload the controller.

The WMI interface is changed, and the RMI is reconfigured to match, maintaining operational consistency.

WMI Configuration Tasks

Details essential configuration tasks for the Wireless Management Interface, including VLAN migration, sample interface configurations, step-by-step CLI setup for controllers, and verification of interface settings for optimal performance.

Migrate VLANs safely for wireless management

Safely migrate from VLAN A to VLAN B for wireless management, ensuring all devices—including those behind Workgroup Bridges (WGB)—continue to correctly receive multicast traffic.

> **Caution**
>
> Do not remove VLAN A or SVI A before creating VLAN B or SVI B and associating SVI B with WMI. This sequence is critical to prevent multicast traffic issues.

Before you begin

Procedure

1. Create VLAN B on the relevant network switches.
2. Configure SVI B (Switched Virtual Interface for VLAN B).
3. Associate SVI B with the WMI interface.
4. Verify multicast traffic is flowing correctly to all devices, including those behind WGBs.
5. Once functionality is confirmed, remove VLAN A from the network configuration.
6. Delete SVI A (Switched Virtual Interface for the old VLAN).
7. Document changes and monitor for any multicast or connectivity issues.

The VLANs are migrated, and multicast traffic is properly received by all devices, including those behind WGBs.

Sample interface configuration

Layer 3 interface configuration:

```
interface GigabitEthernet2
no switchport
ip address <ip_address> <mask>
negotiation auto
no mop enabled
no mop sysid
end
```

Layer 2 interface configuration:

```
interface GigabitEthernet2
switchport trunk allowed vlan 25,169,504
switchport mode trunk
negotiation auto
no mop enabled
no mop sysid
end
```

Configure the WMI interface of a controller (CLI)

Configure the WMI interface on a controller using the CLI.

Set up a controller, and assign a dedicated VLAN and interface for wireless management.

The examples assume the GigabitEthernet 2 interface connects to a trunk interface and that you want to configure multiple VLANs and dedicate one of them for the WMI interface.

Before you begin

- Ensure you have physical console access (for the Cisco Catalyst 9800 Series Wireless Controller Appliance) or a virtual console access (Cisco Catalyst 9800 Series Wireless Controller for Cloud).
- Determine your management VLAN, interface IP address, and credentials.

Procedure

1. Access the CLI using Video Graphics Array(VGA) or monitor console from your preferred hypervisor.
2. Terminate the configuration wizard.

```
Would you like to enter the initial configuration dialog? [yes/no]:
 no
Would you like to terminate autoinstall? [yes]:
 yes
```

3. Enter the global configuration mode and configure the login credentials.

```
Device# conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Device(config)# username <name> privilege 15 password <yourpwd>
```

4. Set a hostname.

```
Device(config)# hostname C9800
```

5. Configure the VLAN for the wireless management interface.

```
Device(config)# vlan 201
Device(config-vlan)# name wireless_management
```

6. Configure the Layer 3 SVI for wireless management interface.

```
Device(config)# int vlan 201
Device(config-if)# description wireless-management-interface
Device(config-if)# ip address 172.16.201.21 255.255.255.192
Device(config-if)# no shutdown
```

7. Configure an interface as trunk and allow the wireless management VLAN.

```
Device(config-if)# interface GigabitEthernet2
Device(config-if)# switchport mode trunk
Device(config-if)# switchport trunk allowed vlan 201,210,211
Device(config-if)# shut
```

```
Device(config-if)# no shut
```

> **Note**
>
> VLANs 210 and 211 are added to the trunk to carry client traffic.

8. Configure a default route or a more specific route to reach the device.

```
Device(config-if)# ip route 0.0.0.0 0.0.0.0 172.16.201.1
```

The controller is configured with a dedicated wireless management VLAN and SVI. You can access the device using SSH, telnet, or GUI. You can use the Cisco Catalyst Center or Cisco Prime Infrastructure to continue with the Day 0 configuration.

Verify WMI Settings

Verify Wireless Management Interface Settings using CLI.

Verify if the Layer 3 interface is configured correctly.

```
Device# show run int vlan 201

Building configuration...

Current configuration : 128 bytes
!
interface Vlan201
 description wireless-management-interface
 ip address 172.16.201.21 255.255.255.0
 no mop enabled
 no mop sysid
end
```

Verify if the wireless management VLAN is active on the uplink to the network. In this case, the uplink is a trunk interface, so the VLAN needs to be active and forwarding state.

```
Device# show interfaces trunk

Port        Mode             Encapsulation  Status        Native vlan
Gi2         on               802.1q         trunking      1
.....
Port        Vlans allowed on trunk
Gi2         201,210-211
.....
Port        Vlans allowed and active in management domain
Gi2         201,210-211
....
Port        Vlans in spanning tree forwarding state and not pruned
Gi2         201,210-211
....
```

Verify if the wireless management interface is up.

```
Device# show ip int brief | i Vlan201
Vlan201   172.16.201.21  YES NVRAM  up  up
```

Verify if the selected interface has been configured as wireless management.

```
Device# show wireless interface summary

Wireless Interface Summary

Interface Name Interface Type VLAN ID IP Address     IP Netmask   NAT-IP Address
 MAC Address
-----------------------------------------------------------------------------------------
Vlan201       Management       201   172.16.201.21  255.255.255.0  0.0.0.0
   001e.e51c.a7ff
```

4 Configure Network Address Translation for Wireless Management IP

Topics:

- Network address translation

Guides the configuration of Network Address Translation for wireless management interfaces and outlines Control and Provisioning of Wireless Access Points discovery settings for public and private IP environments.

Network address translation

Describes Network Address Translation (NAT) and its security details.

Network Address Translation (NAT) is a networking mechanism that

- enables private IP networks that use non-registered IP addresses to connect to the Internet
- translates private addresses from the internal network into public addresses before forwarding packets onto another network, and
- provides security by effectively hiding the private network details through advertising only few addresses for the entire internal network.

NAT deployment with Cisco Catalyst 9800 wireless controller

If you want to deploy your Cisco Catalyst 9800 Wireless Controller on a private network and make it reachable from internet, you need to have the controller behind a router, firewall, or other gateway device that uses one-to-one mapping Network Address Translation (NAT).

To do so, perform these steps:

- Configure the NAT device with 1:1 static mapping of the Wireless Management interface IP address (private IP) to a unique external (public) IP address configured on the NAT device.
- Enable the NAT feature on the Wireless Controller and specify its external public IP address. This public IP is used in the discovery responses to APs, so that the APs can then send CAPWAP packets to the right destination.
- Make sure that the external APs discover the public IP of the controller using DHCP, DNS, or PnP.

Note

Certain ISP routers performing NAT may assign the same public source PORT to different APs. This results in the WLC receiving CAPWAP traffic from same IP:PORT but from different APs. The controller is unable to differentiate the packets are from differnent APs, even if packet A is for DATA and Packet B is for CTRL. The controller does not support CAPWAP connections from different APs behind NAT using same SRC IP:PORT.

Note

You need not enable NAT if the Cisco Catalyst 9800 Wireless Controller is deployed with a public address. Instead you will need to configure the public IP directly on the Wireless Management Interface (WMI).

The IPv6 addresses that are not supported are:

- IPv6 Documentation addresses - 2001:db8::/32
- IPv6 Site Local addresses - fec0::/10
- IPv6 Link Local addresses - fe80::/10
- IPv6 Loopback addresses - ::1

- IPv6 Multicast addresses - FF00::/8

CAPWAP discovery

Describes CAPWAP discovery and its environment details.

CAPWAP discovery is a network mechanism that

- enables lightweight access points to discover wireless controllers using CAPWAP discovery mechanisms
- facilitates CAPWAP join requests from access points to controllers, and
- allows controllers to send CAPWAP join responses to access points for joining the controller.

NAT environment response methods

If the wireless controller is behind a NAT device, the controller responds to the discovery response in the following ways:

- Using the public IP
- Using the private IP
- Using public and private IP

The Public IP needs to be mapped to the controller's Private IP using static 1:1 NAT configuration on the router or firewall performing the NAT translation.

If your wireless controller manages only Access Points reachable through the public internet (external APs), you need to configure the controller so it responds with only the Public IP in the discovery response.

If your wireless controller manages both internal and external APs, you need to configure the controller so it responds with both Public and Private IPs in the discovery response.

> **Note**
>
> In NAT deployments, the APs running internally and externally must use different AP join profiles with CAPWAP Discovery Private and Public enabled separately. This behaviour was introduced from the 17.9.5 release and applies to APs upgraded to Cisco IOS XE 17.9.5, 17.9.6, 17.9.m (m>=5), 17.12.n (n>=1) and later releases.

Configure wireless management interface with a NAT public IP (CLI)

Configure wireless management interface with a NAT public IP using commands.

Enable the controller to use a public NAT IP address for wireless management interface communication.

Configure the controller to use the public NAT IP (this is the public IP that has been configured on the NAT device to statically map 1:1 the WMI's private IP address). This configuration allows the wireless management interface to operate in NAT environments where public IP access is required.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Define the management interface.

Example

```
Device(config)# wireless management interface interface-type interface-number
```

Example

```
Device(config)# wireless management interface vlan 20
```

- *interface-type*: Refers to the VLAN, Gigabit, or loopback types.
- *interface-number*: Is the interface number.

3. Define the external NAT or Public IP.

Example

```
Device(config-mgmt-interface)# public-ip external-public-ip
```

Example

```
Device(config-mgmt-interface)# public-ip 2.2.2.2
```

4. Return to privileged EXEC mode.

Example

```
Device(config-mgmt-interface)# end
```

The wireless management interface is now configured with the NAT public IP and can communicate through the NAT device using the specified public IP address mapping.

CAPWAP discovery IP response configuration

Describes the configuration of CAPWAP discovery to respond with specific IP address types using CLI commands.

CAPWAP discovery IP response configuration is a wireless controller setting that controls which IP addresses the controller includes in CAPWAP discovery responses to access points.

CAPWAP discovery response behavior

The controller's IP response behavior in CAPWAP discovery varies based on configuration:

- **Default behavior**: If the wireless management interface is configured with a public IP, the controller responds with both Public and Private IP in the CAPWAP discovery response.
- **Configurable setting**: The setting to determine the IP (private or public) to include in the discovery response is available in the AP Join profile.

Configure the Controller to Respond only with a Public IP (CLI)

Configure the Controller to respond only with a Public IP using commands.

Enable APs to join the controller over public IP only by preventing responses with internal IP addresses.

Configure the Controller to respond only with a Public IP using commands when you need to restrict AP discovery to public interfaces only.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

   Example

   ```
   Device(config)# ap profile xyz-ap-profile
   ```

3. Instruct the controller to not respond with the internal IP.

   Example

   ```
   Device(config-ap-profile)# no capwap-discovery private
   ```

   Enables AP to join the controller over Public IP only.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-profile)# end
   ```

The controller is now configured to respond only with public IP addresses, ensuring that APs can only join using the public interface.

Configure the controller to respond only with a private IP (CLI)

Configure the controller to respond only with a private IP using commands.

Configure the controller to respond only with private IP addresses during CAPWAP discovery.

This configuration ensures that access points join the controller using private IP addresses instead of public IP addresses, which is useful in network environments where direct public IP communication should be restricted.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

   Example

   ```
   Device(config)# ap profile xyz-ap-profile
   ```

3. Instruct the controller to not respond with the public IP.

   Example

   ```
   Device(config-ap-profile)# no capwap-discovery public
   ```

   Enable AP to join the controller over private IP only.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-profile)# end
   ```

The controller is now configured to respond only with private IP addresses during CAPWAP discovery, ensuring APs join using private network connections.

Verify NAT settings

Provides commands for verifying NAT settings and AP join profile configurations.

Verify NAT Settings using commands.

```
Device# show wireless interface summary

Wireless Interface Summary

Interface Name Interface Type VLAN ID IP Address     IP Netmask     NAT-IP
Address   MAC Address
-------------------------------------------------------------------------------------
Vlan20         Management     20      10.58.20.25    255.255.255.0  2.2.2.2
  001e.4963.1cff
```

To verify the settings in the AP join profile, use this command

```
Device# show run | b ap profile

ap profile default-ap-profile
 no capwap-discovery private
 description "default ap profile"
...
```

5 Hardware Features

Topics:

- BIOS Protection on the Controller
- USB disablement

Outlines hardware security and management capabilities for controllers, detailing BIOS protection and upgrade procedures, as well as USB disablement via CLI to enhance device integrity and reduce vulnerabilities.

BIOS Protection on the Controller

BIOS Protection and how it works.

BIOS Protection enables you to protect and securely update BIOS flash for Intel-based platforms. If BIOS Protection is not used, the flash utility that stores the BIOS for an Intel platform is not write-protected. As a result, when BIOS updates are applied, malicious code also makes its way through.

By default, BIOS Protection works by bundling the flash containing the BIOS image, and by accepting updates only through the BIOS capsules that enable writing on the BIOS Flash.

BIOS or ROMMON Upgrade with BIOS Protection

Configure BIOS or ROMMON Upgrade with BIOS Protection.

To upgrade BIOS or ROMMON use the BIOS Protection feature as follows:

1. The new BIOS image capsule bundled together with the ROMMON binary is inserted into the media of the Cisco device by the ROMMON upgrade scripts.
2. The Cisco device is then reset for the new BIOS/ROMMON upgrade to take place.
3. On reset, the original BIOS detects the updated capsule and determines if the updated BIOS is available.
4. The original BIOS then verifies the digital signature of the BIOS capsule. If the signature is valid, the original BIOS will remove write-protection from the flash utility and update the SPI flash with the new BIOS image. If the BIOS capsule is invalid, the SPI flash is not updated.
5. After the new BIOS/ROMMON image is written to the SPI flash, the required regions of the SPI flash are once again write-protected.
6. After the card is reset, the updated BIOS is rebooted.
7. The capsule is deleted by BIOS.

Upgrading BIOS

Upgrade BIOS using commands.

Procedure

Use the **upgrade rom-monitor filename** command to update the BIOS capsule.

Example

```
upgrade rom-monitor filename bootflash:capsule.pkg <slot>
```

Example

The following example shows you how to verify a BIOS Protection upgrade:

```
Device# upgrade rom-monitor filename bootflash:qwlc-rommon-capsule-p106.pkg
all
Verifying the code signature of the ROMMON package...
Chassis model AIR-CT5540-K9 has a single rom-monitor.

Upgrade rom-monitor
```

```
Target copying rom-monitor image file

Secure update of the ROMMON image will occur after a reload.

8388608+0 records in
8388608+0 records out
8388608 bytes (8.4 MB, 8.0 MiB) copied, 11.9671 s, 701 kB/s
131072+0 records in
131072+0 records out
131072 bytes (131 kB, 128 KiB) copied, 0.414327 s, 316 kB/s
Copying ROMMON environment
8388608+0 records in
8388608+0 records out
8388608 bytes (8.4 MB, 8.0 MiB) copied, 31.1199 s, 270 kB/s
131072+0 records in
131072+0 records out
131072 bytes (131 kB, 128 KiB) copied, 2.44015 s, 53.7 kB/s
131072+0 records in
131072+0 records out
131072 bytes (131 kB, 128 KiB) copied, 2.43394 s, 53.9 kB/s
ROMMON upgrade complete.
To make the new ROMMON permanent, you must restart the RP.
Device#reload
```

USB disablement

Explains how to disable and reenable USB ports using platform commands, notes the absence of system messages when USB devices are inserted while disabled, and clarifies that Bluetooth dongles are not affected by these commands.

USB ports are hardware interfaces on standalone or stacked devices that:

- can be disabled together using the `platform usb disable` command
- can be reenabled using the `no platform usb disable` command, and
- do not generate system messages when a USB device is inserted while the ports are disabled.

The `platform usb disable` command disables USB ports on the device, but it does not disable Bluetooth dongles connected to USB ports.

Disable USB (CLI)

Disable all USB ports on a standalone or stacked device using CLI commands to prevent the use of USB devices for security purposes.

Disable all USB ports on a standalone or stacked device.

Use this task when you want to prevent USB devices from being used on the device. When USB ports are disabled, the device does not generate system messages if a USB device is inserted.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Disable all the USB ports on the device.

Example

```
Device(config)# [no] platform usb disable
```

Use the **no platform usb disable** command to reenable the USB ports.

All USB ports on the device are disabled. The device will not recognize or generate system messages for connected USB peripherals.

6 Smart Licensing Using Policy

Topics:

- Smart Licensing Using Policy

Smart Licensing Using Policy

Smart Licensing Using Policy is a Cisco licensing system that is automatically enabled on controllers and upgrades, ensuring that your Smart Account and Virtual Account in Cisco Smart Software Manager are ready for policy-driven license management by default.

The Smart Licensing Using Policy feature is automatically enabled on the controller. This is also the case when you upgrade to this release. By default, your Smart Account and Virtual Account in Cisco Smart Software Manager (CSSM) are enabled for Smart Licensing Using Policy. For more information, see Smart Licensing Using Policy. To use Smart Licensing, you must first set up a Smart Account on Cisco License Central. For a more detailed overview on Cisco Licensing, visit Licensing Hub.

7 Management over Wireless

Topics:

- Management over wireless

Chapter wrapper for the map in vEWLC config guide.

Management over wireless

Management over Wireless details.

Management over wireless is a feature that allows you to monitor and configure the controller using client devices connected over Wi-Fi. This feature is helpful in situations where a wired connection is not available. The feature:

- allows you to monitor and configure the controller using client devices connected through Wi-Fi,
- is helpful in situations where a wired connection is not available, and
- is usually disabled by default for security reasons.

> **Attention**
>
> By default, the feature is disabled for security reasons. If disabled, the controller blocks any management access attempts from its connected wireless clients. This helps prevent someone connected over Wi-Fi from directly managing the controller they are using.

However, keeping this feature disabled does not block management access from wireless clients that are connected to other controllers in the network. To enforce stricter access control across all wireless clients, use Access Control Lists (ACLs) or similar mechanisms, based on VLANs or IP ranges.

Restrictions on management over wireless

Management over wireless comes with important restrictions regarding EWC and feature disablement.

- The feature does not work for Embedded Wireless Controller (EWC).
- By default, this feature is disabled for security reasons.

Configure management of controller over wireless network (GUI)

Enable Management over Wireless on the controller using GUI.

Enable management of the controller over wireless clients. This allows you to monitor and configure the controller using client devices connected over the wireless network.

Procedure

1. Choose **Configuration > Wireless > Wireless Global**.
2. Check the **Management via Wireless** check box.
3. Click **Apply**.

You can now manage the controller using client devices connected over the wireless networks.

Configure management of controller over wireless network (CLI)

Guides you through enabling or disabling controller management over the wireless network using the CLI, allowing monitoring and configuration from wireless client devices.

Enable management of the controller over wireless clients. This allows you to monitor and configure the controller using client devices connected over the wireless network.

Procedure

1. Enable management over wireless.

   Example

   ```
   Device(config)# wireless mgmt-via-wireless
   ```

   Use the **no** form of this command to disable management over wireless.

You can now manage the controller using client devices connected over the wireless network.

8 Boot Integrity Visibility

Topics:

- Boot integrity visibility

Boot integrity visibility

Boot integrity visibility allows the Cisco platform identity and software integrity information to be visible.

Boot integrity visibility is a security feature that

- makes Cisco platform identity and software integrity information visible,
- enables verification of the boot process through checksum evaluation, and
- assists in ensuring that only trusted code is booted on the platform.

Platform identity provides the platform's manufacturing installed identity. Software integrity exposes boot integrity measurements used to assess if trusted code has been booted.

During the boot process, the software creates a checksum record of each bootloader stage, which can be compared against a Cisco-certified record to verify the authenticity of the software image. Mismatched checksum values might indicate a non-Cisco certified or altered software image.

Feature history for boot integrity visibility

Boot Integrity Visibility feature history.

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 8: Feature History for Boot Integrity Visibility

Release	Feature Information
Cisco IOS XE 17.4.1	This feature was introduced.

Verify software image and hardware

Verify the integrity of the device's software image and hardware by retrieving and reviewing the checksum records created during switch bootup.

This task describes how to retrieve the checksum record that was created during a switch bootup.

Enter the following commands in privileged EXEC mode. The messages `% Error retrieving SUDI certificate` and `% Error retrieving integrity data` signify a real CLI failure.

> **Note**
>
> On executing the following commands, you might see the message `% Please Try After a Few Seconds` displayed on the CLI. This does not indicate a CLI failure, but indicates setting up of underlying infrastructure required to get the required output. We recommend waiting for a few minutes and then try the command again.

Procedure

1. View the checksum record for a specific SUDI

Example

```
Device# show platform sudi certificate sign nonce 123
```

Displays checksum record for the specific SUDI.

- (Optional) **sign** - Shows signature
- (Optional) **nonce** *nonce* - Enter a nonce value

2. View the checksum record for boot stages

Example

```
Device# show platform integrity sign nonce 123
```

Displays checksum record for boot stages.

- (Optional) **sign** - Shows signature
- (Optional) **nonce** *nonce* - Enter a nonce value

You obtain the checksum records for the device's software image and hardware, confirming their integrity or identifying any issues.

Verify platform identity and software integrity

Verify Platform Identity and Software Integrity.

Verify platform identity

The following example displays the Secure Unique Device Identity (SUDI) chain in PEM format. Each SUDI includes the Product ID (PID) and serial number (SN), so you can uniquely identify each device on a large network.

The first certificate is the Cisco Root CA 2048, and the second is the Cisco subordinate CA (ACT2 SUDI CA). You can verify both certificates by comparing them with those published on https://www.cisco.com/security/pki/. The third certificate is the SUDI certificate.

Important

The CLI outputs in this document are examples for your reference. Your output may differ depending on the configuration of your device.

```
Device# show platform sudi certificate sign nonce 123
-----BEGIN CERTIFICATE-----
MIIDQzCCAiugAwIBAgIQX/h7KCtU3I1CoxW1aMmt/zANBgkqhkiG9w0BAQUFADA1
MRYwFAYDVQQKEw1DaXNjbyBTeXN0ZW1zMRswGQYDVQQDExJDaXNjbyBSb290IENB
IDIwNDgwHhcNMDQwNTE0MjAxNzEyWhcNMjkwNTE0MjAyNTQyWjA1MRYwFAYDVQQK
Ew1DaXNjbyBTeXN0ZW1zMRswGQYDVQQDExJDaXNjbyBSb290IENBIDIwNDgwggEg
MA0GCSqGSIb3DQEBAQUAA4IBDQAwggEIAoIBAQCwmrmrp68Kd6ficba0ZmKUeIhH
xmJVhEAyv8CrLqUccda8bnuoqrpu0hWISEWdovyD0My5jOAmaHBKeN8hF570YQXJ
FcjPFto1YYmUQ6iEqDGYeJu5Tm8sUxJszR2tKyS7McQr/4NEb7Y9JHcJ6r8qqB9q
VvYgDxFUl4F1pyXOWWqCZe+36ufijXWLbvLdT6ZeYpzPEApk0E5tzivMW/VgpSdH
```

```
jWn0f84bcN5wGyDWbs2mAag8EtKpP6BrXruOIIt6keO1aO6g58QBdKhTCytKmg9l
Eg6CTY5j/e/rmxrbU6YTYK/CfdfHbBcl1HP7R2RQgYCUTOG/rksc35LtLgXfAgED
o1EwTzALBgNVHQ8EBAMCAYYwDwYDVR0TAQH/BAUwAwEB/zAdBgNVHQ4EFgQUJ/PI
FR5umgIJFq0roIlgX9p7L6owEAYJKwYBBAGCNxUBBAMCAQAwDQYJKoZIhvcNAQEF
BQADggEBAJ2dhISjQal8dwy3U8pORFBi71R803UXHOjgxkhLtv5MOhmBVrBW7hmW
Yqpao2TB9k5UM8Z3/sUcuuVdJcr18JOagxEu5sv4dEX+5wW4q+ffy0vhN4TauYuX
cB7w4ovXsNgOnbFp1iqRe6lJT37mjpXYgyc81WhJDtSd9i7rp77rMKSsH0T8lasz
Bvt9YAretIpjsJyp8qS5UwGH0GikJ3+r/+n6yUA4iGe0OcaEb1fJU9u6ju7AQ7L4
CYNu/2bPPu8Xs1gYJQk0XuPL1hS27PKSb3TkL4Eq1ZKR4OCXPDJoBYVL0fdX4lId
kxpUnwVwwEpxYB5DC2Ae/qPOgRnhCzU=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIEPDCCAySgAwIBAgIKYQlufQAAAAADDANBgkqhkiG9w0BAQUFADA1MRYwFAYD
VQQKEw1DaXNjbyBTeXN0ZW1zMRswGQYDVQQDExJDaXNjbyBSb290IENBIDIwNDgw
HhcNMTEwNjMwMTc1NjU3WhcNMjkwNTE0MjAyNTQyWjAnMQ4wDAYDVQQKEwVDaXNj
bzEVMBMGA1UEAxMMQUNUMiBTVURJIENBMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8A
MIIBCgKCAQEA0m5l3THIxA9tN/hS5qR/6UZRpdd+9aE2JbFkNjht6gfHKd477AkS
5XAtUs5oxDYVt/zEbslZq3+LR6qrqKKQVu6JYvH05UYLBqCj38s76NLk53905Wzp
9pRcmRCPuX+a6tHF/qRuOiJ44mdeDYZo3qPCpxzprWJDPclM4iYKHumMQMqmgmg+
xghHIooWS80BOcdiynEbeP5rZ7qRuewKMpl1TiI3WdBNjZjnpfjg66F+P4SaDkGb
BXdGj13oVeF+EyFWLrFjj97fL2+8oauV43Qrvnf3d/GfqXj7ew+z/sXlXtEOjSXJ
URsyMEj53Rdd9tJwHky8neapszS+r+kdVQIDAQABo4IBWjCCAVYwCwYDVR0PBAQD
AgHGMB0GA1UdDgQWBBRI2PHxwnDVW7t8cwmTr7i4MAP4fzAfBgNVHSMEGDAWgBQn
88gVHm6aAgkWrSugiWBf2nsvqjBDBgNVHR8EPDA6MDigNqA0hjJodHRwOi8vd3d3
LmNpc2NvLmNvbS9zZWN1cml0eS9wa2kvY3JsL2NyY2EyMDQ4LmNybDBQBggrBgEF
BQcBAQREMEIwQAYIKwYBBQUHMAKGNGh0dHA6Ly93d3cuY2lzY28uY29tL3NlY3Vy
aXR5L3BraS9jZXJ0cy9jcmNhMjA0OC5jZXIwXAYDVR0gBFUwUzBRBgorBgEEAQkV
AQwAMEMwQQYIKwYBBQUHAgEWNWh0dHA6Ly93d3cuY2lzY28uY29tL3NlY3VyaXR5
L3BraS9wb2xpY2llcy9pbmRleC5odG1sMBIGA1UdEwEB/wQIMAYBAf8CAQAwDQYJ
KoZIhvcNAQEFBQADggEBAGh1qclr9tx4hzWgDERm371yeuEmqcIfi9b9+GbMSJbi
ZHc/CcCl0lJu0a9zTXA9w47H9/t6leduGxb4WeLxcwCiUgvFtCa51Iklt8nNbcKY
/4dw1ex+7amATUQO4QggIE67wVIPu6bgAE3Ja/nRS3xKYSnj8H5TehimBSv6TECi
i5jUhOWryAK4dVo8hCjkjEkzu3ufBTJapnv89g9OE+H3VKM4L+/KdkUO+52djFKn
hyl47d7cZR4DY4LIuFM2P1As8YyjzoNpK/urSRI14WdIlplR1nH7KNDl5618yfVP
0IFJZBGrooCRBjOSwFv8cpWCbmWdPaCQT2nwIjTfY8c=
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
MIIDfTCCAmWgAwIBAgIEAwQD7zANBgkqhkiG9w0BAQsFADAnMQ4wDAYDVQQKEwVD
aXNjbzEVMBMGA1UEAxMMQUNUMiBTVURJIENBMB4XDTE4MDkyMzIyMzIwNloXDTI5
MDUxNDIwMjU0MVowaTEnMCUGA1UEBRMeUElEOkM5NjAwLVNVUC0xIFNOOkNBVDIy
MzZMMFE5MQ4wDAYDVQQKEwVDaXNjbzEYMBYGA1UECxMPQUNULTIgTGl0ZSBTVURJ
MRQwEgYDVQQDEwtDOTYwMC1TVVAtMTCCASIwDQYJKoZIhvcNAQEBBQADggEPADCC
AQoCggEBANsh0jcvgh1pdOjP9KnffDnDc/zEHDzbCTWPJi2FZcsaSE5jvq6CUqc4
MYpNAZU2Jym7NSD8iQbMXwbnCtoL64QtxQeFhRYmc4d5o933M7GwpEH0I7HUSbO/
Fxyp7JBmGPPgAkY7rKsYENiNK2hiR7Q2O7X2BidOKknEuofWdJMNyMaZgLYLOHbJ
5oXaORxhUy3VRaxNl6qI7kYxuugg2LcAbZ539sRXe8JtHyK8llURNSGMiQ0S17pS
idGmrJJ0pEHA0EUVTZqEny3z+NW9uxLVSzu6+hEJYlqfI+YEf0DbVZly1cy5r/jF
yNdGuGKvd5agvgCly8aYMZa3P+D5S8sCAwEAAaNvMG0wDgYDVR0PAQH/BAQDAgXg
MAwGA1UdEwEB/wQCMAAwTQYDVR0RBEYwRKBCBgkrBgEEAQkVAgOgNRMzQ2hpcElE
PVUxUk5TVEl3TVRjd05qSTFBQUFwZndBQUFBQUFBQUFBQUFBQUFBQUhtSlU9MA0G
CSqGSIb3DQEBCwUAA4IBAQCrpHo/CUyk5Hs/asIcYW0ep8KocSkbNh8qamyd4oWD
e/MGJW9Bs5f09IEbILWPdytCCS2lSyJbxz2HvVDzdxQdxjDwUNiWuu3dWMXN/i67
yuCGM+lA1AAG5dT6lNgWYHh+YzsZm9eoq1+4NM+JuMXWsnzAK8rSy+dSpBxqFsBq
E0OlPsaK7y2h8gs+XrV9x+D48OZQkTRXpxhJfiWvs+EbdgsAM/vBxTAoTJPVmXWN
Cmcj9X52Xl3i4MdOUXocZLO2kh6JSgOYGkFeZifJ0iDvMfAf0cJ6+cEF6bSxAqBL
veel+8LmeiE/2O9h6qGHPPDacCaXA2oJCDHveAt8iPTG
-----END CERTIFICATE-----

Signature version: 1
Signature:
```

The optional RSA 2048 signature covers the three certificates, the signature version, and the user-provided nonce.

```
RSA PKCS#1v1.5 Sign {<Nonce (UINT64)> || <Signature Version (UINT32)> || <Cisco
 Root CA 2048 cert (DER)> ||
<Cisco subordinate CA (DER)> || <SUDI certificate (DER)>}
```

Cisco management solutions interpret this output. You can also use a simple script with OpenSSL commands to display platform identity and verify the signature, which helps confirm the Cisco unique device identity..

```
[linux-host:~]openssl x509 -in sudi_id.pem -subject -noout
subject= /serialNumber=PID:C9600-SUP-1
SN:CAT2239L06B/CN=C9600-SUP-1-70b3171eaa00
```

Verify software integrity

The following example shows the checksum record for the boot stages. The hash measurements appear for each of the three software stages when they boot in sequence. Compare these hashes to the Cisco-provided reference values.

Signing the output ensures it is genuine and unaltered. Provide a nonce to protect against replay attacks.

Note

Boot integrity hashes are not MD5 hashes. For example, if you run **verify /md5 cat9k_iosxe.16.10.01.SPA.bin** command for the bundle file, the hash will not match.

The following is a sample output of the **show platform integrity sign nonce 123** command. This output includes measurements of each installed package file.

```
Device# show platform integrity sign nonce 123
Platform: C9800-L-F-K9
Boot 0 Version: R04.1173930452019-06-11
Boot 0 Hash: A6C92C44976FC77DD42234444FFD87798FB9036A2762FAA4999A190A0258B18C
Boot Loader Version: 16.12(1r)
Boot Loader Hash:
FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
OS Version: 2020-03-19_20.26
OS Hashes:
C9800-L-universalk9_wlc.2020-03-19_20.26.SSA.bin:
53E2DF1A1A082E36EA4CAB817C1794EC9D69AC0E90BCCBFEECF9B0D0BCA9385AA9E9372ABF7431E4A08FC5E5B9670131C09D158E5B8A7B457501EE77AB9F1C26D
C9800-L-mono-universalk9_wlc.2020-03-19_20.26.SSA.pkg:
1D3279D53B0311CE42C669824DF86FB5596CD7CA45CA8D7FDC3D10657B8C9A48F4B0508D7BCFFD645CB6571AC1E674A57A82414E3D6E1666BE64E6132F707671
PCR0: EE14A2D5099DA343B3941C54A429C4AC1D3EE8E9B609F1AC00049768A470734E
PCR8: 78794D0F5667F8FA4E425E3CA2AF3CD99B90B219FD90222D622B3D563416BBAA
```

Note

Only OS and package hashes are supported.

9 SUDI99 Certificate Support

Topics:

- SUDI certificates

SUDI certificates

Explains what SUDI certificates are, their role in secure authentication, and the enhancements brought by SUDI99 certificate support.

A Secure Unique Device Identity (SUDI) is a device credential that

- provides device-specific authentication during secure connection handshakes
- is provisioned in a secure hardware chip during manufacturing, and
- supports multiple certificates for interoperability with diverse network environments.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 9: Feature history for SUDI99 certificate support

Feature Name	Release Information	Feature Description
SUDI99 certificate support	Cisco IOS XE Dublin 17.7.1	This feature allows a network administrator to use SUDI99 certificates for authentication during secure connection handshakes enhancing security through device-specific credentials

SUDI certificates validate device identity, ensuring only genuine devices connect to Cisco networks. They play a crucial role in maintaining the security of network infrastructure.

Note

Some existing SUDI certificates, such as those used in controller and AP platforms, are set to expire in May 2029. To ensure continued secure authentication, the SUDI refresh program migrates SW-SUDI SHA1 certificates to the new SUDI99 certificates with extended validity.

Software selection for SUDI trustpoint certificates

Cisco IOS XE software supports two slots for initializing SUDI certificates from the secure hardware chip. This SUDI99 migration change will rearrange certificate-to-trustpoint mapping as follows:

Table 10: Existing software selection for SUDI trustpoint certificates

Trustpoint Name	Software Selection Among Programmed Certificate Chains
CISCO_IDEVID_SUDI	CMCA2 SHA2 SUDI (SHA2-2037)
CISCO_IDEVID_SUDI_LEGACY	CMCA SHA1 SUDI

Table 11: New software selection for SUDI trustpoint certificates

Trustpoint Name	Software Selection Among Programmed Certificate Chains
CISCO_IDEVID_SUDI	CMCA-III SHA2 SUDI99
CISCO_IDEVID_SUDI_LEGACY	CMCA2 SHA2 SUDI (SHA2-2037)

> **Caution**
>
> Avoid using expired certificates for device authentication to prevent service disruption.

SUDI99 certificate and software support

The table lists the SUDI99 certificate and software support:

Table 12: SUDI99 certificate and software support

Cisco Catalyst 9800 Controllers	SUDI99 Certificate Support	Software Support for SUDI99 Migration
Cisco Catalyst 9800-CL Wireless Controller for Cloud	Not supported.	–
Cisco Catalyst 9800 Series Wireless Controllers • 9800-40 • 9800-80 • 9800-L	Supported	Yes. From Cisco IOS XE Cupertino 17.7.1.
Cisco Embedded Wireless Controller on Catalyst Access Points. • 9105AXI • 9115AXI • 9115AXE • 9117AXI • 9120AXI • 9120AXE • 9120AXP • 9130AXI • 9130AXE	Supported	Yes. From Cisco IOS XE Cupertino 17.7.1.

Cisco Catalyst 9800 Controllers	SUDI99 Certificate Support	Software Support for SUDI99 Migration
Cisco Embedded Wireless Controller on Catalyst Switches • 9300 Series • 9400 Series • 9500 Series • 9500H Series	Not supported.	–

Backward compatibility

Backward compatibility refers to the Cisco Catalyst 9800 Series Wireless Controllers maintaining their functionality with legacy systems by using existing certificates for devices that cannot validate SUDI99.

- If your device (AP or controller) cannot validate the SUDI99 certificate, the controller switches to an older certificate (SHA2-2037) for the connection.
- For NMSP-TLS connections with Cisco CMX, the client certificate is not validated in default security mode. However, in FIPS mode, Cisco CMX validates the controller certificate.
- In FIPS mode, install the new SUDI CA certificates on the earlier version of Cisco CMX, or upgrade to the latest version.

Important

Some applications, such as HTTPS, RADSEC, and WebAuth, do not use the SUDI certificate as their default trustpoint. However, you can configure them to use the SUDI trustpoint explicitly. The SUDI refresh program alters the certificate selection for such services. However, there is no functional impact.

Restrictions of SUDI99 certificates

- Incorrectly programmed SUDI99 certificates are rejected during trustpoint initialization at bootup, reverting to previous mappings.
- We recommend ensuring accurate certificate programming to avoid service disruptions during migration.
- User can verify the SUDI certificate status using the **show platform sudi pki** command.

Disable SUDI99 migration (GUI)

Disable SUDI99 migration using GUI.

SHA1 SUDI certificates on hardware controllers have an imminent expiry date and devices using expired certificates face disruption in service.

Controllers come programmed with newer certificates in their secure hardware chip, ensuring a smooth migration to the latest SUDI99 certificate issued by CMCA-III authority. These certificates are enabled by default and are valid till December 2099.

Follow this procedure if you wish to continue using the current certificate:

Procedure

1. Choose **Configuration** > **Security** > **PKI Management** > **Trustpoints**, go to the **SUDI Status** section.
2. Disable the **Cisco Manufacturing CA III certificate** to continue using the older certificate that is mapped to an existing Trustpoint.
3. Click **Apply**.

What to do next

Reload the device for the configuration to take effect.

Disable SUDI99 migration (CLI)

Disable SUDI99 migration using CLI.

The SUDI99 certificate is set as the default trustpoint in supported hardware units.

Follow the below steps to disable it:

Procedure

1. Form the high availability pair in HA deployments.

   Example

   ```
   no platform sudi cmca3
   ```

2. Save the configuration.
3. Reload the controller to disable the SUDI certificate and fall back to the older trustpoint certificate.

What to do next

To check the certificate validation status, use the **show platform sudi pki** command.

10 Link Aggregation Group

Topics:

- Link aggregation group
- Port aggregation protocol
- Multi-LAG
- Configure link aggregation group

Link aggregation group

Explains how link aggregation groups (LAGs) combine controller ports to improve redundancy, simplify configuration, and support load balancing.

A link aggregation group is a port-channeling technology that

- combines all of a controller's distribution system ports into a single 802.3ad port channel
- reduces the number of IP addresses required for port configuration, and
- dynamically manages port redundancy and load-balances AP transparently to the corresponding user.

LAG simplifies controller configuration by eliminating the need to configure ports for each interface. If any controller port fails, traffic migrates automatically to other functioning ports. As long as at least one controller port is functioning, the system continues to operate. Wireless clients can continue to send and receive data, and APs remain connected to the network.

The wireless management VLAN can be assigned to only one port channel.

LACP is supported on a standalone controller from the Cisco IOS XE Gibraltar 16.12.x release. LACP is supported on an SSO pair from Cisco IOS XE Amsterdam 17.1.1s and later.

If a dual MAC entry is generated in the switchport MAC table when using ISE licensing enforcement, it may consume two licenses for a single AP. To avoid this issue when only one port is used, choose wired0.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 13: Feature history for link aggregation group

Feature Name	Release Information	Feature Description
Link aggregation group	Cisco IOS XE Amsterdam 17.2.1	This feature provides flexibility for connecting the controller to multiple switching infrastructures. Using multi-LAG, you can connect multiple uplinks from the controller to separate uplink switches.

Link aggregation control protocol

LACP is a part of an IEEE specification (802.3ad) that allows you to bundle several physical ports together to form a single logical channel.

A link aggregation control protocol (LACP) is a network protocol that:

- Bundles several physical ports together to form a single logical channel.
- Negotiates automatic bundling by sending LACP packets to a peer.
- Maintains network operation by dynamically grouping and rerouting traffic into a single logical link (channel or aggregate port) if any ports fail.

Link Aggregation Control Protocol (LACP) is a part of the IEEE 802.3ad specification. If any port in the bundle fails, traffic shifts to the remaining working ports, ensuring uninterrupted connectivity. APs and wireless clients can continue to send and receive data.

LACP configuration options for LAG

LACP configuration options for LAG.

To configure LAG using LACP, create multiple port-channel interfaces. Then, add each interface to the corresponding port bundle. Configure LACP on the uplink switch. This configuration allows the LACP bundle to become active.

Steps to configure LAG using LACP:

- Create a port-channel interface
- Add interface to the port-channel
- Add VLAN to LAG
- Add interface to the port-channel

Port aggregation protocol

Explains how port aggregation protocols automate and manage the bundling of Ethernet ports to increase throughput and provide redundancy.

A port aggregation protocol is a Cisco-proprietary network protocol that facilitates the automatic combination of EtherChannels by exchanging PAgP packets between Ethernet ports. PAgP packets are sent between Fast EtherChannel-capable ports in order to form a channel.

When any of the active ports fail, a standby port becomes active. By using PAgP, the controller learns the identity of partners that support PAgP and the capabilities of each port.

PAgP then dynamically groups similarly configured ports (on a single device in a stack) into a single logical link (channel or aggregate port). Similarly configured ports are grouped based on hardware, administrative, and port parameter constraints.

PAgP LAG configuration attributes

PAgP LAG configuration attribute details.

To configure LAG using PAgP, create multiple port-channel interfaces and add them to the corresponding port bundle. Configure PAgP on the uplink switch so that the PAgP bundle can become active.

Follow these steps to configure LAG using PAgP:

- Create a port-channel interface
- Add interface to the port-channel

Port channel interface number

Port channel interface number details.

A port channel interface number is a configuration identifier that uniquely distinguishes a port channel logical interface. It defines allowable ranges and limits depending on the controller model.

From Cisco IOS XE Bengaluru 17.5.1 onwards, port channel interface numbers from one to 64 are supported, with model-dependent limits on the following Cisco Catalyst 9800 Series Wireless Controllers:

- Cisco Catalyst 9800-CL Wireless Controller for Cloud: The available range on the CLI is one to 64. The maximum supported port channel interfaces are 64.
- Cisco Catalyst 9800-L Wireless Controller: The available range on the CLI is one to 64. The maximum supported port channel interfaces are 14.
- Cisco Catalyst 9800-40 Wireless Controller: The available range on the CLI is one to 64. The maximum supported port channel interfaces are 16.
- Cisco Catalyst 9800-80 Wireless Controller: The available range on the CLI is one to 64. The maximum supported port channel interfaces are 64.

For example on the Cisco Catalyst 9800-L Wireless Controller, port-channel interface numbers can be anywhere between one and 64, provided that no more than 14 port-channel interfaces are configured.

If 16 port-channel interfaces have been configured on the Cisco Catalyst 9800-40 Wireless Controller, and the system has reached its limit, an error message appears when you attempt to configure a 17th port-channel interface:

```
Device(config)#
                Dec 15 08:58:22.209 CST: %ETC-5-CANNOT_ALLOCATE_AGGREGATOR:
Aggregator limit reached, cannot allocate aggregator for group 17
```

When you downgrade from Cisco IOS XE Bengaluru 17.5.1 to an earlier version, if port channels are configured beyond the supported range of the earlier version, errors appear upon startup. The unsupported port channels disappear after the downgrade is complete.

```
interface Port-channel29
                ^% Invalid input detected at '^' marker.
                interface Port-channel35
                ^% Invalid input detected at '^' marker.
```

Note that the HA pairing remains intact after downgrade.

Flow control

A flow control mechanism is a network protocol feature that:

- Enables connected Ethernet ports to control traffic rates during congestion by allowing congested nodes to pause link operation at the other end.
- Uses pause frames for signaling congestion and allows congested nodes to pause traffic from other nodes to prevent packet loss.

Flow control is enabled by default for both bay0 and bay1 ports as a first-day behavior. If you see overruns on a Cisco Catalyst 9800-L Wireless Controller due to burst traffic, enable flow control on the uplink switch. This change helps the uplink switch honor the flow control settings enabled on the controller.

Although the show command might indicate that flow control is off or unsupported in the controller's CLI, the actual default behavior is that flow control is enabled. For more information, check https://bst.cloudapps.cisco.com/bugsearch/bug/CSCwk52366.

LAG configuration in ON mode

LAG configuration in ON mode.

To configure LAG in ON mode, create multiple port-channel interfaces and add them to the corresponding port bundle. Configure LACP on the uplink switch so that the LACP bundle becomes active.

- Configure LAG in ON mode (CLI) on page 184

Multi-LAG

Explains how Multi-LAGs provide high availability and increased bandwidth by aggregating multiple network links across different devices.

Multichassis link aggregation group

Multichassis link aggregation group provides flexibility in connecting the controller to a switch's infrastructure.

A multichassis link aggregation group (multi-LAG) is a switch interconnection technology that:

- Connects multiple controller uplinks to separate uplink switches for redundancy and load balancing.
- Supports VLAN-based traffic splitting when connected to a multiswitch topology.
- Provides the ability to distribute traffic on different uplinks, based on VLANs. For example, guest traffic can be completely isolated to a different switch or network from the enterprise network. You cannot configure the same VLAN on both uplinks.

Starting with Cisco IOS XE Amsterdam 17.2.1, the multichassis link aggregation group (multi-LAG) feature, which provides flexibility in connecting the controller to a switch's infrastructure, is supported.

You can connect a LAG to a single switch. However, different VLANs is connected to different LAGs. The redundancy port is connected to the same distribution switch as the uplinks or directly connected in a back-to-back configuration.

Multi-LAG is supported in LAG ON mode, LACP, and PAgP modes.

Prerequisites for Multi-LAG

Prerequisites for Multi-LAG.

To ensure optimal performance and avoid configuration conflicts:

- Connect each LAG to a single switch.
- Assign different VLANs to different LAGs.

Restrictions for Multi-LAG

Multi-LAG restrictions.

- Automatic failover from the primary LAG to the secondary LAG is not supported if the primary LAG fails.
- The interface on the controller does not come up when you shut or unshut the associated switch port.

> **Note**
>
> This restrictions is specific to Cisco Catalyst 9800-CL Cloud Wireless Controllers deployed in KVM environments using SR-IOV.

Supported topologies

Supported topologies details.

Supported topologies are network deployment configurations. They define how Cisco Catalyst 9800 series controllers connect with the network through their available ports and set limitations regarding mixing port types and minimum group sizes for link aggregation.

The Cisco Catalyst 9800-80 Wireless Controller has eight ports, while the Cisco Catalyst 9800-40 and Cisco Catalyst 9800-L wireless controllers have four ports each. Multi-LAGs can only include ports with matching capabilities, such as 2.5 G with 2.5 G, or 10 G with 10 G ports. You cannot include both a 2.5 G port and a 10 G port in a port channel group. Each group must have a minimum of two ports in one LAG.

Figure 2: Single Controller with Multi-LAG

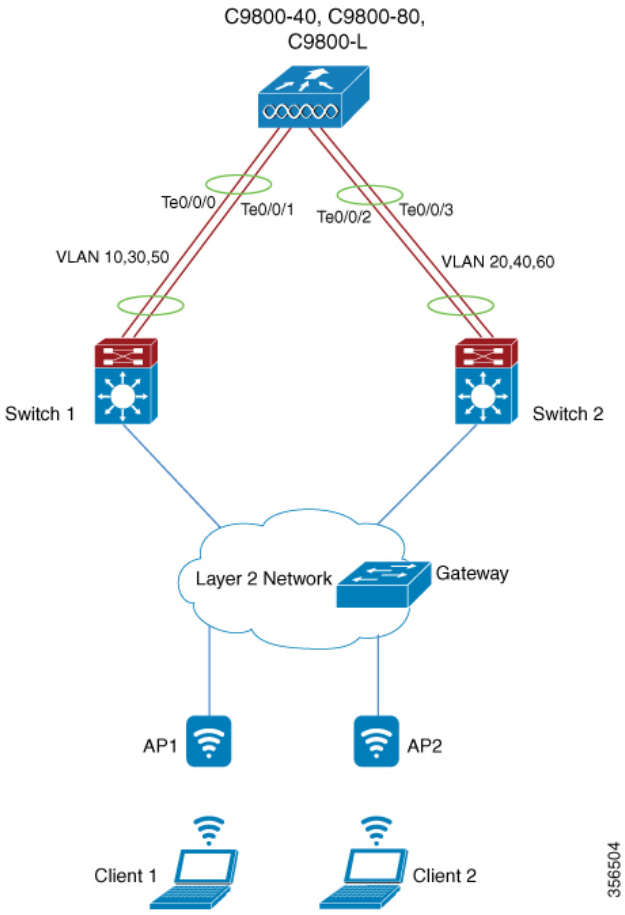
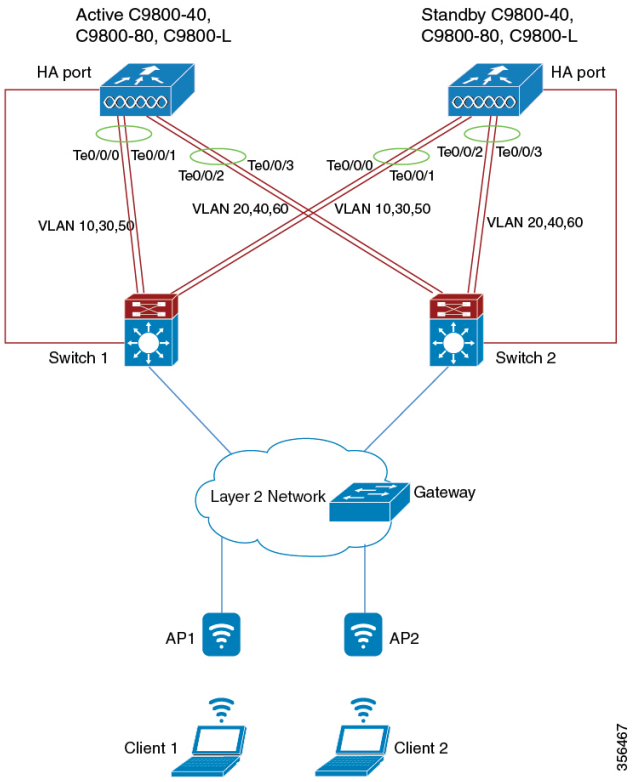


Figure 3: SSO Pair with Multi-LAG



Configure link aggregation group

Configure a link aggregation group to increase bandwidth and provide redundancy by combining multiple physical links.

Configure a port channel interface (GUI)

Configure a port channel interface using GUI.

Add and configure a logical port channel interface using the device GUI.

Procedure

1. Choose **Configuration** > **Interface** > **Logical**.
2. Click the **Port Channel** tab to configure the Port Channel interface.

   The **Port Channel** tab lists all the logical port-channel interfaces on the device.
3. Click **Add** to create a new logical port channel interface.

   The **Add Port Channel Interface** window is displayed.
4. In the **Add Port Channel Interface** window, complete the following procedure:
   a) In the **Port Channel Number** field, enter the port channel number. The valid values are between one to 64.
   b) In the **Description** field, enter the port channel description.
   c) Click the **Admin Status** toggle button to set the admin status as *UP* or *DOWN*.
   d) Click the **Enable Layer 3 Address** toggle button to enable the Layer 3 address.
   e) In the Port Members section, select the port members from the **Available** list box and add them to the **Associated** list.
   f) Choose a switch mode for the interface from the **Switchport Mode** drop-down list.

- If you select **access** as the switch mode, enter the access VLAN ID in the **Access VLAN** field.
- If you select **trunk** as the switch mode, enter the VLAN IDs that you want to assign as trunk links. To allow all VLAN IDs as trunk links, set **Allowed VLANs** to **All**. Specify a native VLAN.
- If you choose **dynamic auto** or **dynamic desirable** as the switch mode, enter the access VLAN ID. Enter the VLAN IDs you want to assign as trunk links. To allow all VLAN IDs as trunk links, set the **Allowed VLANs** to All. Specify a native VLAN.

g) Click **Update & Apply to Device**.

Create a port-channel interface (CLI)

Create a port-channel interface using commands.

Configure a port-channel interface to aggregate multiple network interfaces for increased bandwidth or redundancy.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the port channel and enter interface configuration mode.

   Example

   ```
   Device(config)# interface port-channel port-channel 2
   ```

   The valid values for the port channel number ranges from one to 64.

3. Configure the port as trunk.

   Example

   ```
   Device(config-if)# switchport mode trunk
   ```

4. Enable the interface.

   Example

   ```
   Device(config-if)# no shutdown
   ```

Configure LAG in ON mode (CLI)

Configure LAG in ON mode using commands.

Group multiple physical interfaces for increased bandwidth and redundancy using LAG in ON mode.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the port.

   Example

   ```
   Device(config)# interface TenGigabitEthernet port-slot 0/0/0
   ```

3. Configure the port as trunk.

   Example

   ```
   Device(config-if)# switchport mode trunk
   ```

4. Disable the interface.

   Example

   ```
   Device(config-if)# no shutdown
   ```

5. Assign the port to a channel group, and specify the ON mode.

   Example

   ```
   Device(config-if)# channel-group group-number 3 mode on
   ```

   The valid values for the port channel number ranges from 1 to 64.

6. Assign the allowed VLAN ID to the port when it is in trunking mode.

   Example

   ```
   Device(config-if)# switchport trunk allowed vlan vlan-id 16,17
   ```

Add an interface to a port channel (LACP) (CLI)

Add and configure the interface to a port channel (LACP) using commands.

Assign an Ethernet interface to a port channel using LACP mode to aggregate bandwidth and provide redundancy.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the port.

   Example

   ```
   Device(config)# interface TenGigabitEthernet port-slot 0/0/0
   ```

3. Assign the port to a channel group, and specify the LACP mode.

   Example

   ```
   Device(config-if)# channel-group group-number 1 mode {active | passive}
   ```

   The valid values for the port channel number ranges from 1 to 64.

4. Configure the port as trunk.

   Example

   ```
   Device(config-if)# switchport mode trunk
   ```

Add an interface to a port channel (PAgP) (CLI)

Add and configure the interface to a port channel (PAgP) using commands.

Add and configure an interface as a member of a port channel using PAgP.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the TenGigabit Ethernet interface.

   Example

   ```
   Device(config)# interface TenGigabitEthernet port-slot 0/0/0
   ```

3. Assign the port to a channel group, and specify the PAgP mode.

   Example

   ```
   Device(config-if)# channel-group group-number 1 mode {auto | desirable}
   ```

   The valid values for the port channel number ranges from one to 64.

4. Configure the port as trunk.

Example

```
Device(config-if)# switchport mode trunk
```

Add a VLAN to a port channel (CLI)

Add different VLANs under a port channel using commands.

Add one or more VLANs to the list of allowed VLANs on a port channel.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the port channel.

Example

```
Device(config)# interface port-channel port-channel 1
```

The valid values for the port channel number range from one to 64.

3. Add VLANs to the list of allowed VLANs.

Example

```
Device(config-if)# switchport trunk allowed vlan vlan-id 10,30,50
```

Remove a port channel group from a physical interface (CLI)

Remove a port channel group from a physical port using commands.

Remove a port channel group binding from a specified physical interface to release it from port channel configuration.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Enter the TenGigabit Ethernet interface.

Example

```
Device(config)# interface TenGigabitEthernet port-slot 0/0/0
```

3. Remove the port channel group from the physical port.

Example

```
Device(config-if)# no channel-group
```

4. Exit the interface configuration mode.

Example

```
Device(config-if)# end
```

Verify the LAG configuration

Verify a port channel's state using commands.

To view a port channel's state, use the following command:

```
Device# show etherchannel summary

                                Flags:  D - down        P - bundled in
port-channel
                                I - stand-alone s - suspended
                                H - Hot-standby (LACP only)
                                R - Layer3      S - Layer2
                                U - in use      f - failed to allocate
aggregator

                                M - not in use, minimum links not met
                                u - unsuitable for bundling
                                w - waiting to be aggregated
                                d - default port
                                A - formed by Auto LAG
                                Number of channel-groups in use: 1
                                Number of aggregators:           1
                                Group  Port-channel  Protocol    Ports

------+-------------+-----------+-----------------------------------------------

                                3      Po3(SU)         LACP
Tw0/0/0(P)      Tw0/0/1(P)
                                4      Po4(SU)         LACP
Tw0/0/2(P)      Tw0/0/3(P)
```

To verify an LACP or PAgP configuration, use the following commands:

```
Device# show running-config interface tenGigabitEthernet 0/0/0

                                Building configuration...
```

```
                                Current configuration : 114 bytes
                                !
                                interface TwoGigabitEthernet0/0/0
                                switchport trunk allowed vlan 16,17
                                switchport mode trunk
                                speed 1000
                                no negotiation auto
                                no snmp trap link-status
                                channel-group 3 mode on
```

```
Device# show running-config interface port-channel 1

                                Building configuration...

                                Current configuration : 54 bytes
                                !
                                interface Port-channel1
                                switchport mode trunk
                                switchport trunk allowed vlan 10,30,50
                                end
```

11 Reload Reason History

Topics:

- Reload reason history

Introduces the reload reason history feature for tracking device restart events and guides users through retrieving and verifying reload information using YANG data models.

Reload reason history

Describes a diagnostic and serviceability feature that records controller reload events and maintains historical data for troubleshooting purposes.

Reload reason history is a diagnostic and serviceability feature that

- records the reasons for controller reloads
- maintains a log of the last ten reload events, and
- provides access to this data through CLI commands or programmable interfaces like NETCONF/YANG.

This feature is useful for troubleshooting abnormal reboots and understanding system behavior in production environments. Cisco introduced it in IOS XE Dublin 17.11.1, enhancing previous capabilities that only recorded the last reload reason.

View reload reason history

View the history using the **show version** command and the Network Configuration Protocol (NETCONF).

> **Note**
>
> When you reload the standby controller, the system report files are generated in the controller hard disk.

Feature history

Lists feature history for reload reason history.

This reference provides the feature history for reload reason history functionality.

Table 14: Feature history for reload reason history

Feature name	Release information	Feature description
Reload reason history	Cisco IOS XE 17.11.1	The reload reason history feature tracks the reasons for controller reload. This is done for the last 10 reloads. In Cisco IOS-XE 17.10.x and earlier releases, it was possible to track only the reason for the last reload.

Request reload reason history using YANG

Configure YANG with NETCONF and RESTCONF to request reload reason history for automated and programmable network operations.

Use YANG with NETCONF and RESTCONF to provide the desired solution for automated and programmable network operations.

Procedure

Use RPC to create a NETCONF GET request for reload history data.

Example

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
message-id="urn:uuid:da15955f-5bb7-437c-aeb5-0fc7901a1e9e">
  <nc:get>
    <nc:filter>
      <device-hardware-data
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-device-hardware-oper">
        <device-hardware>
          <device-system-data>
            <reload-history/>
          </device-system-data>
        </device-hardware>
      </device-hardware-data>
    </nc:filter>
  </nc:get>
</nc:rpc>

<rpc-reply message-id="urn:uuid:da15955f-5bb7-437c-aeb5-0fc7901a1e9e"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <device-hardware-data
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-device-hardware-oper">
      <device-hardware>
        <device-system-data>
          <reload-history>
            <rl-history>
              <reload-category>rc-rld</reload-category>
              <reload-desc>Reload Command</reload-desc>
              <reload-time>2022-11-30T01:33:44+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-crit-proc-fault</reload-category>
               <reload-desc>Critical process stack_mgr fault on rp_0_0 (rc=137),
 system report at
bootflash:core/Yang_Test-system-report_20221130-012929-UTC.tar.gz</reload-desc>

              <reload-time>2022-11-30T01:31:11+00:00</reload-time>
              <reload-severity>abnormal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-img-install</reload-category>
              <reload-desc>Image Install </reload-desc>
              <reload-time>2022-11-30T01:25:03+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-crit-proc-fault</reload-category>
              <reload-desc>Critical process rif_mgr fault on rp_0_0 (rc=137),
 system report at
bootflash:core/Yang_Test-system-report_20221130-011127-UTC.tar.gz</reload-desc>

              <reload-time>2022-11-30T01:13:08+00:00</reload-time>
              <reload-severity>abnormal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-rld</reload-category>
              <reload-desc>Reload Command</reload-desc>
              <reload-time>2022-11-30T01:08:26+00:00</reload-time>
              <reload-severity>normal</reload-severity>
```

```
            </rl-history>
            <rl-history>
              <reload-category>rc-crit-proc-fault</reload-category>
              <reload-desc>Critical process wncmgrd fault on rp_0_0 (rc=137),
 system report at
bootflash:core/Yang_Test-system-report_20221130-010338-UTC.tar.gz</reload-desc>

              <reload-time>2022-11-30T01:05:23+00:00</reload-time>
              <reload-severity>abnormal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-rld</reload-category>
              <reload-desc>Reload Command</reload-desc>
              <reload-time>2022-11-30T01:01:09+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-rld</reload-category>
              <reload-desc>Reload Command</reload-desc>
              <reload-time>2022-11-30T00:57:27+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-rld</reload-category>
              <reload-desc>Reload Command</reload-desc>
              <reload-time>2022-11-30T00:22:34+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-force-switchover</reload-category>
              <reload-desc>redundancy force-switchover</reload-desc>
              <reload-time>2022-11-29T23:40:01+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
          </reload-history>
        </device-system-data>
      </device-hardware>
    </device-hardware-data>
  </data>
</rpc-reply>
```

A NETCONF GET request for reload history data is created.

What to do next

Formore information about the YANG models, see these documents:

- The Cisco IOS XE Programmability Configuration Guide at https://www.cisco.com/c/en/us/support/wireless/catalyst-9800-series-wireless-controllers/products-installation-and-configuration-guides-list.html
- The YANG Data Models on Github at https://github.com/YangModels/yang/tree/main/vendor/cisco/xe.
- Contact the Developer Support Community for NETCONF and YANG features at: https://developer.cisco.com/

Reload reason history verification

Lists commands to verify reload history and reason details in the controller console.

Verify reload history and reason details using specific commands to troubleshoot controller issues and understand system behavior.

View reload history details

To view the reload history details, use this command:

```
Device# show reload-history

Reload History:

Reload Index: 1
Reload Code: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 01:33:44 UTC Wed Nov 30 2022

Reload Index: 2
Reload Code: Critical Process Fault
Reload Description: Critical process stack_mgr fault on rp_0_0 (rc=137), system
 report at bootflash:core/Yang_Test-system-report_20221130-012929-UTC.tar.gz
Reload Severity: Abnormal Reboot
Reload Time: 01:31:11 UTC Wed Nov 30 2022

Reload Index: 3
Reload Code: Image Install
Reload Description: Image Install
Reload Severity: Normal Reboot
Reload Time: 01:25:03 UTC Wed Nov 30 2022

Reload Index: 4
Reload Code: Critical Process Fault
Reload Description: Critical process rif_mgr fault on rp_0_0 (rc=137), system
 report at bootflash:core/Yang_Test-system-report_20221130-011127-UTC.tar.gz
Reload Severity: Abnormal Reboot
Reload Time: 01:13:08 UTC Wed Nov 30 2022

Reload Index: 5
Reload Code: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 01:08:26 UTC Wed Nov 30 2022

Reload Index: 6
Reload Code: Critical Process Fault
Reload Description: Critical process wncmgrd fault on rp_0_0 (rc=137), system
 report at bootflash:core/Yang_Test-system-report_20221130-010338-UTC.tar.gz
Reload Severity: Abnormal Reboot
Reload Time: 01:05:23 UTC Wed Nov 30 2022

Reload Index: 7
Reload Code: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 01:01:09 UTC Wed Nov 30 2022

Reload Index: 8
Reload Code: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 00:57:27 UTC Wed Nov 30 2022

Reload Index: 9
Reload Code: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
```

```
Reload Time: 00:22:34 UTC Wed Nov 30 2022

Reload Index: 10
Reload Code: Fast Switchover
Reload Description: redundancy force-switchover
Reload Severity: Normal Reboot
Reload Time: 23:40:01 UTC Tue Nov 29 2022
```

View reason for last reload

To view reason for the last reload, use this command:

```
Device# show platform software tdl-database content ios device data
Device Current time: 04:06:04
Device boot time: 01:33:37
Software version: Cisco IOS Software [Dublin], C9800-CL Software
(C9800-CL-K9_IOSXE), Experimental Version 17.11.20221012:120806
[BLD_POLARIS_DEV_S2C_20221010_023625-1-g5ebdd5c35512:/nobackup/saikarth/polaris_relhis
 103]
Copyright (c) 1986-2022 by Cisco Systems, Inc.
Compiled Wed 12-Oct-22 05:08 by saikarth
Rommon version: IOS-XE ROMMON
Last Reboot reason: Reload Command
Reboot reason severity: Normal Reboot
Unsaved configuration:  * Unknown boolean *

Reload History:

Reload Category: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 11/30/2022 01:33:44 UTC

Reload Category: Critical Process Fault
Reload Description: Critical process stack_mgr fault on rp_0_0 (rc=137), system
 report at bootflash:core/Yang_Test-system-report_20221130-012929-UTC.tar.gz
Reload Severity: Abnormal Reboot
Reload Time: 11/30/2022 01:31:11 UTC

Reload Category: Image Install
Reload Description: Image Install
Reload Severity: Normal Reboot
Reload Time: 11/30/2022 01:25:03 UTC

Reload Category: Critical Process Fault
Reload Description: Critical process rif_mgr fault on rp_0_0 (rc=137), system
 report at bootflash:core/Yang_Test-system-report_20221130-011127-UTC.tar.gz
Reload Severity: Abnormal Reboot
Reload Time: 11/30/2022 01:13:08 UTC

Reload Category: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 11/30/2022 01:08:26 UTC

Reload Category: Critical Process Fault
Reload Description: Critical process wncmgrd fault on rp_0_0 (rc=137), system
 report at bootflash:core/Yang_Test-system-report_20221130-010338-UTC.tar.gz
Reload Severity: Abnormal Reboot
Reload Time: 11/30/2022 01:05:23 UTC
```

```
Reload Category: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 11/30/2022 01:01:09 UTC

Reload Category: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 11/30/2022 00:57:27 UTC

Reload Category: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 11/30/2022 00:22:34 UTC

Reload Category: Fast Switchover
Reload Description: redundancy force-switchover
Reload Severity: Normal Reboot
Reload Time: 11/29/2022 23:40:01 UTC
```

Part II

System Upgrade

Topics:

- Upgrade the Cisco Catalyst 9800 Wireless Controller Software
- In-Service Software Upgrade
- Staggered AP Upgrade
- Software Maintenance Upgrade
- Efficient Image Upgrade and Out-of-band AP Image Download
- Predownloading an Image to an Access Point
- N+1 Hitless Rolling AP Upgrade
- NBAR Dynamic Protocol Pack Upgrade
- Wireless Sub-Package for Switch

12 Upgrade the Cisco Catalyst 9800 Wireless Controller Software

Topics:

- Software upgrade for Cisco Catalyst 9800 Series Wireless Controller

Details upgrade procedures for the Cisco Catalyst 9800 Wireless Controller, outlining key concepts, step-by-step instructions for both GUI and CLI methods, conversion between bundle and install modes, and the process for copying a WebAuth tar bundle to a standby controller.

Software upgrade for Cisco Catalyst 9800 Series Wireless Controller

Methods to upgrade the Cisco Catalyst 9800 Series Wireless Controller Software.

A software upgrade for Cisco Catalyst 9800 Series Wireless Controllers is a series of improvements and maintenance updates that:

- introduce new features and enhancements with each major release,
- offer various upgrade methods including full image upgrades and bug patch applications, and
- support upgrading through either GUI or CLI interfaces for user convenience.

This section describes the upgrade process and the methods to upgrade the Cisco Catalyst 9800 Series Wireless Controller Software.

Newer versions of the controller software are released at regular intervals. This includes major releases as well as rebuild releases that focuses on bug fixes. The version of the AP software is also tied to the controller software release. Every major Cisco IOS XE software release contains new sets of features that are essential for the enterprise-class customers.

Based on your requirement, such as upgrading the full image or applying a software patch for bugs, you can go for an appropriate software upgrade, using either GUI or CLI.

- Upgrade the Controller Software (GUI)
- Upgrade the Controller Software (CLI)

Software Upgrade Options

- **Software Maintenance Upgrade**: This method installs a software package on the system to provide a patch fix or a security resolution to a released image. This upgrade package is provided on a per release and per component basis, and is specific to the platform.
- **Hitless Upgrade**: This method allows the APs to be upgraded in a staggered manner, while still being connected to the same controller. This avoids upgrade downtime even for N+1 networks.
- **In-Service Software Upgrade**: This method upgrades a wireless controller image to a later release while the network forwards packets. This feature is supported only within and between major releases.

> **Note**
>
> We recommend In-Service Software Upgrade if you are upgrading the entire image or cold controller SMU. Use Software Maintenance Upgrade for software patches or bug fixes.

The software upgrade time is estimated to be less than 6 hours for a large network. However, the upgrade time depends on factors such as the number of APs, the percentage of APs to upgrade in each iteration, the controller type (9800-80, 9800-L, and so on), and the connectivity between the controller and the APs.

Device Upgrade Options

The following device upgrade options are available:

- NBAR Dynamic Protocol Pack Upgrade: Protocol packs are software packages that update the Network-Based Application Recognition (NBAR) engine protocol support on a device without replacing the Cisco software on the device. A protocol pack contains information on applications that are officially supported by NBAR, and are compiled and packed together.

- Field Programmable Upgrade: These are hardware programmable packages released by Cisco to upgrade the hardware programmable firmware. Hardware programmable package upgrade is necessary only when a system message indicates that one of the field programmable devices needs an upgrade or when a Cisco technical support representative suggests an upgrade.

Upgrade the controller software using GUI

Upgrade the controller software using GUI.

The purpose of this task is to guide users through the process of upgrading the controller software via the graphical user interface (GUI), ensuring a seamless transition to the latest version.

- Ensure device configurations remain intact during the upgrade process.
- Provide an option for users to select their preferred upgrade and transport modes.

Before you begin

Clean up the old installation files using the **Remove Inactive Files** link.

- Software Maintenance Upgrade
- AP Service Package
- AP Device Package

Procedure

1. Choose **Administration > Software Management** .
2. Choose an option from the **Upgrade Mode** drop-down list:
   - **INSTALL**: The Install mode uses a package-provisioning file named *packages.conf* in order to boot a device.
   - **BUNDLE**: The Bundle mode uses monolithic Cisco IOS images to boot a device. The Bundle mode consumes more memory than the Install mode because the packages are extracted from the bundle and copied to RAM.

   Note

   You get to view the **Destination** field only for BUNDLE upgrade mode.

3. From the **Transport Type** drop-down list, choose the transfer type to transfer the software image to your device as **TFTP**, **SFTP**, **FTP**, **Device**, or **Desktop (HTTP)**.

   Table 15: Transport Type Options

If...	Then...
You choose **TFTP** as the **Transport Type**	Enter the **Server IP Address** of the TFTP server that you want to use. Also, enter the complete **File Path**. In controllers, the IP TFTP source is mapped to the service port by default.

If...	Then...
You choose **SFTP** as the **Transport Type**	Enter the **Server IP Address** of the SFTP server that you want to use. Also, enter the **SFTP Username**, **SFTP Password**, and the complete **File Path**.
You choose **FTP** as the **Transport Type**	Enter the **Server IP Address** of the FTP server that you want to use. Also, enter the **FTP Username**, **FTP Password**, and the complete **File Path**.
You choose **Device** as the **Transport Type**	Choose the **File System** from the drop-down list. In the **File Path** field, browse through the available images or packages from the device and select one of the options, and click **Select**.
You choose **Desktop (HTTPS)** as the **Transport Type**	Choose the **File System** from the drop-down list. In the **Source File Path** field, click **Select File** to select the file, and click **Open**.

4. Click **Download & Install**.
5. To boot your device with the new software image, click **Save Configuration & Activate**.
6. Click **Commit** after the device reboots to make the activation changes persistent across reloads.

> **Note**
>
> For 17.4 and later releases, this step is mandatory for the upgrade to be persistent. If you do not click **Commit**, the auto-timer terminates the upgrade operation after 6 hours, and the controller reverts back to the previous image.

After completing the upgrade, the controller runs the latest version of the software, maintaining continuity and performance enhancements brought by the new update.

Upgrade the controller software (CLI)

Upgrade the controller software using CLI.

The purpose of this task is to update the controller software to the latest version using CLI to ensure optimal performance and security.

Before you begin

- Determine the Cisco IOS release that is currently running on your controller, and the filename of the system image using the **show version** command in user EXEC or privileged EXEC mode.
- Clean up the old installation files using the **install remove inactive** command.
- Use the **show version | include Installation mode** to verify the boot mode.

> **Note**
>
> We recommend that you use install mode for the software upgrade.

For steps on converting the device from bundle-mode to install-mode, see Converting from Bundle-Mode to Install-Mode.

Procedure

1. Download the software from Cisco.com: https://software.cisco.com/download/home/286322524
   - Click IOS XE Software link.
   - Select the release number you want to install, for example Gibraltar-16.12.3.

     > **Note**
     >
     > Cisco recommended release is selected by default. For release designation information, see: https://software.cisco.com/download/static/assets/i18n/reldesignation.html?context=sds

   - Click **Download**.
2. Copy the new image to flash using the command: **copy tftp:***image* **flash:**
3. Verify that the image has been successfully copied to flash using the command: **dir flash:**
4. Upgrade the software by choosing an upgrade process from the options that are currently supported.

   For a list of upgrade options, see Software Upgrade Options on page 202.

Upon completing this task, the controller software will be successfully upgraded to the latest version, enhancing device functionality and security.

Convert from bundle-mode to install-mode

Convert From Bundle-Mode to Install-Mode.

Use the procedure given below to boot in install-mode:

Before you begin

- Clean up the old installation files using the command **install remove inactive**
- Verify the boot mode using the command: **show version | include Installation mode**
- Download the software image from Cisco.com.

  For steps on how to download the software, see *Upgrading the Controller Software (CLI)*

Procedure

1. Copy the new image to flash using the command: **copy tftp:*image* flash:**

```
Device# copy tftp://xx.x.x.x//C9800-universalk9_wlc.xx.xx.xx.SSA.bin flash:

Destination filename [C9800-universalk9_wlc..xx.xx.xx..SSA.bin]?
Accessing tftp://xx.x.x.x//C9800-universalk9_wlc.xx.xx.xx.SSA.bin...
Loading /C9800-universalk9_wlc.xx.xx.xx.SSA.bin from xx.x.x.x (via
GigabitEthernet0/0):
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
[OK - 601216545 bytes]
601216545 bytes copied in 50.649 secs (11870255 bytes/sec)
```

2. Verify that the image has been successfully copied to flash using the command: **dir flash:**

```
Device# dir flash:*.bin

Directory of bootflash:/*.bin

On Active

Directory of bootflash:/

   12  -rw-  1231746613  Jun 11 2020 23:15:49 +00:00
C9800-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20200611_101837.SSA.bin
   17  -rw-  1232457039   Jun 9 2020 21:14:40 +00:00
C9800-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20200609_031801.SSA.bin
   21  -rw-  1219332990  Jun 10 2020 02:06:14 +00:00
C9800-universalk9_wlc.BLD_V173_THROTTLE_LATEST_20200608_003622_V17_3_0_183.SSA.bin

   18  -rw-  1232167230   Jun 8 2020 02:42:22 +00:00
C9800-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20200607_002322.SSA.bin
24811823104 bytes total (16032391168 bytes free)


On Standby
Directory of stby-bootflash:/*.bin

Directory of stby-bootflash:/

   18  -rw-  1232167230   Jun 8 2020 02:42:22 +00:00
C9800-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20200607_002322.SSA.bin
   20  -rw-  1231746613  Jun 11 2020 23:15:49 +00:00
C9800-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20200611_101837.SSA.bin
   17  -rw-  1232457039   Jun 9 2020 21:14:40 +00:00
C9800-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20200609_031801.SSA.bin
   16  -rw-  1219332990  Jun 10 2020 02:06:14 +00:00
C9800-universalk9_wlc.BLD_V173_THROTTLE_LATEST_20200608_003622_V17_3_0_183.SSA.bin
26462998528 bytes total (17686335488 bytes free)
```

3. Set the boot variable to **bootflash:packages.conf**.

```
Device(config)# boot sys flash bootflash:packages.conf
```

4. Save your changes by entering this command: **write memory**.

```
Device(config)# write memory
```

> **Note**
>
> For this configuration, the mode changes from enable to config mode.

5. Verify whether the boot variable is set to bootflash:packages.conf using the command: **show boot**

```
Device# show boot

BOOT variable = bootflash:packages.conf,12;
CONFIG_FILE variable =
BOOTLDR variable does not exist
Configuration register is 0x2102

Standby BOOT variable = bootflash:packages.conf,12;
Standby CONFIG_FILE variable =
Standby BOOTLDR variable does not exist
Standby Configuration register is 0x2102
```

6. Move the device from bundle-mode to install-mode using the command: **install add file** *image.bin location* **activate commit**

```
Device# install add file bootflash:C9800-universalk9_wlc.xx.xx.xx.SPA.bin
activate commit

install_add_activate_commit: START Thu Dec 6 15:43:57 UTC 2018
Dec 6 15:43:58.669 %INSTALL-5-INSTALL_START_INFO: R0/0: install_engine: Started
 install
one-shot bootflash:C9800-xx-universalk9.xx.xx.xx.SPA.bin
install_add_activate_commit: Adding PACKAGE

--- Starting initial file syncing ---
Info: Finished copying bootflash:C9800-xx-universalk9.xx.xx.xx.SPA.bin to the
 selected
chassis
Finished initial file syncing

--- Starting Add ---
Performing Add on all members
[1] Add package(s) on chassis 1
[1] Finished Add on chassis 1
Checking status of Add on [1]
Add: Passed on [1]
Finished Add

Image added. Version: xx.xx.xx.216
install_add_activate_commit: Activating PACKAGE
Following packages shall be activated:
/bootflash/C9800-xx-rpboot.xx.xx.xx.SPA.pkg
/bootflash/C9800-xx-mono-universalk9.xx.xx.xx.SPA.pkg
This operation requires a reload of the system. Do you want to proceed? [y/n]y
--- Starting Activate ---
```

```
Performing Activate on all members
[1] Activate package(s) on chassis 1
--- Starting list of software package changes ---
Old files list:
Removed
C9800-xx-mono-universalk9.BLD_Vxxxx_THROTTLE_LATEST_20181022_153332.SSA.pkg
Removed C9800-xx-rpboot.BLD_Vxxxx_THROTTLE_LATEST_20181022_153332.SSA.pkg
New files list:

Added C9800-xx-mono-universalk9.xx.xx.xx.SPA.pkg
Added C9800-xx-rpboot.xx.xx.xx.SPA.pkg
Finished list of software package changes
[1] Finished Activate on chassis 1
Checking status of Activate on [1]
Activate: Passed on [1]
Finished Activate

--- Starting Commit ---
Performing Commit on all members
[1] Commit package(s) on chassis 1
[1] Finished Commit on chassis 1
Checking status of Commit on [1]
Commit: Passed on [1]
Finished Commit

Install will reload the system now!
SUCCESS: install_add_activate_commit Thu Dec 6 15:49:21 UTC 2018
Dec 6 15:49:21.294 %INSTALL-5-INSTALL_COMPLETED_INFO: R0/0: install_engine:
Completed
install one-shot PACKAGE bootflash:C9800-xx-universalk9.xx.xx.xx.SPA.bin
```

Note

The system reloads automatically after executing the **install add file activate commit** command. You do not have to manually reload the system.

If upgrade fails, cleanup is required before attempting the upgrade procedure again. An upgrade failure may occur due lack of disk space, validation failure of extracted image, system crashes, and so on. Should a system failure occurs during upgrade process, wait till the system is back in service and check the system image version.

- If it is a new image, check for the stability and functionality of the system, and decide whether to commit and complete the upgrade procedure or discard the upgrade procedure.
- If it is a new image, use the cleanup procedure and reattempt the upgrade procedure.

7. Click **yes** to all the prompts.
8. Verify the boot mode using the command: **show version**

```
Device# show version | in Installation mode is

Installation mode is INSTALL
```

Copy a WebAuth tar bundle to the standby controller (GUI)

Copy a WebAuth tar bundle to the standby controller in a high-availability configuration.

Ensure the standby controller receives the WebAuth tar bundle to maintain seamless high-availability functionality.

Copy a WebAuth tar bundle to the standby controller in a high-availability configuration.

Procedure

1. Go to **Administration** , select **Management** , and then select **Backup and Restore**.
2. Select **To Device** from the **Copy** drop-down list.
3. Select **WebAuth Bundle** from the **File Type** drop-down list.
4. Select **TFTP** , **SFTP** , **FTP** , or **HTTP** from the **Transfer Mode** drop-down list.

   The required values for the **Server IP Address** and **File Path** fields depend on the selected transfer mode.

   - **TFTP**
     - **IP Address (IPv4/IPv6)** : Enter the server IP address (IPv4 or IPv6) of the TFTP server you want to use.
     - **File Path**: Enter the file path. Start the file path with a slash, such as */path*.
     - **File Name**: Enter a file name.

       Do not use spaces in the file name. Use underscores and hyphens. Make sure the file name ends with .tar, for example, webauthbundle.tar.
   - **SFTP**
     - **IP Address (IPv4/IPv6)**: Enter the server IP address (IPv4 or IPv6) of the SFTP server that you want to use.
     - **File Path**: Enter the file path. Start the file path with a slash, such as */path*.
     - **File Name**: Enter a file name.

       Do not use spaces in the file name. Only underscores and hyphens are allowed. Make sure the file name ends with .tar, for example, webauthbundle.tar.
     - **Server Login Username**: Enter the SFTP server login user name.
     - **Server Login Password**: Enter the SFTP server login passphrase.
   - **FTP**
     - **IP Address (IPv4/IPv6)**: Enter the server IP address (IPv4 or IPv6) of the FTP server that you want to use.
     - **File Path**: Enter the file path. Start the file path with a slash, such as */path* .
     - **File Name**: Enter a file name.

       Do not use spaces in the file name. Only underscores and hyphens are allowed. Make sure the file name ends with .tar, for example, webauthbundle.tar.
     - **Logon Type**: Choose **Anonymous** or **Authenticated** . If you choose **Authenticated**, these fields are activated:
       - **Server Login Username**: Enter the FTP server login user name.
       - **Server Login Password**: Enter the FTP server login passphrase.
   - **HTTP**

- **Source File Path**: Click **Select File** to select the configuration file, and click **Open**.

5. Click the **Yes** or **No** radio button to back up the existing startup configuration to Flash.

   Save the configuration to Flash to propagate the WebAuth bundle to other members, including the standby controller.

6. Click **Download File**.

The WebAuth tar bundle is successfully copied to the standby controller, ensuring high-availability readiness.

13 In-Service Software Upgrade

Topics:

- In-service software upgrade (ISSU)

In-service software upgrade (ISSU)

Explains how In-Service Software Upgrades (ISSUs) allow network administrators to upgrade Cisco wireless controllers without disrupting network packet forwarding.

An In-Service Software upgrade (ISSU) is a wireless controller procedure that

- upgrades the controller software image to a later release while the network continues to forward packets
- enables administrators to avoid network outages during software upgrades, and
- allows cold patches to be applied without impacting the active network.

ISSU is supported only on these Cisco Catalyst 9800 Series Wireless Controllers. It supports only upgrade operations.

- Cisco Catalyst 9800-80 Wireless Controller
- Cisco Catalyst 9800-40 Wireless Controller
- Cisco Catalyst 9800-L Wireless Controller
- Cisco Catalyst 9800-CL Wireless Controller (Private Cloud)
- Cisco Wireless 9800M Wireless Controller
- Cisco Wireless 9800H1/H2 Wireless Controller

High-Level Workflow of ISSU

1. Onboard the controller software image to the flash memory.
2. Download the AP image to the AP.
3. Install the controller software image.
4. Commit the changes.

Table 16: Feature history for In-Service Software Upgrade (ISSU)

Feature Name	Release Information	Feature Description
In-Service Software Upgrade (ISSU)	Cisco IOS XE 17.1.1s	This feature upgrades an image to another image on a device while the network continues to forward packets. ISSU helps network administrators avoid a network outage when performing a software upgrade.

Prerequisites for performing ISSU

Prerequisites for Performing ISSU.

- Ensure that both Active and Standby controllers are in install mode and are booted from *bootflash:/packages.conf*.
- Ensure that the network or device is not being configured during the upgrade.
- Schedule the upgrade when your network is stable and steady.
- Ensure that the system has an uninterrupted power supply. A power interruption during the upgrade procedure can corrupt the software image.

Guidelines for ISSU

ISSU guidelines and restrictions.

- During ISSU upgrade, the system displays a warning message similar to:

  ```
  found 46 disjoint TDL objects
  ```

  The warning message does not impact system functionality.

- During ISSU upgrade, if both the controllers (active and standby) have different images after the power cycle, an auto cancel of ISSU is triggered to bring both the controllers to the same version. The following is a sample scenario: Install Version 1 (V1) software on the active controller. Apply a SMU hot patch. Perform a commit. Upgrade to Version 2 using ISSU. Power cycle the active controller. The system now has a version mismatch (V1 and V2). After bulk synchronization, the active controller reloads. Both controllers have the same version (V1 and V1).
- To enable the **clear install** command, you should first run the **service internal** command in global configuration mode, and then run the **clear install** command in privileged EXEC mode.
- A stale rollback history can affect image rollback after the stack forms. We recommend running the **clear install state** command to clear stale information and boot the controller in bundle mode.
- After the image upgrade, when the new active controller starts up, it does not retain the old logs on the web GUI window as part of show logs.
- During a stateful switchover (SSO) or HA event, the rolling AP upgrade stops. Use the **ap image upgrade** command to restart the upgrade process.
- Use the clear **ap predownload statistics** running the **show ap image** command. This ensures you get accurate data after every pre-download.
- To avoid a software version mismatch between the active and standby controller, manually cancel the ISSU process by using the **install issu abort** command in the appropriate scenarios.
  - The system may disable an RP link after standby HOT during an ISSU procedure. The link stays down after the auto-abort timer expires.
  - An RP link is brought down before the standby controller reaches standby HOT during an ISSU procedure.
- The ISSU upgrade from version 17.12 to 17.15 fails if WPA3 suite-b-192, suite-b, gcmp128, gcmp256, or ccmp256 are already configured.

Restrictions for ISSU

- If you do not run the **install commit** command within six hours of the **install activate issu** command, the system reverts to the original commit position. Delay the commit by using the Delayed Commit procedure.
- You cannot use the **install abort** command during the AP rolling upgrade. You should use the **install abort issu** command, instead to cancel the upgrade.
- If configuration synchronization fails on the standby controller, the ISSU upgrade cancels and rolls back to software image V1. However, this information isn't available in the **show install** command log. Run the **show issu state detail** command to see the current ISSU state.
- The **clear install state** command doesn't delete the SMU file from flash or storage. To remove a SMU, use either the **install remove file** command or the **install remove inactive** command.
- If HA does not form after the ISSU procedure, reload one chassis.
- You cannot use Cisco TrustSec (CTS) on RMI interfaces.

- If a switchover occurs while performing an AP upgrade using ISSU, the upgrade process will restart automatically after the switchover.

Upgrade software using ISSU (CLI)

Upgrade software using ISSU using commands.

Use this procedure to upgrade the image from one version to another.

Note

ISSU is supported only within and between major releases. For example, upgrades are supported from 17.3.*x* to 17.3.*y* or 17.6.*x* to 17.6.*y* (within a major release), as well as from 17.3.*x* to 17.6.*x* and from 17.3.*x* to 17.9.*x* (among major releases). For more information, check https://www.cisco.com/c/en/us/td/docs/wireless/controller/9800/tech-notes/b_issu_9800.html#id_129557. ISSU is not supported within or between minor releases or between minor and major releases. For example, upgrades such as 17.4.*x* to 17.4.*y*, 17.4.*x* to 17.5.*x* or 17.3.*x* to 17.4.*x* are not supported.

ISSU downgrade is not supported for Cisco Catalyst 9800 Series Wireless Controller platforms.

Note

We recommend configuring the percentage of APs to be upgraded by using the ***ap upgrade staggered*** command.

Procedure

1. Install the file to expand the wireless package file.

   Example

   ```
   Device# install add file file-name
   ```

   The controller software image is added to the flash and expanded.

   Note

   In Cisco Catalyst 9800 Wireless Controller for Switch, run the **install add file** *sub-package-file-name* command to expand the wireless subpackage file.

2. Perform predownload of the AP image.

   Example

   ```
   Device# ap image predownload
   ```

   To see the progress of the predownload, use the **show ap image** command.

3. Run compatibility checks, install the package, and update the package status details.

   Example

   ```
   Device# install activate issu [auto-abort-time timer]
   ```

   Optionally, you can configure the time limit to cancel the addition of new software without committing the image. The valid values are from 30 to 1200 minutes.

4. Run either of the following commands:

   - Cancel the upgrade process and return to the device to the previous installation state.

     ```
     Device# install abort issu
     ```

     This is applicable for both controller and the AP.

   - Commit the activation changes to be persistent across reloads.

     ```
     Device# install commit
     ```

     Note

     If you do not run the **install commit** command within 6 hours of completing the previous step, the system will revert to the original commit position.

Upgrade software using ISSU (GUI)

Upgrade software using ISSU using GUI.

Before you begin

1. The device should be in Install mode.
2. It should also have a high-availability (HA) pair. The standby controller should be online and in SSO mode.

   You can verify the details using **show issu state detail** command.

Procedure

1. Choose **Administration** > **Software Management**.
2. Check the **ISSU Upgrade (HA Upgrade) (Beta)** check box under the **Software Upgrade** tab.
3. Select the **AP Upgrade per Iteration** percentage option of APs to be upgraded from the drop-down list in the **AP Upgrade Configuration** section.
4. Click **Download & Install**.

   This initiates the upgrade process and you can view the progress in the **Status** dialog box.

   Click the **Show Logs** link to view the upgrade process details.

> **Note**
>
> An SSO takes place while activating the image on the active controller. After the SSO, you should login again to the controller.

5. The system enables the **Commit** and **ISSU Abort** buttons after the upgrade.

   Click **Commit** to commit the activation changes, or **ISSU Abort** to terminate the upgrade process and return the device to the previous installation state.

Upgrade software using ISSU with delayed commit (CLI)

Upgrade the controller software using ISSU with delayed commit.

Use this procedure to upgrade the controller software with delayed commit. This lets you run and test the new software without committing the image.

Procedure

1. Add and expand the controller software image to the flash.

   Example

   ```
   Device# install add file file-name
   ```

   > **Note**
   >
   > In Cisco Catalyst 9800 Wireless Controller for Switch, run the **install add file** *sub-package-file-name* command to expand the wireless subpackage file.

2. Perform predownload of the AP image.

   Example

   ```
   Device# ap image predownload
   ```

3. Run compatibility checks, install the package, and update the package status details.

   Example

   ```
   Device# install activate issu
   ```

   The ISSU completes activation and a message that the **terminal state is reached** is displayed in the **show issue state detail** command output.

4. Stop the termination timer.

Example

```
Device# install auto-abort-timer stop
```

This ensures that the upgrade process is not terminated after the default termination time of six to eight hours.

5. Commit the activation changes to be persistent across reloads.

Example

```
Device# install commit
```

6. Optional: Abort ISSU and go back to the previous version.

Example

```
Device# install abort issu
```

Monitor ISSU

Monitor ISSU using these commands.

To view the ISSU state after the install add ISSU and before the install activate ISSU, use the command:

```
Device# show issu state detail

-- Starting local lock acquisition on chassis 1 ---
Finished local lock acquisition on chassis 1
Current ISSU Status: Enabled
Previous ISSU Operation: Abort Successful
=====================================================
System Check Status
-----------------------------------------------------
Platform ISSU Support Yes
Standby Online Yes
Autoboot Enabled Yes
SSO Mode Yes
Install Boot Yes
Valid Boot Media Yes
=====================================================
No ISSU operation is in progress
show install summary
[ Chassis 1 2 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type St Filename/Version
--------------------------------------------------------------------------------
IMG I 17.1.1.0.432
IMG C 16.12.2.0.2707
--------------------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------------------
```

To view the ISSU state after activating ISSU, use the command:

```
Device# show issu state detail

Current ISSU Status: In Progress
Previous ISSU Operation: Abort Successful
=========================================================
System Check Status
---------------------------------------------------------
Platform ISSU Support Yes
Standby Online Yes
Autoboot Enabled Yes
SSO Mode Yes
Install Boot Yes
Valid Boot Media Yes
=========================================================
Operation type: Step-by-step ISSU
Install type : Image installation using ISSU
Current state : Activated state
Last operation: Switchover
Completed operations:
Operation Start time
---------------------------------------------------------
Activate location standby Chassis 2 2019-09-17:23:41:12
Activate location active Chassis 1 2019-09-17:23:50:06
Switchover 2019-09-17:23:52:03
State transition: Added -> Standby activated -> Active switched-over
Auto abort timer: automatic, remaining time before rollback: 05:41:53
Running image: bootflash:packages.conf
Operating mode: sso, terminal state reached
show install summary
[ Chassis 1/R0 2/R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type St Filename/Version
--------------------------------------------------------------------------------
IMG U 17.1.1.0.432
--------------------------------------------------------------------------------
Auto abort timer: active on install_activate, time before rollback - 05:41:49
--------------------------------------------------------------------------------
```

To view the ISSU state after installing the commit, use the command:

```
Device# show issu state detail

--- Starting local lock acquisition on chassis 1 ---
Finished local lock acquisition on chassis 1
Current ISSU Status: Enabled
Previous ISSU Operation: Successful
=========================================================
System Check Status
---------------------------------------------------------
Platform ISSU Support Yes
Standby Online Yes
Autoboot Enabled Yes
SSO Mode Yes
Install Boot Yes
Valid Boot Media Yes
=========================================================
No ISSU operation is in progress
```

```
show install summary
[ Chassis 1/R0 2/R0 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type St Filename/Version
--------------------------------------------------------------------------------
IMG C 17.1.1.0.432
--------------------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------------------
--------------------------------------------------------------------------------
```

To view the ISSU state after terminating the ISSU process, use the command:

```
Device# show issu state detail
Current ISSU Status: In Progress
Previous ISSU Operation: Abort Successful
=====================================================
System Check Status
-----------------------------------------------------
Platform ISSU Support Yes
Standby Online Yes
Autoboot Enabled Yes
SSO Mode Yes
Install Boot Yes
Valid Boot Media Yes
=====================================================
Operation type: Step-by-step ISSU
Install type : Image installation using ISSU
Current state : Timeout-error state
Last operation: Commit Chassis 1
Completed operations:
Operation Start time
-----------------------------------------------------
Activate location standby Chassis 2 2019-09-17:23:41:12
Activate location active Chassis 1 2019-09-17:23:50:06
Switchover 2019-09-17:23:52:03
Abort 2019-09-18:00:14:13
Commit Chassis 1 2019-09-18:00:28:23
State transition: Added -> Standby activated -> Active switched-over ->
Activated -> Timeout-error
Auto abort timer: inactive
Running image: bootflash:packages.conf
Operating mode: sso, terminal state reached
```

To view the summary of the active packages in a system, use the command:

```
Device# show install summary

[ Chassis 1 2 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted,
C - Activated & Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type St Filename/Version
--------------------------------------------------------------------------------
IMG C 16.12.2.0.2707
--------------------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------------------
```

Troubleshoot ISSU

Troubleshoot ISSU using command.

Use **install activate issu** command before completing AP pre-download.

This scenario applies when you run the **install activate issu** command before AP pre-download is complete. In this situation, run the **ap image predownload** command and then activate the system.

```
Device# install activate issu

install_activate: START Wed Jan  8 04:48:04 UTC 2020
System configuration has been modified.
Press Yes(y) to save the configuration and proceed.
Press No(n) for proceeding without saving the configuration.
Press Quit(q) to exit, you may save configuration and re-enter the command.
[y/n/q]
y
Building configuration...
[OK]Modified configuration has been saved
install_activate: Activating ISSU
NOTE: Going to start Activate ISSU install process
STAGE 0: System Level Sanity Check
================================================
--- Verifying install_issu supported ---
--- Verifying standby is in Standby Hot state ---
--- Verifying booted from the valid media ---
--- Verifying AutoBoot mode is enabled ---
--- Verifying Platform specific ISSU admission criteria ---
CONSOLE: FAILED: Install operation is not allowed.
Reason -> AP pre-image download is mandatory for hitless software upgrade.
Action -> Trigger AP pre-image download.
FAILED: Platform specific ISSU admission criteria
ERROR: install_activate  exit(2 ) Wed Jan  8 04:48:37 UTC 2020
```

14 Staggered AP Upgrade

Topics:

- Access point upgrade configurations

Access point upgrade configurations

Learn about access point upgrade configurations and how to customize upgrade processes in your network..

Access point upgrade configuration are controller software parameters that:

- control the number of APs upgraded in each batch or iteration,
- include several upgrade strategies (including new staggered and serial modes), and
- are applicable across all upgrade scenarios such as In-Service Software Upgrade (ISSU), N+1 Hitless Rolling AP Upgrade, and Access Point Service Pack (APSP) or Access Point Device Package (APDP).

These configurations stagger AP updates, and provide more flexibility in large network environments.

> ⚠ **Caution**
>
> Avoid using these configurations for wncd-config upgrades (such as Locally Significant Certificate [LSC] renewal).

Examples of upgrade

- One percent per iteration: Only one percent of APs are upgraded at a time, minimizing disruption.
- Serial: One AP is upgraded at a time, resulting in the least disruption but taking much longer.

Feature history for access point upgrade management enhancements

AP Power Save feature history.

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 17: Feature history for access point upgrade management enhancements

Release	Feature Information
Cisco IOS XE 17.18.1	Provides configurations for reduced AP batch size used during AP upgrades. These configurations aim to reduce disruption during the upgrade process by providing even greater control over AP upgrade batch sizes. Since the batch sizes can be greatly reduced, such configurations should not be used during maintenance windows as it would result in long upgrade time The following commands are modified: • **ap upgrade staggered** . Two new keywords **1** and **serial** are introduced. • **show ap upgrade** The YANG model changes include new options for AP upgrade configurations and marking the node **stgrd-upgd-ap** as obsolete to support 6000 APs. The new node **comp-ap** lists APs with successfully completed upgrade.

Access point upgrade configuration and behavior

AP Upgrade batch sizes.

New configurations and operational changes in controller software give you more control over access point (AP) upgrade processes.

Here are the enhancements:

- You have more granular control with AP upgrade batch sizes, allowing minimum disruption of your network.
- You can use serial and one percent per-iteration upgrade modes to perform in-service upgrades.

> **Note**
>
> These new configurations must be used strictly outside the maintenance windows. See Warning: Extended upgrade durations for serial and one percent batch modes on page 224.

- The CLI and YANG models now support these options, so that you can manage upgrades more effectively.

These changes help you reduce service impact during AP upgrades and are particularly useful for large-scale or in-service environments.

Staggered and serial AP upgrade concepts

Learn about staggered and serial access point upgrade concepts for minimizing client impact during upgrades.

A staggered and serial AP upgrade concept is a network upgrade methodology that:

- sequentially updates subsets of access points (APs) to minimize client impact,

- allows configuration of different batch sizes, including single AP upgrades, and
- provides options for client steering and de-authentication controls to optimize network continuity during in-service upgrades.

When the AP batch size is small, the rollout process may be slower. However, because very few APs are affected in each iteration, clients have many options to retain connectivity, and large networks benefit from improved network performance.

Staggered upgrades distribute the process across multiple iterations. Serial upgrades process one AP per iteration. Use these approaches when service continuity is critical during in-service upgrades

Controller CLI enhancements for AP upgrade management

Learn about new CLI commands that help you manage AP upgrades.

Controller CLI enhancements for AP upgrade management are:

- New keywords **serial** and **1** added to the **ap upgrade staggered** command specifying serial mode and upgrade batch sizes.
- Warnings are issued for configurations that may lead to extended upgrade durations.
- The output of **show ap upgrade** now reflects the new upgrade processes.

These changes help you configure and monitor upgrades more precisely and with greater awareness of possible service impacts.

For more information about these commands, see **ap upgrade staggered** and **show ap upgrade**

Warning: Extended upgrade durations for serial and one percent batch modes

If you choose serial or one percent per iteration upgrades, the process can take a long time. For example, 6000 APs at three minutes takes over 12 days). Use these modes only when minimizing disruption is more important than upgrade speed.

Configure staggered or serial AP upgrades

Configure the wireless controller to upgrade access points in staggered batches or one at a time, minimizing client disruption.

Use this task to perform upgrades in large, production, or in-service environments where minimizing service impact is crucial.

Follow these steps to configure staggered or serial AP upgrades:

You can also configure these options through the GUI. See Configure AP certificate auto-renewal (LSC) (GUI) on page 1760

Before you begin

- Review network size and upgrade impact to select an appropriate mode.

Procedure

1. Enter global configuration mode.
2. Use the **ap upgrade staggered** command to view available options and select a batch size:
   - Enter 1 for one percent per iteration (minimal disruption, slowest).
   - Enter serial for one AP at a time (least disruption, longest duration).

- Enter 5, 15, or 25 for other batch percentages as needed.

3. Optional: Configure client de-authentication or steering.
   The controller displays a warning if **serial** or **1%** mode is selected.
4. Exit the global configuration mode.

After you complete these steps, the controller staggers AP upgrades based on your configuration.

CLI and GUI options for upgrading APs

The AP upgrade management options available on CLI and GUI

The table lists CLI and GUI options for upgrading APs.

Table 18: CLI and GUI options for upgrading APs

Command Option	Description
1	1% of APs per iteration
5	5% of APs per iteration
15	15% of APs per iteration
25	25% of APs per iteration
client-deauth	Enable or disable client de-authentication
client-steering	Enable or disable client steering
iteration	Configure iteration accounting parameters
one-shot	Upgrade all APs in a single iteration
serial	Upgrade one AP at a time

YANG Model Update

The YANG oper model for **ap-upgrade status** with **stgrd-upgd-ap** is obsolete starting with Cisco IOS XE Release 17.18.x.

A new xpath with **comp-ap** is introduced to support newer APs. The list includes all the APs that have completed the upgrade operation.

Old XPATH

```
xpath /access-point-oper-data/stgrd-upg-report/stgrd-upgd-ap
```

NEW XPATH

```
xpath /access-point-oper-data/stgrd-upg-report/comp-ap
```

Node properties of comp-ap

```
Name            comp-ap
nodetype        list
description     List of APs with successfully completed upgrades. This is the
 new list for upgraded APs, with support for serial AP upgrades.
module          Cisco-IOS-XE-wireless-access-point-oper
revision        2025-07-01
```

```
xpath           /access-point-oper-data/stgrd-upg-report/comp-ap
prefix          wireless-access-point-oper
namespace      http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-access-point-oper
schema node id  /access-point-oper-data/stgrd-upg-report/comp-ap
Keys            "iter-number""radio-mac"
access          read-only
operations      "get"
```

15 Software Maintenance Upgrade

Topics:

- Software maintenance upgrade
- AP device package
- Per site or per AP model service pack (APSP)

Software maintenance upgrade

Explains how software maintenance upgrades (SMUs) provide targeted, platform-specific patches and security fixes for released images on Cisco IOS XE systems.

A software maintenance upgrade (SMU) is a software package that

- is installed on a system to provide a patch or a security fix resolution to a released image, and
- is on a per component basis and is specific to the corresponding platform.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 19: Feature history for Software maintenance upgrade

Feature Name	Release Information	Feature Description
Software maintenance upgrade	Cisco IOS XE Gibraltar 16.11.1b	This feature provides network administrators with targeted, platform-specific patches and security fixes for released images, applicable either on a per-component basis or for the entire corresponding platform.

A SMU offers significant benefits over classic Cisco IOS software. It lets you address network issues promptly and reduces required testing time and scope. The Cisco IOS XE platform validates SMU compatibility internally and prevents installation of non-compatible SMUs.

All SMUs are also integrated into subsequent Cisco IOS XE software maintenance releases. A SMU is an independent and self-sufficient package without prerequisites or dependencies. You can choose which SMUs to install or uninstall in any order.

Note

SMUs are supported only on Extended Maintenance releases and throughout the full lifecycle of the underlying software release.

Note

Activate the file using the **install add file** command only from the filesystems of the active device. If you use a file from the standby or member filesystems, the **install add file** command fails.

Note

When the SMU file is deleted and the device is rebooted, the device may display an error message such as:

```
--- Starting SMU Add operation ---
Performing SMU_ADD on all members
    FAILED: Improper
State./bootflash/<previously-installed-smu-filename>.smu.bin not present.
Please restore file for stability.
Checking status of SMU_ADD on [1/R0]
SMU_ADD: Passed on []. Failed on [1/R0]
Finished SMU Add operation
FAILED: add_activate_commit /bootflash/<tobeinstalled-wlc-smu-filename>.smu.bin
 Wed Aug 02 08:30:18 UTC 2023.
```

This error occurs because the previous SMU file was not properly removed from the controller. As a result, you may not be able to install new SMU or APSP files.

Use the **install remove file** command to remove previous instances of APSP or SMU files from the bootflash.

You can use SMU infrastructure to meet these requirements in wireless contexts:

- Controller SMU: Controller bug fixes or Cisco Product Security Incident Response information (PSIRT).
- APSP: APSP is used for AP bug fixes, PSIRTs, or minor features that do not require controller changes.
- APDP: APDP supports new AP models without introducing new hardware or software capabilities.

> **Note**
>
> The **show ap image** command displays cumulative statistics for AP images in the controller. Clear the statistics using the **clear ap predownload statistics** command, before using the **show ap image** command, to ensure correct data is displayed.

SMU Workflow

Begin the SMU process by requesting approval from the SMU committee. Contact customer support to raise an SMU request. During the release, the SMU package is posted on the Cisco Software Download page. You can then download and install it.

Warning: Commit changes within six hours of activation or deactivation to avoid rollback

Always run the **install commit** command within 6 hours after executing either **install activate** or **install deactivate** .

If you do not commit changes within this window

- the system automatically reverts to the previous commit state
- this reversion can lead to service interruption, especially over low-bandwidth links where image transfers may not complete in time, and
- remote deployments with slow transfer rates are particularly vulnerable.

To reduce these risks

- run **install commit** after activation or deactivation
- monitor image transfer progress proactively, and
- plan for available bandwidth and duration at remote sites.

SMU Package

An SMU package contains the metadata and the fix for the reported issue that prompted the request.

SMU Reload

The SMU type describes the effect on a system after you install a SMU. SMUs can be non-traffic-affecting, or they can require a device restart, reload, or switchover.

A controller cold patch requires a cold reload of the system during activation. A cold reload means completely restarting the operating system.

This process affects traffic during two phases:

- the reload of the wireless controller, and
- the time needed for all APs to rejoin the controller, receive the new image, and upgrade to the new SMU patch. This reload ensures that all processes use the correct libraries and files installed as part of the SMU.

The reload ensures all processes use the correct libraries and files installed as part of the SMU.

With controller hot patching, a SMU is effective immediately after activation, without rebooting the system. After you commit the SMU, activation changes persist across reloads. Hot patching SMU packages contain metadata listing processes to restart for activation. During activation, each process is restarted one by one until activation completes.

Install a SMU (GUI)

Install a SMU using GUI.

Add and activate a SMU image on your device using the GUI.

Procedure

1. Choose **Administration** > **Software Management** and click the **Software Maintenance Upgrade** tab.
2. Click **Add** to add a SMU image.
3. From the **Transport Type** drop-down list, select the transfer type, TFTP, SFTP, FTP, Device, or Desktop (HTTP) to transfer the software image to your device.
   a) If you choose **TFTP** as the **Transport Type**, enter the **Server IP Address (IPv4/IPv6)**, **File path** and select a **File System** from the drop-down list. For example, if the SMU file is at the root of the TFTP server you can enter `/C9800-universalk9_wlc.17.03.02a.CSCvw55275.SPA.smu.bin` in the **File path** field.
   b) If you choose **SFTP** as the **Transport Type**, enter the **Server IP Address (IPv4/IPv6)**, **SFTP Username**, **SFTP Password**, **File path** and select a **File System** from the drop-down list.
   c) If you choose **FTP** as the **Transport Type**, enter the **Server IP Address (IPv4/IPv6)**, **FTP Username**, **FTP Password**, **File path**, and select a **File System** from the drop-down list.
   d) If you choose **Device** as the **Transport Type**, enter the **File path** and select a **File System** from the drop-down list. This is possible when the software is already present on the device due to an earlier download and activation, followed by a subsequent deactivation.

      Note

      Depending on your device, available file systems may vary. On physical controllers, store files on bootflash or hard disk. On virtual controllers, use bootflash.

   e) If you choose **Desktop (HTTPS)** as the **Transport Type**, select a **File System** from the drop-down list and click **Select File** to navigate to the **Source File Path**.
4. Enter the **File Name** and click **Add File**.

   When you complete this step, the maintenance update package is copied to the device, platform and image versions are checked for compatibility, and the SMU package is added for all members. After adding a SMU, you see a message indicating success and letting you know the SMU can be activated. The message displays the name of the package (SMU) that is available for activation. It lists the SMU details: Name, Version, State (active or inactive), Type (reload, restart, or non-reload), and other compatibility details. If the SMU type is 'reload,' any operation (activate, deactivate, or rollback) causes the device to reload. The 'restart' type involves only a process restart. If it is 'non-reload,' no process changes occur.
5. Select the SMU and click **Activate** to activate it on the system, install the package, and update the package status details.
6. Select the SMU and click **Commit** to make these activation changes persistent across reloads.

   The Commit operation creates commit points, which are similar to snapshots. Use commit points to decide which changes to activate or roll back if you have a problem with the SMU. You can commit after activation when the system is up, or after the first reload. If a package is activated but not committed, it remains active after the first reload, but not after the second reload.

Install SMU (CLI)

Install SMU using commands.

Update device software with a SMU using CLI.

Procedure

1. Copy the maintenance update package from a remote location to the device, and perform a compatibility check for the platform and image versions.

   Example

   ```
   Device# install add file bootflash filename
   ```

   This command runs base compatibility checks on a file to verify that the SMU package is supported on the platform. The command also adds an entry to the package or SMU.sta file. This entry allows you to monitor and maintain the package's status.

2. Run compatibility checks, install the package and update the package status details.

   Example

   ```
   Device# install activate file bootflash filename
   ```

   For a restartable package, the command triggers the appropriate post-install scripts to restart the necessary processes. For non-restartable packages it triggers a reload.

3. Commit the activation changes to be persistent across reloads.

   Example

   ```
   Device# install commit
   ```

   You can perform the commit after activation while the system is up, or after the first reload. If a package is activated but not committed, it remains active after the first reload; however, it is no longer active after the second reload.

4. Display the image version on the device.

   Example

   ```
   Device# show version
   ```

5. Display information about the active package.

   Example

   ```
   Device# show install summary
   ```

   The output of this command varies according to the install commands that are configured.

Roll back an image (GUI)

Roll back an image using GUI.

Return the software image on the system to a previous stable state using the GUI.

Procedure

1. Choose **Administration** > **Software Management**.
2. Go to **SMU**, **APSP** or **APDP**.
3. Click **Rollback**.
4. In the **Rollback to** drop-down list, select **Base**, **Committed** or **Rollback Point**.
5. Click **Add File**.

Roll back SMU (CLI)

Roll back SMU using commands.

Return the device to a previous software state by rolling back a SMU using CLI.

Procedure

1. Return the device to the previous installation state.

   Example

   ```
   Device(config)# install rollback to {base | committed | id | committed}
   committed ID id 1234
   ```

   After the rollback, a reload is required.

2. Commit the activation changes to be persistent across reloads.

   Example

   ```
   Device# install commit
   ```

Deactivate SMU (CLI)

Deactivate SMU using commands.

Deactivate an active SMU package on a network device using CLI.

Procedure

1. Deactivate an active package, update the package status, and trigger a process to restart or reload.

   Example

   ```
   Device# install deactivate file bootflash filename
   ```

2. Commit the activation changes to be persistent across reloads.

Example

```
Device# install commit
```

Configuration examples for SMU

Sample of the SMU configuration.

This example shows the SMU configuration after the system completes the install add for the SMU.

```
Device# show install summary

[ Chassis 1 2 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted, C - Activated &
Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type  St   Filename/Version

--------------------------------------------------------------------------------


IMG   C    16.8.1.0.39751


--------------------------------------------------------------------------
Auto abort timer: inactive

--------------------------------------------------------------------------
```

AP device package

Explains how AP device packages use the SMU infrastructure to extend wireless network support for new AP hardware without controller upgrades.

An AP device package is a software module that

- allows new AP hardware models to connect to an existing wireless network
- uses the SMU infrastructure to deliver new AP images, and
- enables flexible, out-of-cycle support for new AP devices independent of controller major releases.

AP Service Pack (APSP)

When a new AP hardware model is introduced, it is shipped along with the corresponding controller related major software version. Subsequently, you must wait for the release of a controller version compatible with the new AP model before upgrading the entire network.

Starting with version 16.11.1, you can add new AP models to your wireless network using the SMU infrastructure, without upgrading the controller version. This solution is called AP Device Package (APDP).

SMU Process or Workflow

The SMU process builds APDP to detect code changes and to build APDP. You can add new AP image files to APDP and include AP images as needed.

The workflow is as follows:

- install add
- install activate
- install commit

For more details, see Managing AP Device Package.

Warning: Commit changes within six hours of activation or deactivation

To complete the APSP or APDP activation or deactivation process, always run the **install commit** command within six hours after executing either **install activate** or **install deactivate**.

If you do not commit the changes within this window, the system reverts to the previous commit state automatically. Potential issues include:

- service interruptions, especially over low bandwidth links where image transfers may not finish in time, and
- rollbacks for remote deployments with slow transfer rates.

To avoid these risks:

- Run the **install commit** command immediately after activation or deactivation.
- Monitor the progress of the image transfer.
- Plan to have sufficient time available at remote sites.

SMU package and AP image changes

SMU Package

A SMU package is a software metadata unit that:

- contains AP model and capability-related details
- is updated when new AP models are introduced, and
- is bundled into APDP packages for deployment.

AP Image Changes

When new AP models are introduced, there may or may not be corresponding new AP images. AP images are mapped to the AP model families. If a new AP model belongs to an existing AP model family there will be existing AP image entries (for example: ap3g3 and ap1g5). If an AP model belongs to either ap3g3 or ap1g5, update the respective image file with the correct AP image location. The corresponding metadata file is also updated with the new AP model capability information.

If a new AP model belongs to a new AP model family and requires a new image file, create a new image entry file in the correct AP image location. The corresponding metadata file is updated with the new AP model capability information.

> **Note**
>
> The APDP images must not be renamed to prevent impact on its functionality.

Install AP device package (GUI)

Install AP device package using GUI.

Add or upgrade AP device support in your network.

Procedure

1. Choose **Administration** > **Software Management**.
2. Click **AP Device Package (APDP)** tab.
3. Click **Add**.
4. From the **Transport Type** drop-down list, select the transfer type to transfer the software image to your device as TFTP, SFTP, FTP, Device, or Desktop (HTTP).
   a) If you choose **TFTP** as the **Transport Type**, enter the **Server IP Address (IPv4/IPv6)**, **File path** and select a **File System** from the drop-down list.
   b) If you choose **SFTP** as the **Transport Type**, enter the **Server IP Address (IPv4/IPv6)**, **SFTP Username**, **SFTP Password**, **File path** and select a **File System** from the drop-down list.
   c) If you choose **FTP** as the **Transport Type**, enter the **Server IP Address (IPv4/IPv6)**, **FTP Username**, **FTP Password**, **File path**, and select a **File System** from the drop-down list.
   d) If you choose **Device** as the **Transport Type**, enter the **File path** and select a **File System** from the drop-down list.
   e) If you choose **Desktop (HTTPS)** as the **Transport Type**, select a **File System** from the drop-down list.
   f) Click **Select File** to navigate to the **Source File Path**.
5. Enter the **File Name** and click **Add File**.
6. From the **AP Upgrade Configuration** section, select the percentage of APs to be included from the **AP Upgrade per iteration** drop-down list.
7. Click **Apply**.

Install AP device package (CLI)

Install AP device package using commands.

Install and activate new AP device packages on the controller using the CLI.

Procedure

1. Extract AP images from APDP and place them in SMU or APDP specific mount location.

   Example

   ```
   Device# install add file bootflash filename
   ```

> **Note**
>
> Here, the SMU does not trigger the Wireless module.

2. Add the AP software in APDP to the existing current active AP image list.

   Example

   ```
   Device# install activate file bootflash filename
   ```

   Also, update the capability information for the new AP models in the controller .

   > **Note**
   >
   > Even if the new AP module supports new hardware capabilities, the controller recognizes only the capability information that its base version supports.

   At this point, the controller accepts the new connection from the new AP model. The new AP model then joins the controller .

3. Commit the new AP software to be persistent across reloads.

   Example

   ```
   Device# install commit
   ```

   The commit occurs after activation when the system is up, or after the first reload. If a package is activated but not committed, it remains active after the first reload but becomes inactive after the second reload.

4. (Optional) Deactivate an active APDP, update the package status, and trigger a process to restart or reload.

   Example

   ```
   Device# install deactivate file bootflash filename
   ```

5. Display the image version on the device.

   Example

   ```
   Device# show version
   ```

Verify APDP on the Controller

Verify the status of APDP packages on the controller and the embedded wireless controller.

To verify the status of APDP packages on the controller , use the command:

```
Device# show install summary

[ Chassis 1 ] Installed Package(s) Information:
State (St): I - Inactive, U - Activated & Uncommitted, C - Activated &
Committed, D - Deactivated & Uncommitted
--------------------------------------------------------------------------------
Type  St   Filename/Version
--------------------------------------------------------------------------------
APDP  I    bootflash:apdp_CSCvp12345.bin
IMG   C    17.1.0.0
--------------------------------------------------------------------------------
Auto abort timer: inactive
--------------------------------------------------------------------------------
```

Note

The output of this command varies based on the packages, and the package states that are installed.

Per site or per AP model service pack (APSP)

Explains per site or per AP model service packs (APSPs) as targeted software maintenance updates for access points, enabling administrators to control and tailor update deployment by site or device model.

A per-site and per-AP model service pack is a software maintenance update option that:

- allows administrators to roll out critical AP updates and bug fixes to subsets of APs at specific sites or to particular AP models
- enables staggered and targeted deployment (by site or AP model) to control the propagation of a SMU in your network by selecting which sites to include in the SMU activation.
- integrates with FlexConnect, local mode, and SD-Access wireless deployments for selective upgrade strategies.

Ensure all sites are at the same SMU level before you roll out a new SMU to selected sites or start an image upgrade on the system.

With per-AP model SMU, you can update only specific AP models. The software downloads and activates only on chosen AP models at a site. If you include specific model images in a SMU, you must also include software images for those models in future updates.

Note

After you apply the AP site filter for a per-site SMU upgrade, install a new image only if you have applied the site filter to all sites, or if you have removed the current site filter.

Restrictions and workflow

Restrictions for configuring APs to a primary controller

- If you do not configure APs to a primary controller, all APs receive the same discovery response from controllers with and without the APSP image. This causes your APs to switch repeatedly between two controllers.

Workflow of AP SMU Upgrade

- To upgrade the AP SMU, run a query to check for ongoing activities, such as AP image predownload or AP rolling upgrade.
- Identify the sites for SMU installation and set up a site filter.
- Trigger the SMU predownload for the sites in the site filter.
- Activate the SMU after the predownload completes.
- Commit the update.

Note

After you set up the site filter, you can add more sites. To make your changes take effect, apply the filter again using the **ap image site-filter file** *file-name* **apply** command. If you clear the site filter, updates are applied to all remaining sites. When you deactivate or roll back images, the action is applied to all sites, not just filtered sites.

Rolling AP upgrade

Rolling AP upgrade details.

A rolling AP upgrade is an upgrade method that

- updates APs in staggered batches such that some APs are always up in the network and provide seamless coverage to clients while the other APs are selected to be upgraded, and
- requires preloading the image in advance to ensure that all APs scheduled for upgrade have the new version.

Note

The time needed to complete a rolling AP upgrade depends on several factors: the number of APs, the percentage in each iteration, the controller type, and the controller's connectivity to the APs. To estimate the upgrade time, multiply the number of iterations by the maximum iteration time. Each iteration can take up to 5 minutes multiplied by the expected number of iterations. You can view the end time by checking the **iteration expiry time** field in the **show ap upgrade** command.

Rolling AP upgrade process

Rolling AP upgrade process.

Rolling AP upgrade is done on a per controller basis. The number of APs to be upgraded at a given time is a percentage of the total number of APs connected to the controller. The percentage is capped at a user configured value. The default percentage is 15. Upgrade all non-client APs before upgrading any client APs.

The key components involved in the process are:

- Controller
- APs
- Clients

The process involves the following stages:

1. Candidate AP set selection:

   In this stage, a set of AP candidates is selected based on neighboring AP information. For example, if you identify an AP for upgrade, a certain number (N) of its neighbors are excluded from candidate selection. The N values are generated in the following manner:

   If the user configurable capped percentage is 25%, then N=6 (Expected number of iterations =5)

   If the user configurable capped percentage is 15%, then N=12 (Expected number of iterations=12)

   If the user configurable capped percentage is 5%, then N=24 (Expected number of iterations =22)

   If you cannot select candidates using neighboring AP information, select candidates from indirect neighbors. If this still fails, the AP is upgraded without failure.

   Note

   If you have more candidate APs than the configured percentage allows, remove the extra candidates to maintain the percentage cap.

2. Client steering:

   Move clients connected to candidate APs to other APs before rebooting the candidate APs. Each AP sends a request to its associated clients, providing a list of the best available APs (excluding candidate APs). The system marks candidate APs as unavailable for neighbor lists and resets these markings during the AP rejoin and reload process.

3. AP rejoin and reload process:

   After steering clients, if any clients remain connected to the candidate AP, de-authorize those clients and reload the AP so it restarts with a new image. Set a three-minute timer for the APs to rejoin. When the timer expires, check if all candidate APs have joined the controller or mobility peer. If 90% have joined, end the iteration. If not, extend the timer by three minutes and check again. After three attempts, start the next iteration. Each iteration usually lasts about 10 minutes.

   You only need to configure the number of APs to upgrade at a time as a percentage of the total APs in the network. The default is 15 percent.

```
Device(config)# ap upgrade staggered {25 | 15 | 5}
```

Use the command to trigger the rolling AP upgrade:

```
Device# ap image upgrade [test]
```

Note

Rolling AP upgrade does not resume after an SSO. To restart it, use the **ap image upgrade** command. This restarts the upgrade for all APs, including Mesh APs.

Install AP service package (GUI)

Install AP service package using GUI.

Use the GUI to install a new or updated AP service package on your device when needed.

Before you begin

Procedure

1. Choose **Administration** > **Software Management**.
2. Click the **AP Service Package (APSP)** tab.
3. Click **Add**.
4. From the **Transport Type** drop-down list, choose the transfer type to transfer the software image to your device as TFTP, SFTP, FTP, Device, or Desktop (HTTP).
   a) If you choose **TFTP** as the **Transport Type**, enter the **Server IP Address (IPv4/IPv6)**, **File path**, and select **File System** from the drop-down list.
   b) If you choose **SFTP** as the **Transport Type**, enter the **Server IP Address (IPv4/IPv6)**, **SFTP Username**, **SFTP Password**, **File path**, and select **File System** from the drop-down list.
   c) If you choose **FTP** as the **Transport Type**, enter the **Server IP Address (IPv4/IPv6)**, **FTP Username**, **FTP Password**, **File path**, and select **File System** from the drop-down list.
   d) If you choose **Device** as the **Transport Type**, enter the **File path** and select **File System** from the drop-down list.
   e) If you choose **Desktop (HTTPS)** as the **Transport Type**, select **File System** from the drop-down list, and click **Select File** to navigate to the **Source File Path**.
5. Enter the **File Name** and click **Add File**.
6. From the **AP Upgrade Configuration** section, select the percentage of APs to include from the **AP Upgrade per iteration** drop-down list.
7. Click **Apply**.

Install AP service package (CLI)

Install AP service package using commands.

Use this procedure to selectively update APs with SMU. Apply image updates only to APs that are filtered by site tags using CLI.

Procedure

1. Check for ongoing activities such as AP image predownload or AP rolling upgrade.

   Example

   ```
   Device# install add file flash filename
   ```

   If there are no such activities, populate the predownload directory to install a package file to the system.

2. Add a site tag to a site filter.

   Example

   ```
   Device# ap image site-filter file filename flash add site-tag bgl18
   ```

3. Optional: Remove a site tag from a site filter.

Example

```
Device# ap image site-filter file filename flash remove site-tag bgl18
```

4. Optional: Perform predownload of an AP image.

   Example

   ```
   Device# ap image predownload
   ```

   This image predownload is filtered by the site filter, set up in the previous step.

5. Trigger the AP upgrade in a rolling, staggered fashion for the APs added in the site filter.

   Example

   ```
   Device# install activate file filename flash
   ```

6. Commit the image update.

   Example

   ```
   Device# install commit
   ```

   During the commit, the mapping from file to site is saved in the persistent database. This ensures the mapping remains available after a reload.

Add a site to a filter (CLI)

Add a site to a filter using commands.

Add a site tag to a site filter to control which APs receive image pre-downloads and upgrades.

Procedure

1. Add a site tag to a site filter.

   Example

   ```
   Device# ap image site-filter file filename flash add site-tag bgl18
   ```

2. Predownload the image and upgrade the APs based on the site filter.

   Example

   ```
   Device# ap image site-filter file filename flash apply
   ```

3. Clear the site filter table, predownload the image, and perform a rolling AP upgrade to all sites where it is not active.

Example

```
Device# ap image site-filter file flash filename clear
```

Deactivate an image (CLI)

Deactivate an image using commands.

Remove or deactivate a specific AP image from all sites using CLI.

Procedure

Perform rolling AP upgrade based on the AP models present in the prepare file.

Example

```
Device# install deactivate file flash filename
```

Deactivation is not limited by site and applies to all sites.

> **Note**
>
> The system takes action only if the APs in a site are not running the SMU that is being deactivated.

Roll back APSP (CLI)

Roll back APSP using commands.

Revert AP images to a previous software state by using rollback profiles and image predownload support.

Procedure

1. Optional: Move to any rollback point using AP image predownload support.

   Example

   ```
   Device# install add profile rollback_profile-name rollback_id1
   ```

   > **Note**
   >
   > To get a list of available rollback profile names, use the **show install profile** command.

2. Optional: Perform predownload of an AP image.

Example

```
Device# ap image predownload
```

This image predownload is filtered by the site filter, which you set up earlier.

3. Perform rollback of the image for the affected AP models.

Example

```
Device# install rollback to rollback_id rollback_id1
```

The roll back action is not filtered by site. Therefore, rollback applies to all the sites.

> **Note**
>
> APs that are in the base image, or at a point before the rollback action takes effect, are not affected.

Cancel the Upgrade (CLI)

Cancel the upgrade using commands.

Stop an in-progress or pending software upgrade on your device using CLI.

Procedure

Abort the upgrade by resetting the APs in a rolling fashion.

Example

```
Device# install abort
```

Verify the upgrade

Verify the upgrade using commands.

To see the summary of the AP software install files, enter the command:

```
Device# show ap image file summary

AP Image Active List
============================
Install File Name: base_image.bin
------------------------------
AP Image Type    Capwap Version  Size (KB)         Supported AP models
-------------    --------------
-----------------------------------------------------------------------
     ap1g1       17.3.0.30        13300   NA
```

```
     ap1g2       17.3.0.30        34324   NA

     ap1g3       17.3.0.30        98549   AP803

     ap1g4       17.3.0.30              34324   AP1852E, AP1852I, AP1832I,
AP1830I, AP1810W, OEAP1810

     ap1g5       17.3.0.30              23492   AP1815W, AP1815T, OEAP1815,
AP1815I, AP1800I, AP1800S, AP1815M, 1542D, AP1542I, AP1100AC, AP1101AC, AP1840I


     ap1g6       17.3.0.30        93472               AP2900I, C9117AXI

    ap1g6a       17.3.0.30        247377              C9130AXI, C9130AXE,
C9140AXI, C9140AXD, C9140AXT

     ap1g7       17.3.0.30        23988               AP1900I, C9115AXI,
AP1900E, C9115AXE, C9120AXE, C9120AXP, C9120AXI

     ap1g8      17.3.0.30       23473   C9105AXI, C9105AXW, C9110AXI, C9110AXE


     ap3g1       17.3.0.30        23422   NA

     ap3g2       17.3.0.30        23411   AP1702I

     ap3g3       17.3.0.30        23090   AP3802E, AP3802I, AP3802P, AP4800,
AP2802E, AP2802I, AP2802H, AP3800, AP1562E, AP1562I, AP1562D, AP1562PS,
IW-6300H-DC, IW-6300H-AC, IW-6300H-DCW, ESW-6300

     c1570       17.3.0.30        13000   AP1572E, 1573E, AP1572I

     c3700       17.3.0.30        14032   AP3702E, AP3701E, AP3701I, AP3702I,
 AP3701P, AP3702P, AP2702E, AP2702I, AP3702, IW3702, AP3701, AP3700C

 virtApImg       17.3.0.30                    177056        APVIRTUAL


AP Image Prepare List**
============================
Install File Name: base_image.bin
------------------------------
============================
Install File Name: base_image.bin
------------------------------
AP Image Type    Capwap Version  Size (KB)         Supported AP models
-------------    --------------
---------------------------------------------------------------------
     ap1g1       17.3.0.30     13300             NA

     ap1g2       17.3.0.30     34324             NA

     ap1g3       17.3.0.30     98549             AP803

     ap1g4       17.3.0.30           34324             AP1852E, AP1852I,
AP1832I, AP1830I, AP1810W, OEAP1810

     ap1g5       17.3.0.30           23492             AP1815W, AP1815T,
OEAP1815, AP1815I, AP1800I, AP1800S, AP1815M, 1542D, AP1542I, AP1100AC,
AP1101AC, AP1840I

     ap1g6       17.3.0.30     93472                         AP2900I, C9117AXI
```

```
   ap1g6a       17.3.0.30     247377                      C9130AXI, C9130AXE,
 C9140AXI, C9140AXD, C9140AXT

     ap1g7       17.3.0.30     23988                   AP1900I, C9115AXI,
AP1900E, C9115AXE, C9120AXE, C9120AXP, C9120AXI

     ap1g8       17.3.0.30     23473            C9105AXI, C9105AXW, C9110AXI,
 C9110AXE

     ap3g1       17.3.0.30     23422             NA

     ap3g2       17.3.0.30     23411             AP1702I

     ap3g3       17.3.0.30     23090             AP3802E, AP3802I, AP3802P,
AP4800, AP2802E, AP2802I, AP2802H, AP3800, AP1562E, AP1562I, AP1562D, AP1562PS,
 IW-6300H-DC, IW-6300H-AC, IW-6300H-DCW, ESW-6300

     c1570       17.3.0.30     13000             AP1572E, 1573E, AP1572I

     c3700       17.3.0.30     14032             AP3702E, AP3701E, AP3701I,
AP3702I, AP3701P, AP3702P, AP2702E, AP2702I, AP3702, IW3702, AP3701, AP3700C

 virtApImg       17.3.0.30                 177056            APVIRTUAL

**The difference between the Active and Prepare lists identifies images being
 predownloaded to APs.
```

To see the summary of the AP site-filtered upgrades, enter the command:

```
Device# show ap image site summary

Install File Name: vwlc_apsp_16.11.1.0_74.bin

Site Tag                                    Prepared         Activated
  Committed
-------------------------------------------------------------------------------
bgl-18-1                                    Yes              Yes
  Yes
bgl-18-2                                    Yes              Yes
  Yes
bgl-18-3                                    Yes              Yes
  Yes
default-site-tag                            Yes              Yes
  Yes
```

To see the summary of AP upgrades, enter the command:

```
Device# show ap upgrade summary
```

To check the status of an APSP, enter the command:

```
Device# show install summary

[ Chassis 1 ] Installed Package(s) Information:

State (St): I - Inactive, U - Activated & Uncommitted,

C - Activated & Committed, D - Deactivated & Uncommitted
```

```
---------------------------------------------------------------------------

Type St Filename/Version

---------------------------------------------------------------------------

APSP I bootflash:vwlc_apsp_16.11.1.0_74.bin

IMG C 16.11.1.0.1249



---------------------------------------------------------------------------

Auto abort timer: inactive

---------------------------------------------------------------------------
```

Verify AP upgrade on the controller

Verify AP upgrade on the controller using commands.

Use the **show** command to verify the AP upgrade on the controller:

```
Device #show ap upgrade

AP upgrade is in progress
From version: 8 16.9.1.6
To version: 9 16.9.1.30
Started at: 03/09/2018 21:33:37 IST
Percentage complete: 0
Expected time of completion: 03/09/2018 22:33:37 IST
Progress Report
---------------
Iterations
----------
Iteration Start time End time AP count
-------------------------------------------------------------------------
0 03/09/2018 21:33:37 IST 03/09/2018 21:33:37 IST 0
1 03/09/2018 21:33:37 IST ONGOING 0
Upgraded
--------
Number of APs: 0
AP Name Ethernet MAC Iteration Status
-------------------------------------------------------------------------
In Progress
-----------
Number of APs: 1
AP Name Ethernet MAC
-----------------------------------------------
APf07f.06a5.d78c f07f.06cf.b910
Remaining
---------
Number of APs: 3
AP Name Ethernet MAC
-----------------------------------------------
APCC16.7EDB.6FA6 0081.c458.ab30
AP38ED.18CA.2FD0 38ed.18cb.25a0
AP881d.fce7.5ee4 d46d.50ee.33a0
```

16 Efficient Image Upgrade and Out-of-band AP Image Download

Topics:

- Efficient image upgrades for FlexConnect APs
- Out-of-band AP image download

Efficient image upgrades for FlexConnect APs

Introduces Efficient Image Upgrade capabilities, outlining restrictions, the image upgrade download process, and instructions for enabling pre-download through the GUI to streamline wireless network software updates.

Efficient image upgrade is an optimized upgrade method for FlexConnect APs that

- reduces WAN link load by limiting simultaneous image downloads to three secondary APs per primary AP.
- allows simultaneous secondary AP downloads from a designated primary AP, and
- enables faster image pre-downloads for FlexConnect APs.

Feature History

Release	Feature	Feature Information
Cisco IOS XE 17.11.1	Out-of-Band AP Image Download	This feature enhances the AP image upgrade method to make upgrades faster and more flexible. Commands Introduced: • **ap upgrade method https** • **ap file-transfer https port** • **show ap upgrade method**

Analogy: relay race

Efficient image upgrades work like a relay race for file sharing. Imagine a group of runners (APs) at a remote location needing a baton (the software image) from the main building (the controller). Instead of each runner making the trip directly and crowding the hallway (the WAN link), one designated runner (the primary AP) gets the baton first and brings it back. The others then take the baton from this runner in controlled small groups (three at a time), so the hallway never gets overcrowded, saving time and reducing congestion.

Restrictions

- Efficient image upgrades work only in FlexConnect (flex mode).
- Ensure all FlexConnect APs are grouped under the same site tag, and are physically co-located.
- Efficient image upgrades do not operate when the default-site-tag is applied.
- You cannot remove flex profile configurations from a site tag that is configured as a site.
- Use the **no local-site** command to prepare a site tag for Flexconnect. Otherwise, the Flex profile configuration is not applied.
- Efficient AP image download uses port 8443 for HTTPS connections. The listener also uses this port to upload client debug bundles and transfer Clean Air spectral recordings from the AP to the controller. This port remains open even if you disable efficient AP image download because multiple services use it.

Efficient image upgrade download

Efficient AP image download minimizes WAN load, accelerates image distribution for co-located FlexConnect APs, and ensures reliable and consistent image deployment within site tags.

Summary

The key components involved in the process are:

- **Controller**: Hosts and distributes AP images through WAN.
- **Primary AP**: Selected for each model per site tag to fetch the image from the controller.
- **Secondary APs**: Retrieve the image from the primary AP through TFTP. Up to three secondary APs can download simultaneously.

Workflow

These stages describe the efficient AP image download:

1. The controller selects one AP per model and site tag as the primary AP.
2. The primary AP downloads the required image directly from the controller over WAN.
3. Up to three secondary APs download the image simultaneously through TFTP from the primary AP. This approach minimizes WAN usage.
4. The process uses port 8443 for HTTPS connections. This port is also used for uploading debug bundles and spectral recordings.
5. The port remains open, as multiple services use this listener even if efficient AP image download is disabled.

Result

Efficient AP image download minimizes WAN load, accelerates image distribution for co-located FlexConnect APs, and ensures consistent, reliable image deployment within site tags.

Enable pre-download in a FlexConnect AP (GUI)

Enable Pre-Download using GUI.

Allow access points to download software images in advance to minimize downtime during upgrades.

Use this task when you want APs to pre-download image updates through the GUI.

Follow these steps to enable pre-download for an AP:

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. In the **Access Points** window, expand the **All Access Points** section and click the name of the AP to edit.
3. In the **Edit AP** window, click the **Advanced** tab and from the **AP Image Management** section, click **Predownload**.
4. Click **Update & Apply to Device**.

The selected AP downloads the software image in advance, reducing downtime during the upgrade process.

Configure an efficient image upgrade for FlexConnect APs

To configure pre-download, you must complete the described series of tasks:

1. Create a flex profile using the CLI and enable pre-download. You can also use an existing flex profile. See *Enable pre-download in a flex profile (CLI)*.
2. Configure a site tag. Site tags group APs and apply common settings, such as the flex profile you created in the previous task. See *Configure a site tag for FlexConnect APs (CLI)*.

3. Attach the policy tag and site tag to the AP. This task configures APs with the correct policy and site tag. See *Attach policy tag and site tag to an AP (CLI).*
4. Trigger pre-download for a site tag to start the image pre-download to the APs associated with the site tag. You can then verify the pre-download status. See *Trigger and verify pre-download to a site tag*.

Configure a flex profile for pre-download (CLI)

Enable Pre-Download using commands.

Enable image pre-downloading for remote APs in a FlexConnect flex profile from the CLI.

Use this task to pre-download software images on APs associated with a flex profile. This accelerates upgrade rollouts and reduces network downtime.

Before you begin

- Identify the target flex profile to be updated.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a flex profile and enter the flex profile configuration mode.

   Example

   ```
   Device(config)# wireless profile flex rr-xyz-flex-profile
   ```

3. Enable pre-download of the image.

   Example

   ```
   Device(config-wireless-flex-profile)# predownload
   ```

4. Exit the configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-flex-profile)# end
   ```

Pre-download is enabled for the selected flex profile. APs in this profile download the image before upgrades.

What to do next

Monitor AP status to verify successful pre-download before scheduling the upgrade.

Configure a site tag for FlexConnect APs (CLI)

Configure a Site Tag using CLI.

Set up a site tag for wireless deployments to enable site-specific configurations using the CLI.

Before you begin

Gather required information, such as the desired site tag name and flex profile name.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a site tag and enter site tag configuration mode.

   Example

   ```
   Device(config)# wireless tag site rr-xyz-site
   ```

3. Configure a flex profile.

   Example

   ```
   Device(config-site-tag)# flex-profile rr-xyz-flex-profile
   ```

   - You cannot remove flex profile configurations from a site tag that is configured as a local site.

     Use the **no local-site** command to remove local site configurations before applying flex profile configurations.

4. Add a description for the site tag.

   Example

   ```
   Device(config-site-tag)# description "default site tag"
   ```

5. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

   Example

   ```
   Device(config-site-tag)# end
   ```

6. Optional: Display the number of site tags.

   Example

   ```
   Device# show wireless tag site summary
   ```

The site tag is configured on the device, and site-specific settings are applied.

Verify site tag configuration

Verify the site tag configuration:

- To view detailed information about a site, use the **show wireless tag site detailed** *site-tag-name* command.
- To view default tag (site-tag) type when both site tag and policy tag are not configured, use the output of the **show wireless loadbalance tag affinity wncd** *wncd-instance-number* command.

Attach policy tag and site tag to an AP (CLI)

Attach Policy Tag and Site Tag to an AP using commands.

Assign a policy tag and site tag to an AP using the CLI.

Use this procedure to associate specific network policies and locations with an AP in your Cisco wireless deployment.

Before you begin

Make sure you have the wired MAC address of the AP.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a Cisco AP and enters AP profile configuration mode.

   Example

   ```
   Device(config)# ap F866.F267.7DFB
   ```

   > **Note**
   >
   > The *mac-address* should be a wired mac address.

3. Map a policy tag to the AP.

   Example

   ```
   Device(config-ap-tag)# policy-tag rr-xyz-policy-tag
   ```

4. Map a site tag to the AP.

   Example

   ```
   Device(config-ap-tag)# site-tag rr-xyz-site
   ```

5. Associate the RF tag.

Example

```
Device(config-ap-tag)# rf-tag rf-tag1
```

6. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

Example

```
Device(config-ap-tag)# end
```

7. Optional: Display AP details and the tags associated to it.

Example

```
Device# show ap tag summary
```

8. Display the AP name with tag information.

Example

```
Device# show ap name "ap-name" tag info
```

9. Optional: Display the AP name with tag details.

Example

```
Device# show ap name ap-name tag detail
```

The AP is now associated with the specified policy, site, and optionally, RF tags you selected

Trigger and verify pre-download to a site tag

Trigger Predownload to a Site Tag using commands.

Start and confirm an image pre-download for a specific site tag by using CLI

Use this task to distribute a new software image to primary and secondary APs in advance of a full deployment.

Before you begin

Identify the site tag to which you want to pre-download the image.

Procedure

---

1. Enter the privileged EXEC mode.

Example

```
Device> configure terminal
```

2. Instruct the primary APs to start image predownload.

Example

```
Device# ap image predownload site-tag rr-xyz-site start
```

3. Display the list of primary APs per AP model per site tag.

Example

```
Device# show ap primary list
```

4. Display the pre-downloading state of primary and secondary APs.

Example

```
Device# show ap image
```

> **Note**
>
> To check if efficient image upgrade is enabled in the AP, use the **show capwap client rcb** command on the AP console.

Example

The output displays the primary AP.

```
Device# show ap primary list
AP Name                          WTP Mac           AP Model           Site Tag

-------------------------------------------------------------------------------------
AP0896.AD9D.3124                 f80b.cb20.2460    AIR-AP2802I-D-K9 ST1
```

The output shows that the primary AP has started pre-downloading the image.

```
Device# show ap image
Total number of APs: 6

AP Name              Primary Image   Backup Image  Predownload Status
Predownload Version  Next Retry Time   Retry Count
-------------------------------------------------------------------------------------
APE00E.DA99.687A    16.6.230.37     0.0.0.0       None                0.0.0.0
              N/A                0
AP188B.4500.4208    16.6.230.37     8.4.100.0     None                0.0.0.0
              N/A                0
AP188B.4500.4480    16.6.230.37     0.0.0.0       None                0.0.0.0
              N/A                0
AP188B.4500.5E28    16.6.230.37     16.4.230.35   None                0.0.0.0
              N/A                0
AP0896.AD9D.3124    16.6.230.37     8.4.100.0     Predownloading
16.6.230.36          0                 0
```

```
AP2C33.1185.C4D0    16.6.230.37     8.4.100.0     None                0.0.0.0
              N/A               0
```

The output shows that the primary AP has completed pre-download and the pre-download has been initiated in the secondary APs.

```
Device# show ap image

Total number of APs: 6
AP Name              Primary Image   Backup Image  Predownload Status
Predownload Version  Next Retry Time   Retry Count
-------------------------------------------------------------------------------
APE00E.DA99.687A    16.6.230.37     0.0.0.0       Initiated
16.6.230.36          N/A               0
AP188B.4500.4208    16.6.230.37     8.4.100.0     None                0.0.0.0
              N/A               0
AP188B.4500.4480    16.6.230.37     0.0.0.0       None                0.0.0.0
              N/A               0
AP188B.4500.5E28    16.6.230.37     16.4.230.35   None                0.0.0.0
              N/A               0
AP0896.AD9D.3124    16.6.230.37     8.4.100.0     Complete
16.6.230.36          0                 0
AP2C33.1185.C4D0    16.6.230.37     8.4.100.0     Initiated
16.6.230.36          0                 0
```

The output shows image status of a particular AP.

```
Device# show ap name APe4aa.5dd1.99b0 image
AP Name : APe4aa.5dd1.99b0
Primary Image : 16.6.230.46
Backup Image : 3.0.51.0
Predownload Status : None
Predownload Version : 000.000.000.000
Next Retry Time : N/A
Retry Count : 0
```

The output shows pre-download completion on all APs.

```
Device# show ap image
Total number of APs: 6

Number of APs
        Initiated                 : 0
        Predownloading            : 0
        Completed predownloading  : 3
        Not Supported             : 0
        Failed to Predownload     : 0

AP Name              Primary Image   Backup Image  Predownload Status
Predownload Version  Next Retry Time   Retry Count
-------------------------------------------------------------------------------
APE00E.DA99.687A    16.6.230.37     16.6.230.36   Complete
16.6.230.36            N/A               0
AP188B.4500.4208    16.6.230.37     8.4.100.0     None                0.0.0.0
              N/A               0
AP188B.4500.4480    16.6.230.37     0.0.0.0       None                0.0.0.0
              N/A               0
AP188B.4500.5E28    16.6.230.37     16.4.230.35   None                0.0.0.0
```

```
                N/A                0
AP0896.AD9D.3124    16.6.230.37     16.6.230.36   Complete
16.6.230.36           0                0
AP2C33.1185.C4D0    16.6.230.37     16.6.230.36   Complete
16.6.230.36           0                0
```

Out-of-band AP image download

Explains the out-of-band access point image download process, including key concepts and implementation details for managing AP firmware updates independently of the primary network connection.

Out-of-band AP image download is an enhanced upgrade method that

- moves AP image downloads out of the CAPWAP control path using an enhanced webserver (nginx) on the controller
- improves upgrade speed and flexibility compared to traditional in-band methods, and
- falls back to CAPWAP method automatically if the out-of-band download fails.

Out-of-band download advantages

In WLAN deployments, APs traditionally gather their software image and configuration from the controller (in-band) during the join, predownload, and upgrade phases over the CAPWAP control path. This mechanism has limitations in the context of CAPWAP window size, processing of CAPWAP packets, and parallel image downloads. With image upgrade being a significant activity in the lifecycle of APs, upgrades become a time-consuming activity when the deployment size increases, especially for remote deployments, because the image always comes from the controller, irrespective of the deployment types.

To make upgrades faster and more flexible, the AP image upgrade method is enhanced in Cisco IOS XE Dublin 17.11.1 release. An enhanced webserver (nginx) running on the controller helps the AP image downloads to be available out of the CAPWAP path (out of band).

Key considerations for out-of-band AP image download:

- HTTPS configuration done at the global level applies to all the APs joining the controller.
- When AP image download over an Out-of-Band method fails, the download falls back to the CAPWAP method, as a result of which the APs will not be stranded.
- AP image download over HTTPS may fail if the HTTPS server Trustpoint has a chain of CA certificates.
- Before you downgrade from Cisco IOS XE Dublin 17.11.1 to an earlier version, ensure that the Out-of-Band AP Image Download feature is disabled, as it is not supported in previous releases.

Feature history for out-of-band AP image download

Lists release and related information for out-of-band AP image download feature.

This table provides release and related information for the feature explained in this module.

This feature is available in all the releases subsequent to the one in which it is introduced in, unless noted otherwise.

Table 20: Feature history for out-of-band AP image download

Feature	Release	Feature Information
Out-of-Band AP Image Download	Cisco IOS XE Dublin 17.11.1	The AP image upgrade method is enhanced to make the upgrades faster and more flexible.

Restrictions for Out-of-Band AP image download

Outlines platform restrictions for Out-of-Band AP Image Download feature.

This feature is not supported on the following platforms:

- Cisco Embedded Wireless Controller on Catalyst Access Points
- Cisco Embedded Wireless Controller on Catalyst Switches
- Cisco Wave 1 Access Points

Download AP image from controller using HTTPS (CLI)

Configure AP image download from the controller using HTTPS using commands.

Configure APs to download images from the controller over HTTPS for efficient out-of-band image updates.

APs can download images efficiently from the controller using HTTPS when the AP supports out-of-band download methods. This feature reduces upgrade time and network overhead compared to traditional CAPWAP-based image transfers.

Before you begin

- HTTPS configuration must be enabled.
- The nginx server must be running on the controller. Use the **show platform software yang-management process** command to check whether the nginx server is running.
- The custom-configured port must be reachable between the controller and the corresponding AP.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the corresponding AP to download the image over HTTPS from the controller.

   Example

   ```
   Device(config)# ap upgrade method https
   ```

   This command configures the AP to use out-of-band AP image download method if the AP supports it.

   You can check whether the AP supports efficient download method using the **show AP config general** command.

   Use the **no** form of this command to disable out-of-band AP image download method.

3. Configure a custom port for image download from the nginx server running on the controller.

   Example

   ```
   Device(config)# ap file-transfer https port port-number
   ```

   Example

   ```
   Device(config)# ap file-transfer https port 8445
   ```

   For HTTPS port, the valid values range from 0 to 65535, with a default of 8443. You cannot use port 443 for AP file transfers because it is the default port used for other HTTPS requests. Also, avoid configuring standard and well-known ports because the configuration may fail.

   By default, the Efficient AP image download feature uses port 8443 for HTTPS. If the same port is configured for HTTPS access for controller GUI, then GUI access will not work. In such instances, use a port number other than 8443 for controller GUI Access or configure a different port for AP file transfer over HTTPS instead of 8443.

   The port 8443 is customizable. A sample config is given below:

   ```
   Source= wireless controller
   Destination= Access Point
   Protocol=HTTPS
   Destination Port=8443
   Source Port=any
   Description= "Out of Band AP Image Download"
   ```

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The AP is now configured to download images from the controller using HTTPS over the specified port, enabling efficient out-of-band image updates.

Download AP Image from Controller Using HTTPS (GUI)

Configure AP image download from the Controller using HTTPS through the GUI interface.

Enable efficient AP image upgrade by configuring HTTPS method for out-of-band file transfer from the controller to access points.

The HTTPS method provides an efficient out-of-band file transfer mechanism for AP image upgrades. This configuration allows access points to download images directly from the controller using HTTPS protocol on a designated port.

Procedure

---

1. Choose **Configuration > Wireless > Wireless Global**.
2. In the **AP Image Upgrade** section, enable the **HTTPS Method** to allow image download on APs from the controller, over HTTPS.

> **Note**
>
> The AP should support out-of-band image download. You can verify this in the **Configuration > Wireless > Access Points** window. Select the AP, and in the **Edit AP > Advanced** tab, view the details of the support in the **AP Image Management** section.

3. Enter the **HTTPS Port** to designate AP file transfers on that port.

   Valid values range from 0 to 65535, with the default being 8443. Note that you cannot use port 443 for AP file transfers because that is the default port for other HTTPS requests.

   By default, the Efficient AP image download feature uses port 8443 for HTTPS. If the same port is configured for HTTPS access for controller GUI, then GUI access will not work. In such instances, use a port number other than 8443 for controller GUI Access or configure a different port for AP file transfer over HTTPS instead of 8443.

4. Click **Apply to Device** to save the configuration.

The HTTPS method for AP image download is now enabled and configured. Access points that support out-of-band image download can now efficiently download images from the controller using the specified HTTPS port.

Verify image upgrade

Provides verification commands for out of band image download status and methods.

Verify out of band image download capabilities, status, and methods used by access points for image upgrades.

To check whether an AP supports efficient download method, use this command:

```
Device# show ap config general

Cisco AP Name : AP002C.C862.E880
=================================================

Cisco AP Identifier : 002c.c88b.0300
Country Code : Multiple Countries : IN,US
Regulatory Domain Allowed by Country : 802.11bg:-A 802.11a:-ABDN
AP Country Code : US - United States
AP Regulatory Domain
802.11bg : -A
AP Upgrade Out-Of-Band Capability : Enabled
AP statistics : Disabled
```

To view the AP image download statistics, use this command.

Use the **show AP image** command to see the detailed output.

```
Device# show ap image summary

Total number of APs  : 1
Number of APs
        Initiated                 : 0
        Downloading               : 0
        Predownloading            : 0
        Completed downloading     : 0
        Completed predownloading  : 0
```

```
        Not Supported              : 0
        Failed to Predownload      : 0
        Predownload in progress    : No
```

To view the method used to download the AP image, use this command:

```
Device# show wireless stats ap image-download

AP image download info for last attempt
AP Name  Count ImageSize StartTime         EndTime            Diff(secs)
Predownload Aborted  Method
-------------------------------------------------------------------------------------------
mysore1  1     40509440  08/23/21 22:17:59 08/23/21 22:19:06  67         No
        No       CAPWAP
```

To view the method used to download the AP image, use this command:

```
Device# show ap upgrade method

AP upgrade method HTTPS : Disabled
```

To view the port used for the AP image transfer, use this command:

```
Device# show ap file-transfer https summary


       Configured port                : 8443
       Operational port               : 8443

!If different ports are shown under 'Configured port' and 'Operations port'
!that means custom port configuration has failed and is continuing with the
previous port.
!The failure reason could be the input port, which is a well-known port and
already in use.
```

To view the whether an AP supports image download over HTTPS, use this command:

```
Device# show ap name AP2800 config general | sec Upgrade

AP Upgrade Out-Of-Band Capability               : Enabled
```

To view the detailed output an AP's pre-image, use this command:

```
Device# show ap image

Total number of APs  : 2
Number of APs
        Initiated                  : 0
        Downloading                : 0
        Predownloading             : 0
        Completed downloading      : 2
        Completed predownloading   : 0
        Not Supported              : 0
        Failed to Predownload      : 0
```

```
        Predownload in progress    : No
AP Name    Primary Image Backup Image Predownload Status Predownload Version
Next Retry Time Retry Count Method
------------------------------------------------------------------------------
AP_3800_1  17.11.0.69    17.11.0.71   None               0.0.0.0
N/A             0           HTTPS
AP2800     17.11.0.69    17.11.0.71   None               0.0.0.0
N/A             0           HTTPS

!The 'method' column indicates the download method used by the AP.
```

17 Predownloading an Image to an Access Point

Topics:

- Image pre-downloads to AP
- AP image download time enhancement

Image pre-downloads to AP

Introduces AP image pre-download concepts, detailing feature history and outlining restrictions that impact image deployment to access points on controllers.

An image pre-download is a software distribution process that

- downloads an upgrade image to an AP from the device,
- minimizes network outages and allows continuous connectivity during the image download process, and
- ensures the AP can join and register with the controller immediately after the upgrade.

Additional reference information

Previously, you could download an upgrade image to the device and reset it. This AP then entered discovery mode. After discovering the controller with the new image, the AP downloads the new image, resets itself, enters discovery mode, and rejoins the device.

You can now download the upgrade image to the controller. When the controller is operational with the upgrade image, the AP joins the controller and moves to **Registered** state, because the AP image has been predownloaded to the AP.

Feature history for AP pre-download

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 21: Feature history for AP pre-download

Release	Feature Information
Cisco IOS XE Bengaluru 17.5.x	This feature now has GUI and YAML support. For more information, see the Programmability Configuration Guide, Cisco IOS XE Bengaluru 17.5.x.
Cisco IOS XE Gibraltar 16.11.1	This feature minimizes network outages, downloads an upgrade image to an access point from the device without resetting the access point or losing network connectivity. This feature has CLI support only.

Restrictions and best practices for pre-downloading an image to an AP

Restrictions for pre-downloading an image to an AP.

Restrictions

- You can pre-download images for up to 100 APs per Wireless Network Control Daemon (WNCD) instance on the controller when using the default CAPWAP method, and up to 50 APs per WNCD instance when using the HTTPS upgrade method. For the Catalyst 9800-L model, image pre-download is limited to 25 APs per WNCD instance for CAPWAP method.

  The controller initiates pre-download requests periodically until the maximum number of APs allowed for that interval is reached. The trigger interval is every ten seconds for CAPWAP-based upgrades and every one second for HTTPS-based upgrades.

- If an AP has only 16 MB total available memory, it may not have enough free memory to download an upgrade image. To free up space, the AP may automatically delete crash information files, radio files, and backup images. Pre-downloading is not affected because the pre-download image replaces any backup images on the AP.
- Ensure that all primary, secondary, and tertiary controllers run the same images. If the images differ, the feature does not work.
- Each AP can store only two software images.
- The controller supports only self-signed certificates and does not support Cisco certificates. If AP join fails after you move APs between controllers, run the **capwap ap erase all** command to remove the hash string from the APs.
- If the latest upgrade image is already on the AP, pre-download is not triggered. Check whether the primary and backup image versions are the same as the upgrade image, using the **show ap image** command.

Known behavior

- Predownload of the AP image is based on the AP model rather than the image type.
- If an AP remains in the predownloading state after an SSO event, enter:
  - **ap image predownload abort**
  - **clear ap predownload stats**
- When you upgrade Cisco Wave 1 APs from Cisco AireOS Release 8.3 to Cisco IOS XE Gibraltar 16.10.1, the AP may download the image twice. This can increase downtime during migration.

Best Practices

- Before you reset the controller, ensure all APs have completed downloading the image.
- To view cumulative statistics about AP images in the controller, you must first use the **clear ap predownload statistics** command to clear the statistics and ensure that the correct data is displayed. Then, use the **show ap image** command to view cumulative statistics about AP images in the controller,
- During AP image pre-download, the WNCD CPU may reach 99 percent. This is normal and does not cause crashes or disconnect problems for clients or APs.

Procedures

Provides step-by-step procedures for predowloading images to access points using CLI, GUI, and YANG interfaces, and explains monitoring of the pre-download process to ensure deployment success.

Pre-download an image to AP (CLI)

Pre-download an Image to Access Points using commands.

Minimize downtime during controller upgrades by pre-downloading and activating a new software image on connected APs.

Use this task if you want to push an image to APs before an upgrade, minimizing downtime during switchover.

For more information, see this video.

Before you begin

- Ensure the controller is in install mode.

- Confirm the AP model exists in the new capability XML file on the controller and maintained internally in the controller. With the appropriate modification of the capability XML through a controller upgrade, APSP, or APDP installation, the controller can override the existing AP image for a particular model.
- Copy the new image either from the TFTP server, flash image, or USB.

Procedure

1. Add and expand the controller software image.

   Example

   ```
   Device# install add file bootflash:image.bin
   ```

2. Download the new image to all the APs or a specific AP connected to the device.

   Example

   ```
   Device# ap image predownload
   Device# ap name ap1 image predownload
   ```

3. Verify the AP's pre-download status.

   Example

   ```
   Device# show ap image
   ```

   This command initially displays the status as Predownloading and then moves to Completed, when download is complete.

4. Display image details of a particular AP.

   Example

   ```
   Device# show ap name myapname image
   ```

5. Swap images of the APs that have completed pre-download.

   Example

   ```
   Device# ap image swap
   ```

   ```
   Device# ap name myapname image swap
   ```

   ```
   Device# ap image swap completed
   ```

> **Note**
>
> You can swap the AP images using **ap image swap** command even without pre-downloading a new image to the AP and there are no restrictions or prerequisites to swap the image.

6. Run compatibility checks, install the package, and update the package status details

   Example

   ```
   Device# install activate
   ```

   For restartable packages, the command triggers the appropriate post-install scripts to restart the necessary processes,

   For non-restartable packages, the command triggers a reload.

   > **Note**
   >
   > This step reloads the complete controller stack. If HA is used, both primary and secondary controllers.

7. Commit the activation changes to be persistent across reloads.

   Example

   ```
   Device# install commit
   ```

   .

   The commit can be done after activation while the system is up, or after the first reload. If the package is activated but not committed, it remains active after the first reload, but not after the second reload.

The new software image is downloaded and activated on all designated APs

What to do next

Monitor AP connectivity and operation to confirm successful upgrade.

Abort AP pre-download (CLI)

Abort an AP image pre-download using CLI commands.

Stop an AP image pre-download that is in progress.

Use this task when you need to stop an AP image pre-download before it completes.

Before you begin

- Ensure that you have access to privileged EXEC mode.

- Use the **show ap image** command to verify the AP image pre-download status.

Procedure

1. Verify the AP image pre-download status.

   Example

   ```
   Device# show ap image
   ```

   Verify that one or more access points have a pre-download status of **Predownloading**.

2. Abort the AP image pre-download.

   Example

   ```
   Device# ap image predownload abort
   ```

   > **Note**
   >
   > This command stops the current AP image pre-download operation. It does not reboot the access points or swap their images.

3. Verify the AP image status after aborting the pre-download.

   Example

   ```
   Device# show ap image
   ```

   Verify that the access points are no longer in the **Predownloading** state.

4. If you need to initiate another pre-download, clear the pre-download state and start the pre-download again.

   Example

   ```
   Device# clear ap predownload stats
   Device# ap image predownload
   ```

   > **Note**
   >
   > Run **clear ap predownload stats** only when you need to initiate another pre-download after aborting the current operation.

The AP image pre-download is aborted.

What to do next

To cancel an AP image upgrade, use the **ap image upgrade abort** command instead.

Pre-download an image to AP (GUI)

Pre-download an Image to Access Points using GUI.

Before you begin

- Ensure the Upgrade mode of the controller is in install mode.

You can minimize downtime during controller upgrades by pre-downloading and activating a new software image on connected APs using the connector GUI.

You can also see the video demonstration.

Procedure

1. Choose **Administration** > **Software Management** and click the **Software Upgrade** .
2. Set necessary parameters:
   a) Select a **Transport Type**.
   b) Select a **File Path**.
   c) Select **Source File Path**.
3. Check the **AP Image Predownload** check box.

   If you already have an inactive image file on your device, a dialog box prompts you to remove the unused image and proceed with the latest image download.
4. Click **Download and Install** to initiate the upgrade process.

   You can view and verify the pre-download progress in the **Status** dialog box. You can also check the progress log by clicking the **Show Logs** icon.
5. When the pre-download is completed, click **Save Configuration and Activate**.
6. Click **Yes** to confirm the activation.

   This step runs compatibility checks, installs the package, and updates the package status details. After a successful activation, the AP reloads. If there are uncommitted files, the system prompts you to remove those.
7. Click **Commit** to complete the upgrade process.

The new image is pre-downloaded, and activated on the APs. The APs reload with upgraded software.

Pre-download an image to AP (YANG)

Prepare an AP for a software upgrade or deployment by pre-downloading an image using YANG automation.

Use YANG with NETCONF and RESTCONF to automate and program network operations

Procedure

Send the NETCONF remote procedure call (RPC) call using your management client to initiate the image pre-download. Replace `<uuid>` with the correct value for your environment

```
<rpc xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="101">
  <set-rad-predownload-all
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-access-point-cmd-rpc">
    <uuid>12312341231234</uuid>
  </set-rad-predownload-all>
</rpc>
```

The image is pre-downloaded to the access point, making it available for upgrade or installation.

What to do next

Refer to the Cisco IOS XE Programmability Configuration Guide and YANG Data Models on Github website for more details.

Contact the Developer Support Community for NETCONF or YANG features.

Monitor AP pre-download

Monitor the AP Pre-download from controller.

This section describes the commands for monitoring AP pre-download from the controller.

Verify the AP pre-download status and progress from the controller:

```
        Controller# show ap image
        Total number of APs  : 1

        Number of APs
        Initiated                   : 1
        Predownloading              : 1
        Completed predownloading    : 0
        Not Supported               : 0
        Failed to Predownload       : 0

        AP Name                           Primary Image   Backup Image
Predownload Status    Predownload Ver...  Next Retry Time    Retry Count

---------------------------------------------------------------------------------------------------

        AP1                               10.0.1.66       10.0.1.66
Predownloading        10.0.1.67           NA                          0
```

```
        Controller# show ap image

        Total number of APs  : 1

        Number of APs
        Initiated                   : 1
        Predownloading              : 0
```

```
        Completed predownloading   : 1
        Not Supported              : 0
        Failed to Predownload      : 0

        AP Name                            Primary Image   Backup Image
Predownload Status    Predownload Ver...  Next Retry Time    Retry Count

------------------------------------------------------------------------------------------------------

        AP1                                10.0.1.66       10.0.1.67
Complete              10.0.1.67           NA
        0
```

Verify the image details of a particular AP:

```
        Controller# show ap name APe4aa.5dd1.99b0 image

        AP Name : APe4aa.5dd1.99b0
        Primary Image : 10.0.1.66
        Backup Image : 10.0.1.67
        Predownload Status : None
        Predownload Version : 000.000.000.000
        Next Retry Time : N/A
        Retry Count : 0
```

AP image download time enhancement

Describes AP Image Download Time Enhancement for OEAP or Teleworker deployments.

AP image download time enhancement is a feature that

- adds support to multiple sliding windows for control packets going from controller to AP
- reduces image download time in Cloud and FlexConnect deployments where the AP and controller are separated by a WAN or Internet link, and
- is supported only on OEAP profiles.

CAPWAP communication and window enhancement

The wireless controller and the access point (AP) communicate with each other using CAPWAP. The CAPWAP has two channels, namely control and data. The control channel is used to send configuration messages, download images and client keys, or the context to the AP. The control channel has a single window in the current implementation. A single window means that every message that is sent from the controller has to be acknowledged by the AP. The next control packet is not transmitted till the earlier one is acknowledged by the AP.

The AP Image Download Time Enhancement feature adds support to multiple sliding windows for control packets going from controller to AP. The sliding window can be set to N (static) instead of a single window. The request queue size is decided based on the maximum window size the AP supports.

The following table shows the recommended window size for different network conditions:

Table 22: Recommended window size

Link Bandwidth[1]	Less than 200 ms RTT	Greater than 200 ms RTT
More than 20 Mbps	10	15
Between 5 and 20 Mbps	10	15
Between 1 and 5 Mbps	5	10
Less than 1 Mbps	3	5

Note

- The window size can be changed only during the AP join process.
- All image upgrades should be in the **install** mode for faster upgrade. Image upgrade should be done from the **one-shot** command to include OEAP predownload.
- Configure the window size only for AP profiles that are exclusively used for Teleworker or Office Extend Access Points (OEAP).
- An AP reload is not required after disabling this feature.
- This feature is supported only on the OEAP profiles.
- GUI does not support AP predownload. Therefore, the AP downloads after disjoining the controller during CAPWAP join phase. This causes a long disruption in the network as the Image download for AP can take upto one hour.

Important

If you downgrade the software to Cisco IOS XE Gibraltar 16.12.4 or earlier from Cisco IOS XE Amsterdam 17.3.1, you should reset the CAPWAP multi window to a single window prior to the downgrade. Failure to do so necessitates a manual AP recovery.

High-level workflow for AP Image Download Time Enhancement:

1. Select an existing AP join profile or create a new one.
2. Set the CAPWAP window size.
3. Associate the AP join profile to an existing site tag or new one.
4. Apply the site tag to the AP using: Static, Filter, Location, AP, or Default mapping method.

Configure AP Image Download Time Enhancement (GUI)

Configure AP Image Download Time Enhancement using GUI to optimize access point image download performance.

[1] The window size recommendation provided in the table is for packet loss of less than one percent (< 1%). If the network supporting the CAPWAP link has packet loss of more than one percent (> 1%), use a smaller value for window size. For good links with round-trip time (RTT) of about 100ms and packet drops of less than half a percent (< 0.5%), use a window size of up to 20 for better performance.

AP Image Download Time Enhancement optimizes the download performance of access point images by configuring CAPWAP window size settings to improve data transfer efficiency.

Use this procedure when you need to optimize access point image download performance through CAPWAP configuration settings. This enhancement is particularly useful in environments with high-latency network connections or when managing large numbers of access points.

Follow these steps to configure AP Image Download Time Enhancement using the GUI:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join** > **CAPWAP > Advanced**.
2. In the **CAPWAP Window Size** field, enter the unit of measurement of the window.
3. Click **Save & Apply to Device**.

The AP Image Download Time Enhancement configuration is applied to the device. Access points will use the optimized CAPWAP window size settings for improved image download performance.

Configure AP Image Download Time Enhancement (CLI)

Configure AP Image Download Time Enhancement using commands.

Enable AP Image Download Time Enhancement to optimize the download process for access points.

AP Image Download Time Enhancement improves the efficiency of firmware downloads to access points by configuring the CAPWAP control packet transmit queue size. This configuration is particularly beneficial for teleworker or OEAP deployments.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile.

   Example

   ```
   Device(config)# ap profile ap-profile-name
   ```

   Example

   ```
   Device(config)# ap profile capwap_multiwindow
   ```

3. Configure the AP CAPWAP control packet transmit queue size.

   Example

   ```
   Device(config-ap-profile)# capwap window size window-size
   ```

Example

```
Device(config-ap-profile)# capwap window size 20
```

Note

Configure the window size only for AP profiles that are exclusively used for teleworker or OEAP.

Be aware that any change in window size may impact other APs.

For standard AP deployments, retain the default CAPWAP window size of `1`. Configure a higher value only for AP profiles used exclusively for Teleworker or OfficeExtend APs. If APs repeatedly disconnect and rejoin the controller or experience CAPWAP keepalive timeouts, reset the value to `1` using `no capwap window size` or `capwap window size 1`.

4. Return to privileged EXEC mode.

Example

```
Device(config-ap-profile)# end
```

The AP Image Download Time Enhancement is now configured with the specified CAPWAP window size, optimizing firmware download performance for the designated AP profile.

Verify AP image download time enhancement configuration

Provides commands to verify AP Image Download Time Enhancement Configuration.

Verify AP Image Download Time Enhancement Configuration using commands to view CAPWAP window size, status, and modes.

To view the CAPWAP window size present in an AP profile, use this command:

```
Device# show ap profile name default-ap-profile detailed | in wind

Capwap window size : 10
```

To view the CAPWAP status and modes, use this command:

```
Device# show capwap client rcb

OperationState                      : UP
Name                                : AP4001.7A39.2D5A
MwarHwVer                           : 0.0.0.0
Location                            : default location
ApMode                              : Remote Bridge
ApSubMode                           : Not Configured
CAPWAP Path MTU                     : 1485
Software Initiated Reload Reason    : Reload command
CAPWAP Sliding Window
Active Window Size                  : 10
Last Request Send To Application    : 184
```

```
Expected Seq Num                    : 185
Received Seq Num                    : 184
Request Packet Count                : 42424
Out Of Range Packets Count          : 0
Window Moved  Packets Count         : 0
In Range Packets Count              : 960
Expected Packets Count              : 41464
```

To view the AP configuration details, including the CAPWAP window size, use this command:

```
Device# show ap config general | in Wind

Capwap Active Window Size                         : 5
Capwap Active Window Size                         : 10
Capwap Active Window Size                         : 1
```

18 N+1 Hitless Rolling AP Upgrade

Topics:

- N+1 hitless rolling AP upgrade and version skew
- Site-based rolling AP upgrade in N+1 network
- Client steering enhancements

N+1 hitless rolling AP upgrade and version skew

N+1 hitless rolling AP upgrade and version skew are network upgrade strategies that enable wireless network upgrades with minimal or no downtime.

N+1 hitless rolling AP upgrade and version skew are network upgrade strategies that

- enable wireless network upgrades with minimal or no downtime
- allow software mismatches between wireless controllers and access points during upgrades, and
- support staged AP upgrades while maintaining service continuity.

Version skew: A compatibility feature that allows a controller to operate with APs running a different software version.

N+1 Hitless Rolling AP Upgrade: A controller-assisted method that migrates APs to a temporary controller during upgrade, avoiding service disruption.

Understanding N+1 Hitless Rolling AP Upgrade and Version Skew

Traditionally, all APs and the controller needed to be on the same software version to function properly, causing network downtime during upgrades. Version skew removes this limitation by allowing staged upgrades. Combined with the N+1 hitless rolling AP upgrade process, this strategy ensures continuous service by migrating APs to a temporary controller, upgrading the original controller, and then migrating APs back.

This method is especially beneficial in large-scale deployments where network availability is critical.

The following table contrasts the features of a traditional upgrade withvVersion skew + N+1 Upgrade:

Table 23: Comparison of Traditional Upgrade vs. Version Skew + N+1 Upgrade

Feature	Traditional Upgrade	Version Skew + N+1 Upgrade
Requires same version on all devices	Yes	No
Causes downtime	Yes	No
Supports staged AP upgrades	No	Yes
Uses secondary controller	No	Yes (WLC2 as mobility member)
Needs image pre-download	Not applicable	Yes, for ap image upgrade command

Examples of N+1 Hitless Rolling AP Upgrade

- A network admin can upgrade the controller (WLC1) while APs continue to operate from a mobility member (WLC2).
- Using the **`ap image move`** command after predowloading images avoids downtime even without simultaneous AP upgrades.

Counter-examples for N+1 Hitless Rolling AP Upgrade

- Upgrading all APs simultaneously without version skew or N+1 support would cause network downtime.
- Using the **`ap image upgrade`** command without pre-downloading the image will fail.

How N+1 hitless rolling AP upgrades work

This DITA XML content outlines the process for an N+1 hitless rolling AP upgrade.

This process involves upgrading controllers and managing access points using specific commands.

Summary

The N+1 hitless rolling AP upgrade process involves several critical steps: establishing mobility tunnels, upgrading software, moving APs, and activating new software images.

This process ensures a seamless upgrade of APs with minimal service disruption.

Workflow

Workflow for the N+1 hitless rolling AP upgrade feature

1. Establish a mobility tunnel from the controller (WLC1) to a mobility member (WLC2).
2. Upgrade the controller software (WLC1) using the **install add file bootflash:***new_version.bin* command.
3. You can also choose to upgrade the AP image.. For more information, see Predownloading an Image to an Access Point chapter.
4. Use the **ap image upgrade destination** *controller-name controller-ip report-name* privileged EXEC command to upgrade and move all the APs from WLC1 (source) to WLC2 (destination).
5. Activate the new image in WLC1 using the **install activate** command.
6. Commit the changes using the **install commit** command.
7. Move the APs back to WLC1 from WLC2 using the **ap image move destination** *controller-name controller-ip report-name* command.

> **Note**
>
> The **ap image upgrade destination** command does not work without an image pre-download. If you do not perform an image pre-download, use the **ap image move** command to move the APs. When APs download the image and join the destination controller, you must set the iteration time as high. Also, you can customize the iteration time by configuring the **ap upgrade staggered iteration timeout** command.

Result

The process upgrades access points effectively without disrupting service. It maintains optimal network functionality during transition stages.

Configure hitless upgrade

Configure hitless upgrade using commands.

The purpose of this task is to perform a hitless upgrade using a zero downtime network upgrade process in an N+1 deployment. This process ensures that all necessary components are pre-checked for compliance, and the upgrade is carried out smoothly with minimal disruption.

Before you begin

- Provide the hostname and wireless management IP of the destination controller in the privileged EXEC command.
- Predownload the image on APs running on the destination controller.

Procedure

1. Move the APs to the specified destination controller with the Swap and Reset command.

   Example

   ```
   Device# ap image upgrade destination wlc2 10.7.8.9
   ```

   After this, the parent controller activates the new image and reloads with the new image. Establish the mobility tunnel, and then return APs to the parent controller without a swap and reset.

   > **Note**
   >
   > Ensure that you establish a mobility tunnel from controller (WLC1) to a mobility member (WLC2) before image upgrade.

2. Optional: Moves APs to the specified destination controller with a swap and reset command.

   Example

   ```
   Device# ap image upgrade destination wlc2 10.7.8.9
   ```

   > **Note**
   >
   > Steps 2 to 4 should only be performed if Step 1 is not being executed.

3. Move the APs back to the parent controller.

   Example

   ```
   Device# ap image move destination wlc1 10.7.8.6
   ```

4. Optional: Move APs to the specified destination controller with a swap and reset command.

   Example

   ```
   Device# ap image upgrade destination wlc2 10.7.8.9 fallback
   ```

   After that, APs move back to the parent controller (without a swap and reset) once the new image is manually installed and the parent controller is reloaded.

5. Optional: Move APs to the specified destination controller using the Swap and Reset command.

   Example

   ```
   Device# ap image upgrade destination wlc2 10.7.8.9 reset
   ```

After this, the parent controller activates the new image and reloads with the new image.

The result of this task is a successful upgrade of the network without any downtime, ensuring the APs are returned to the parent controller seamlessly after the upgrade is complete. The network continues to function optimally with the new image installed.

Verify hitless upgrade

Verify Hitless Upgrade using commands.

Use these commands to verify hitless upgrade.

These are the commands to verify hitless upgrade.

- To view all the upgrade report names, use this command:

```
Device# show ap upgrade summary

Report Name      Start time
-------------------------------------------------------------------------------------------
AP_upgrade_from_VIGK_CSR_2042018171639 05/20/2018 17:16:39 UTC
```

- To view AP upgrade information based on the upgrade report name, use these commands:

```
Device# show ap upgrade name test-report

AP upgrade is complete
From version: 16.10.1.4
To version: 16.10.1.4
Started at: 05/20/2018 17:16:39 UTC
Percentage complete: 100
End time: 05/20/2018 17:25:39 UTC
Progress Report
---------------
Iterations
----------
Iteration Start time End time AP count
-------------------------------------------------------------------------------------------
0 05/20/2018 17:16:39 UTC 05/20/2018 17:16:39 UTC 0
1 05/20/2018 17:16:39 UTC 05/20/2018 17:25:39 UTC 1
Upgraded
--------
Number of APs: 1
AP Name Ethernet MAC Iteration Status
-------------------------------------------------------------------------------------------
AP-SIDD-CLICK 70db.9848.8f60 1 Joined
In Progress
-----------
Number of APs: 0
AP Name Ethernet MAC
-----------------------------------------------
Remaining
---------
Number of APs: 0
AP Name Ethernet MAC
-----------------------------------------------
```

Site-based rolling AP upgrade in N+1 network

Introduces site-based rolling AP upgrade in N+1 networks, detailing prerequisites, restrictions, use cases, upgrade workflows, CLI configuration, hitless software upgrades, and verification procedures for optimal AP migration and network reliability.

The site-based rolling AP upgrade in an N+1 network is a feature that

- effectively achieves a zero-downtime network upgrade in an N+1 network
- allows you to perform a software upgrade of a site or all the sites managed by the controller, and
- upgrades the software of the APs belonging to a site and monitors the network to see whether it is functioning as intended before adding more sites to the site filter.

Feature history for site-based rolling AP upgrade in N+1 networks

This table provides release and related information for the features explained in this module.

These features are available in all releases subsequent to the one they were introduced in, unless noted otherwise.

Table 24: Feature history

Release	Feature	Feature Information
Cisco IOS XE 17.9.1	Site-Based Rolling AP Upgrade in N+1 Network	This feature helps to achieve a zero downtime network upgrade in N+1 networks.

> **Note**
>
> If the upgrade fails to meet the objectives, all the sites in the site filter can be removed using the **ap image site-filter file any-image remove-all** command.

The **ap image site-filter** command is modified to include the **any-image** keyword that is a substitute for the image file name to support the N+1 AP move site filter.

Prerequisites for site-based rolling AP upgrade in N+1 networks

Prerequisites for site-based rolling AP upgrade in N+1 networks.

Ensure these conditions are met for site-based rolling AP upgrade in N+1 networks

- The source and destination controllers should be in the same mobility group (preferably running the latest image) but with different AP image versions.
- Image of the destination controller should be available on the source controller.
- Both the source and destination controllers should be in INSTALL mode.
- If you reboot the source or destination controller during the N+1 upgrade, you must re-execute the procedure.

Restrictions for site-based rolling AP upgrade in N+1 networks

Restrictions for site-based rolling AP upgrade in N+1 networks

- Site filter operations are supported only for N+1 upgrade and N+1 move; **fallback** and **reset** options of the **ap image upgrade destination** command are not supported.
- You can move APs between controllers that have the same software.
- The keywords **any** and **remove-all** of the **ap image site-filter** command work only for the N+1 procedures: AP upgrade or move. It will not work for other site filter operations such as AP Model Service Pack (APSP) or AP Device Package (APDP).

Best practices for N+1 redundancy deployments

Use Cases for N+1 redundancy deployments.

Use spare controllers in N+1 redundancy deployments to allow APs to fail over when the primary controller becomes non-operational. This approach ensures minimal network downtime, typically lasting 30 to 40 seconds, as APs discover and join the network in local mode.. Ensure all devices reboot and converge efficiently within the N+1 deployment framework to achieve effective zero-downtime during network upgrades.

Configure site-based N+1 upgrade (CLI)

Configure site-based N+1 upgrade using commands.

This document provides a reference on configuring site-based N+1 upgrades using CLI commands. The configuration involves several critical settings and actions which ensure seamless transitions and effective management of access points across controllers.

These are the key commands and configuration steps

- Configure Iteration settings for N+1 Upgrade on Primary Controller
- Configure Site Filters for Predownload and Move APs to Spare Controller
- Return APs (Per-Site) to Primary Controller using Site Filters

Configure iteration settings for N+1 upgrade on source controller (CLI)

N+1 upgrade and move to destination controller using commands.

This task involves executing a sequence of CLI commands to set iteration parameters for conducting an N+1 upgrade on the source controller.

Before you begin

- See the *Prerequisites for Site-based Rolling AP Upgrade in an N+1 Network* section
- Run all the commands only on the source controller
- To disable the 11v message, use **no ap upgrade staggered client-steering** command.

Procedure

1. Enter global configuration mode

Example

```
Device# configure terminal
```

2. Optional: Disable client steering

   Example

   ```
   Device# no ap upgrade staggered client-steering
   ```

3. Optional: Configure the minimum percentage of APs that must join the destination controller to signal iteration completion.

   Example

   ```
   Device(config)# ap upgrade staggered iteration completion 50
   ```

4. Optional: Configure the action to be taken when APs are missing after an iteration

   Example

   ```
   Device(config)# ap upgrade staggered iteration error action stop
   ```

5. Optional: Configure the maximum time allowed per iteration during AP upgrade.

   Example

   ```
   Device(config)# ap upgrade staggered iteration timeout 18
   ```

   Valid values range from 9 to 60.

6. Return to privileged EXEC mode

   Example

   ```
   Device(config)# exit
   ```

The APs upgrade and move to the designated destination controller within the mobility group. Any configured site filters and settings will be applied, ensuring that the APs operate under optimal conditions as defined during the configuration process.

Configure site filters for predownload and move APs to destination controller (CLI)

N+1 upgrade and move to destination controller using commands.

The purpose of this task is to guide users through configuring site filters and performing a controlled predownload of AP images, ensuring a seamless upgrade and transition to the designated destination controller within an N+1 network environment.

Before you begin

See the *Prerequisites for Site-based Rolling AP Upgrade in an N+1 Network* section.

- Run all the commands only on the source controller.

Procedure

1. Add a site tag to a site filter

   Example

   ```
   Device# ap image site-filter any-image add site1
   ```

   You can repeat this step to set up a multisite filter.

2. Move the APs to a different controller in the mobility group.

   Example

   ```
   Device# ap image move destination controller2 10.9.34.4
   ```

   Note

   It is preferable to move the APs to a different controller running the same image.

   Wait for the upgrade to complete.

   If upgrade is not completed successfully, you can use the **ap image upgrade destination** or **ap image move destination** commands to restart the upgrade process.

3. Add additional site tag to a site filter

   Example

   ```
   Device# ap image site-filter file any-image add site2
   ```

4. Predownloads the image and upgrades the APs based on the site filter.

   Example

   ```
   Device# ap image site-filter file any-image apply
   ```

   Note

   Wait for the upgrade to complete.

5. Optional: Clear the site filter table and predownloads the image and does a rolling AP upgrade to all the sites.

   Example

   ```
   Device# ap image site-filter file any-image clear
   ```

6. Optional: Remove all the site filters

Example

```
Device# ap image site-filter file any-image remove-all
```

Upon completion of this task, APs will be upgraded and transitioned to the selected destination controller, operating under optimized conditions with all site filter configurations applied. This process ensures robust performance and continuity within the mobility group.

Return APs (per-site) to source controller using site filters

N+1 move to destination controller using commands.

This task will help to manage and transition APs in an N+1 network environment by adding site tags, moving APs to controllers, and upgrading the APs based on defined site filters.

Run all the commands only on the source controller.

Before you begin

See the *Prerequisites for Site-based Rolling AP Upgrade in an N+1 Network* section.

Procedure

---

1. Add a site tag to a site filter.

   Example

   ```
   Device# ap image site-filter any-image add site1
   ```

2. Move the APs back to the parent controller.

   Example

   ```
   Device# ap image move destination controller2 10.9.34.2
   ```

   Note

   Wait for the upgrade to complete.

3. Add an additional site tag to a site filter.

   Example

   ```
   Device# ap image site-filter any-image add site2
   ```

4. Upgrade the APs based on the site filter.

   Example

   ```
   Device# ap image site-filter any-image apply
   ```

> **Note**
>
> Wait for the upgrade to complete.
>
> If upgrade is not completed successfully, use the **ap image upgrade destination** or **ap image move destination** command to restart the upgrade process.

5. Optional: Clear the site filter table and predownloads the image, and performs a rolling AP upgrade to all sites where it is not active.

Example

```
Device# ap image site-filter any-image clear
```

Completing this task successfully ensures efficient management of APs within the N+1 network environment. All APs are upgraded and operate with the correct site tags and configurations.

Upgrade workflow for site-based N+1 upgrade

Upgrade workflow for site-based N+1 upgrade

The site-based N+1 upgrade process enhances network reliability by utilizing spare controllers, ensuring seamless software updates without disrupting ongoing service and supporting continuous uptime in enterprise environments.

Summary

Hitless software upgrade uses the concept of N+1 high availability using a spare controller to upgrade the CAPWAP infrastructure comprising controllers and APs. Depending on your choice, the APs are upgraded in a staggered fashion either per site or on all sites using the rolling AP upgrade feature, thereby avoiding network disruption. This ensures that the clients are serviced by neighboring APs while one or selected APs undergo the upgrade process.

Workflow

1. Initiate upgrade on the source controller. You can choose to upgrade all sites or per site based on your preference.
2. Move the APs to the destination controller. APs are upgraded in a staggered fashion using the rolling AP upgrade algorithm.
3. After all the APs move to the destination controller through multiple iterations, activate the target image on the source controller.
4. The source controller reloads for the new image to take effect.
5. (Optional) Move the APs back to the source controller using the cli commands.

Result

The process ensures N+1 availability with uninterrupted service to clients while successfully updating AP infrastructure.

Configure site-based N+1 upgrade (GUI)

Hitless software upgrade using GUI.

This task provides instructions to configure a site-based N+1 upgrade using the GUI on a controller. This ensures that the upgrade process is seamless and minimizes downtime for APs and controllers.

Before you begin

- The controller should be in **INSTALL** mode.
- The controller should be paired with another controller and both should be part of the same mobility group.

  The spare controller should be upgraded with the target image.

Procedure

1. Choose **Administration > Software Management** .
2. From the **Transport Type** drop-down list, choose an option.

   Note

   In the **File Path** field, enter the complete path from where you want to download the software image file, including the name of the file.

   - If you choose **My Desktop** as the transport type, click **Select File** to navigate to the file from the **Source File Path** field.
   - If you choose **SFTP** as the transport type, enter the source IP address, SFTP username, SFTP password, file path, and select the destination.
   - If you choose **FTP** as the transport type, enter the source IP address, FTP username, FTP password, file path, and select the destination.
   - If you choose **TFTP** as the transport type, enter the source IP address, file path, and select the destination.

     Note

     In controllers, the IP TFTP source is mapped to the service port by default.

   - If you choose **Device** as the transport type, choose the file system and file path.
3. Check the **Enable Hitless Upgrade** check box to allow the APs and the controller to be upgraded.
4. From the **Site Filter** drop-down list, choose **All Sites** or one or more **Custom Sites**.

   In case you choose to upgrade for **All Sites**, you can optionally enable **Fallback after Upgrade** so that the APs move back to the parent controller after the new image has been activated and the parent controller has reloaded.

   In case you choose a **Custom Site**, select the site from the **Site Tags** drop-down list. In this case, the APs do not move back to the parent controller automatically and you will have to manually move them using CLIs.
5. In the **Controller IP Address (IPv4/IPv6)** field, enter the source controller's IPv4/IPv6 address.
6. In the **Controller Name** field, enter the source controller's name.
7. In the **AP Upgrade Configuration** section, use the **AP Upgrade per Iteration** drop-down list to select the percentage of APs to be upgraded per iteration. This configures the minimum percentage of APs that must join the destination controller to signal completion of iteration.
8. Check the **Client Steering** check box to move clients attached to APs undergoing an upgrade to other APs. If the clients still persist on the candidate APs, they are disconnected and the APs will reload with the new image.
9. In the **Accounting Percentage** field, choose the percentage of APs that should join the destination controller after each iteration (of the staggered AP upgrade) to consider the iteration as successful. The default value is 90%.

10. Tap to select the type of **Accounting Action** to configure for the APs. If you enable **Terminate**, the upgrade is terminated if the configured percentage of APs does not join the mobility peer, and a notification is sent via Syslog message. If you choose **Ignore**, the upgrade continues irrespective of whether the configured percentage of APs are joining the controller or not.
11. In the **Iteration Expiry** field, select the number of minutes from the drop-down list to configure the expiry time for each iteration.
12. Click **Download & Install**.
13. Click **Save Configuration & Activate**.
14. Click **Commit** to make the activation changes persistent across reloads.

Upon successful completion of the upgrade process, the controllers and APs should operate with the new software image with minimal disruption to network operations. The reliability and performance of the network are maintained during the transition.

Verify site-based rolling AP upgrade in a N+1 network

Verify site-based rolling AP upgrade in a N+1 network using commands.

The commands ensure the successful implementation and seamless operation of site-based rolling AP upgrades within an N+1 network. By utilizing these commands, you will be able to monitor the progress, evaluate the status, validate updates, and troubleshoot potential issues related to AP upgrades.

Use these AP commands to check the progress of the upgrade and debugging

- **show ap summary**
- **show ap tag summary**
- **show ap status**
- **show wireless mobility summary**
- **show ap image**
- **show ap upgrade**
- **show ap upgrade site**
- **show ap upgrade site summary**
- **show ap upgrade name** *report-name*
- **show wireless mobility ap-list**

To view the summary of all the connected APs, use this command

```
Device# show ap summary

Number of APs: 8

AP Name             Slots  AP Model           Ethernet MAC    Radio MAC
 Location           Country  IP Address    State
-------------------------------------------------------------------------------
AP00D7.8F9A.43DE    2      AIR-AP2802I-D-K9   00d7.8f9a.43de  002c.c8df.3ca0
 default location    IN       10.9.48.254  Registered
AP4C77.6D21.9098    2      AIR-AP2802E-N-K9   4c77.6d21.9098  00be.7573.b340
 default location    IN       10.10.10.52  Registered
AP00F2.8B27.BB2C    2      AIR-AP2802I-D-K9   00f2.8b27.bb2c  0896.ad9b.f9e0
 default location    IN       10.9.44.51   Registered
```

```
APA023.9F41.5A38    2      AIR-AP2802I-D-K9   a023.9f41.5a38  1880.90f4.7b00
 default location    IN       10.10.10.51  Registered
AP00A3.8E4A.762C    2      AIR-AP2802I-D-K9   00a3.8e4a.762c  1880.90f5.14e0
 default location    IN       10.9.48.54   Registered
AP40CE.2485.D616    2      AIR-AP3802I-D-K9   40ce.2485.d616  4001.7aca.5960
 default location    IN       10.9.50.42   Registered
AP40CE.2485.D62C    2      AIR-AP3802I-D-K9   40ce.2485.d62c  4001.7aca.5aa0
 default location    IN       10.10.10.53  Registered
AP2C57.4188.4BC4    3      C9130AXE-D         2c57.4188.4bc4  cc7f.75a8.78e0
 default location    IN       10.9.34.207  Registered
```

To view the summary of all the access points with policy tags, use this command

```
Device# show ap tag summary
Number of APs: 8

AP Name             AP Mac           Site Tag Name     Policy Tag Name
RF Tag Name       Misconfigured    Tag Source
-------------------------------------------------------------------------------
AP00D7.8F9A.43DE    00d7.8f9a.43de   site3              default-policy-tag
default-rf-tag    No               Static
AP4C77.6D21.9098    4c77.6d21.9098   site3              default-policy-tag
default-rf-tag    No               Static
AP00F2.8B27.BB2C    00f2.8b27.bb2c   site3              default-policy-tag
default-rf-tag    No               Static
APA023.9F41.5A38    a023.9f41.5a38   default-site-tag   default-policy-tag
default-rf-tag    No               Default
AP00A3.8E4A.762C    00a3.8e4a.762c   site1              default-policy-tag
default-rf-tag    No               Static
AP40CE.2485.D616    40ce.2485.d616   site2              default-policy-tag
default-rf-tag    No               Static
AP40CE.2485.D62C    40ce.2485.d62c   site2              default-policy-tag
default-rf-tag    No               Static
AP2C57.4188.4BC4    2c57.4188.4bc4   default-site-tag   default-policy-tag
default-rf-tag    No               Default
```

To view the status of the access points, use this command

```
Device# show ap status
AP Name                                Status     Mode              Country
-------------------------------------------------------------------------------
AP00A3.8E4A.762C                       Enabled    Local             IN
AP00D7.8F9A.43DE                       Enabled    Monitor           IN
AP00F2.8B27.BB2C                       Enabled    Local             IN
AP2C57.4188.4BC4                       Enabled    Local             IN
AP40CE.2485.D616                       Enabled    Local             IN
AP40CE.2485.D62C                       Enabled    Local             IN
AP4C77.6D21.9098                       Enabled    Local             IN
APA023.9F41.5A38                       Enabled    Local             IN
```

To display the summary of the mobility manager, use this command

```
Device# show wireless mobility summary

Mobility Summary

Wireless Management VLAN: 34
Wireless Management IP Address: 10.9.34.5
```

```
Wireless Management IPv6 Address:
Mobility Control Message DSCP Value: 48
Mobility High Cipher : False
Mobility DTLS Supported Ciphers: TLS_ECDHE_RSA_AES128_GCM_SHA256,
TLS_RSA_AES256_GCM_SHA384, TLS_RSA_AES128_CBC_SHA
Mobility Keepalive Interval/Count: 10/3
Mobility Group Name: mobility-1
Mobility Multicast Ipv4 address: 10.0.0.1
Mobility Multicast Ipv6 address: ::
Mobility MAC Address: 001e.14a5.b3ff
Mobility Domain Identifier: 0x39ab

Controllers configured in the Mobility Domain:

 IP        Public Ip  MAC Address    Group Name  Multicast IPv4 Multicast IPv6
  Status  PMTU
-------------------------------------------------------------------------------
10.9.34.5  N/A        001e.14a5.b3ff mobility-1  0.0.0.0        ::
   N/A     N/A
10.9.34.2  10.9.34.2   001e.bd2d.f2ff mobility-1  0.0.0.0        ::
    Up      1385
10.9.34.3  10.9.34.3   001e.14c1.cbff mobility-1  0.0.0.0        ::
    Up      1385
10.9.34.4  10.9.34.4   001e.140e.4bff mobility-1  0.0.0.0        ::
    Up      1385
```

To view the cumulative statistics regarding the AP images in the controller, use this command

```
Device# show ap image

Total number of APs  : 8

Number of APs
        Initiated                 : 0
        Downloading               : 0
        Predownloading            : 0
        Completed downloading     : 0
        Completed predownloading  : 0
        Not Supported             : 0
        Failed to Predownload     : 0
        Predownload in progress   : No
AP Name           Primary Image Backup Image Predownload Status Predownload
Version  Next Retry Time Retry Count  Method
-------------------------------------------------------------------------------
AP00D7.8F9A.43DE  17.9.0.19     17.8.0.74    None               0.0.0.0
        N/A             0           N/A
AP4C77.6D21.9098  17.9.0.19     17.8.0.74    None               0.0.0.0
        N/A             0           N/A
AP00F2.8B27.BB2C  17.9.0.19     17.9.1.19    None               0.0.0.0
        N/A             0           N/A
APA023.9F41.5A38  17.9.0.19     17.8.0.74    None               0.0.0.0
        N/A             0           N/A
AP00A3.8E4A.762C  17.9.0.19     17.9.1.19    None               0.0.0.0
        N/A             0           N/A
AP40CE.2485.D616  17.9.0.19     17.9.1.19    None               0.0.0.0
        N/A             0           N/A
AP40CE.2485.D62C  17.9.0.19     17.8.0.82    None               0.0.0.0
        N/A             0           N/A
AP2C57.4188.4BC4  17.9.0.19     17.9.1.19    None               0.0.0.0
        N/A             0           N/A
```

To verify the AP upgrade on the controller, use this command

```
Device# show ap upgrade

AP upgrade is in progress

From version: 17.9.0.19
To version: 17.9.1.25

Started at: 01/28/2022 09:53:07 IST
Configured percentage: 5
Percentage complete: 0
Expected time of completion: 01/28/2022 13:33:07 IST

Client steering: Enabled
Iteration expiry time: 15 minutes
Accounting percentage: 95%
Accounting action: Abort

Rolling AP Upgrade Site Summary
-------------------------------
site3

Progress Report
---------------
Iterations
----------
Iteration       Start time                  End time                    AP
count
-------------------------------------------------------------------------------
0               01/28/2022 09:53:07 IST     01/28/2022 09:53:07 IST     1
1               01/28/2022 09:53:07 IST     ONGOING                     0

Upgraded
--------
Number of APs: 1
AP Name                 Radio MAC       Iteration   Status     Site
-------------------------------------------------------------------------------
AP00D7.8F9A.43DE        002c.c8df.3ca0  0           Rebooted    site3

In Progress
-----------
Number of APs: 1
AP Name                           Radio MAC
-------------------------------------------------
AP00F2.8B27.BB2C                  0896.ad9b.f9e0

Remaining
---------
Number of APs: 1
AP Name                           Radio MAC
-------------------------------------------------
AP4C77.6D21.9098                  00be.7573.b340

APs not handled by Rolling AP Upgrade
-------------------------------------
AP Name      Radio MAC           Status          Reason for not handling by
Rolling AP Upgrade
-------------------------------------------------------------------------------
```

To verify the AP upgrade information on the sites, use this command

```
Device# show ap upgrade site

Site-filtered AP upgrade report data
====================================
Source controller: Controller1
Destination controller: Controller2

From version: 17.9.0.19
To version: 17.9.1.25
Site-filters present: Yes

AP image upgrade site summary
-----------------------------
Operation: N+1 upgrade

Site Tag                                    Status
------------------------------------------------------------
site3                                       In Progress

AP upgrade reports linked to these site-filters
-----------------------------------------------

Start time               Operation type             Report name
-------------------------------------------------------------------------------
01/28/2022 09:53:07 IST  AP image upgrade/move CLI
AP_upgrade_to_DEvice2_28020229536
```

To verify the AP image upgrade site summary, use this command

```
Device# show ap upgrade site summary

AP image upgrade site summary
-----------------------------
Operation: N+1 upgrade

Site Tag                                    Status
------------------------------------------------------------
site3                                       In Progress
```

To view AP upgrade information based on the upgrade report name, use this command

```
Device# show ap upgrade name AP_upgrade_to_Device2

AP upgrade is complete

From version: 17.9.0.19
To version: 17.9.1.25

Started at: 01/28/2022 14:12:49 IST
Configured percentage: 5
Percentage complete: 100
End time: 01/28/2022 14:18:59 IST

Client steering: Enabled
Accounting percentage: 95%
Iteration expiry time: 15 minutes
Accounting action: Abort
```

```
Rolling AP Upgrade Site Summary
-------------------------------
site1
site2

Progress Report
---------------
Iterations
----------
Iteration     Start time                    End time                      AP
count
-------------------------------------------------------------------------------------
0             01/28/2022 14:12:49 IST       01/28/2022 14:12:49 IST     0
1             01/28/2022 14:12:49 IST       01/28/2022 14:15:54 IST     1
2             01/28/2022 14:15:54 IST       01/28/2022 14:18:59 IST     1

Upgraded
--------
Number of APs: 2
AP Name               Radio MAC         Iteration  Status          Site
-------------------------------------------------------------------------------------
AP40CE.2485.D616      4001.7aca.5960    1          Joined Member   site2
AP40CE.2485.D62C      4001.7aca.5aa0    2          Joined Member   site2

In Progress
-----------
Number of APs: 0
AP Name                           Radio MAC
-------------------------------------------------

Remaining
---------
Number of APs: 0
AP Name                           Radio MAC
-------------------------------------------------

APs not handled by Rolling AP Upgrade
-------------------------------------
AP Name      Radio MAC      Status    Reason for not handling by Rolling AP
Upgrade
-------------------------------------------------------------------------------------
```

To display the list of access points known to the mobility group, use this command

```
Device# show wireless mobility ap-list

AP name                            AP radio MAC      Controller IP     Learnt
from
-------------------------------------------------------------------------------
Unknown                            002c.c8df.3ca0    10.9.34.5           Self
Unknown                            00be.7573.b340    10.9.34.5           Self
Unknown                            0896.ad9b.f9e0    10.9.34.5           Self
Unknown                            1880.90f4.7b00    10.9.34.5           Self
Unknown                            1880.90f5.14e0    10.9.34.5           Self
Unknown                            4001.7aca.5960    10.9.34.5           Self
Unknown                            4001.7aca.5aa0    10.9.34.5           Self
Unknown                           687d.b45e.4b60    10.9.34.3          Mobility
 Group
Unknown                            cc7f.75a8.78e0    10.9.34.5           Self
```

Client steering enhancements

Introduces client steering enhancements that optimize client management during staggered AP upgrades, minimize unnecessary disconnections, and provide greater control over steering behavior for both new and existing clients.

Client steering enhancements are configuration options that

- improve how clients are managed during staggered access point (AP) upgrades,
- help maintain client connectivity by avoiding unnecessary disconnections, and
- offer more control over steering behavior for both new and existing clients.

Client deauthentication

This topic aims to provide a comprehensive understanding of client deauthentication during staggered upgrades of access points in a wireless network.

When access points (APs) of a wireless network are upgraded in a staggered manner, the clients connected to those APs are moved to other APs. During this period, clients that are unaware of an ongoing upgrade may try to reassociate with the same AP. Similarly, new clients may also try to join the AP. To avoid this scenario, Cisco IOS XE Dublin 17.11.1 introduces the option to not deauthenticate clients connected to the APs that are selected for the upgrade. Using the `no ap upgrade staggered client-deauth` command, you can stop deauthenticating clients before the AP performs an upgrade.

Client deauthentication affects both the 802.11v clients and non-802.11v clients. When client steering is enabled, 802.11v clients receive Basic Set Service (BSS) transition frames that guide them to roam. If client steering is disabled and deauthentication is enabled, even 802.11v clients receive deauthentication messages, which may result in dropped connections.

Best practice for preserving client connectivity during upgrade

Best practice for preserving client connectivity during upgrade

Ensure that client deauthentication is disabled on APs selected for upgrade. This minimizes disruption and maintains better client experience, especially in high-availability wireless environments.

Deauthenticate clients

Deauthenticate clients associated with the AP before an image upgrade.

This task allows you to manage client connections by deauthenticating clients associated with the access point (AP) before performing an image upgrade.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Stop deauthentication of the clients associated with the AP before the AP starts to upgrade.

   Example

   ```
   Device(config)# no ap upgrade staggered client-deauth
   ```

3. Return to privileged EXEC mode

Example

```
Device(config)# end
```

Successful execution of this task ensures that clients are not unexpectedly disconnected during an AP image upgrade.

19 NBAR Dynamic Protocol Pack Upgrade

Topics:

- Network-based application recognition protocol pack upgrade

This chapter explains how to update and manage NBAR protocol packs to extend application recognition without rebooting, describes deployment and limitations (encryption, QUIC, DoH), and provides CLI upgrade steps.

Network-based application recognition protocol pack upgrade

Introduces NBAR protocol packs, detailing their role in updating application protocol support, key components, and upgrade features that enable seamless version changes without device reloads or service disruption.

A protocol pack is a software module for Network-Based Application Recognition (NBAR) that:

- adds or updates support for application protocols without replacing the Cisco software on the device
- comes as a built-in component for each Cisco software release, and
- allows easy version upgrades and reversion without service disruption or device reload.

Protocol packs contain:

- application protocols that are compiled and officially supported by NBAR, and
- information on the signatures and attributes of the application.

Protocol packs offer these features:

- They can be loaded easily and quickly.
- You can upgrade to a later version protocol pack or revert to an earlier version.
- Device reload is not required.
- No service disruption occurs.

Application visibility controls with DPI

Deep packet inspection for application visibility and control.

A deep packet inspection (DPI) is a method that

- inspects data packets for detailed application visibility,
- allows for custom traffic definitions, and
- supports distributed analysis for wireless networks.

Wireless products use the Application Visibility and Control feature through a distributed system where Network-Based Application Recognition (NBAR) runs on controllers or access points, performs deep packet inspection, and reports via NetFlow messages.

Deployment mode

Although NBAR is supported in all the modes, protocol pack upgrade and custom applications are only supported in local mode (central switching) and in FlexConnect mode (central switching).

Important

When you upgrade the AVC protocol pack:

- Copy it to both route processors (RPs), active and standby.
- Otherwise, the standby protocol pack upgrade will fail and cause a synchronization failure crash.

NBAR limitations

Limitations of NBAR.

NBAR struggles to classify traffic when SaaS applications use certain technologies, impacting traffic classification.

- End-to-end encryption.
- Quick UDP Internet Connections (QUIC) protocol.
- DNS over HTTPS (DoH) protocol.

> **Note**
>
> In such a case, the encrypted traffic, including DoH and QUIC without SNI, limits the ability of NBAR to send the correct Protocol ID, causing issues with traffic classification.

Protocol pack upgrades and custom application configuration

Overview of protocol pack upgrades and custom application configuration.

Optimize network management by implementing protocol pack upgrades and customizing application recognition rules to meet specific traffic analysis needs.

Protocol pack upgrade

- Protocol pack upgrades allow the update of the NBAR engine to recognize new types of protocols or traffic without requiring an update to the entire switch or appliance image.
- This process eliminates the need to restart the entire system after the upgrade.
- NBAR protocol packs can be downloaded from Cisco Software Center.

Custom applications

- Configure the NBAR engine to recognize traffic based on custom rules, such as destination IP, hostname, or URL.
- The web UI and NetFlow collector display custom application names, allowing for easier identification and management.

Upgrade the NBAR2 protocol pack (CLI)

Upgrade the NBAR2 protocol pack using commands.

The purpose of this task is to upgrade the NBAR2 protocol pack on your network device. By upgrading, you enhance the application's recognition capabilities, allowing for more effective network management and traffic classification.

Upgrade the NBAR2 protocol pack by completing these steps

Before you begin

Download the protocol pack from the Software Download page and copy it into the bootflash, `#copy tftp://10.10.10.1/pp-adv-c9800-1712.1-49-70.0.0.pack bootflash:`

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Load the protocol pack.

   Example

   ```
   Device(config)# ip nbar protocol-pack bootflash:mypp.pack
   ```

Upgrading the protocol pack successfully ensures that your device can recognize a wider range of applications, enhancing traffic monitoring and classification.

Verify the NBAR2 protocol pack

This section provides details on how to verify the NBAR2 protocol pack.

To verify NBAR protocol pack version, use this command:

```
Device# show ip nbar protocol-pack active
```

20 Wireless Sub-Package for Switch

Topics:

- Wireless sub-packages

Wireless sub-packages

Introduces wireless sub-package concepts and guides users through installing, removing, and upgrading wireless packages. Details procedures for sub-package installation, single-step and multi-step methods, stack deployment, upgrading packages, deactivation, and enabling or disabling auto-upgrade functionality.

A wireless sub-package is a component within a network architecture that

- integrates seamlessly with existing network designs such as multitier, Routed Access, and Virtual Switching System (VSS)
- uses Locator/ID Separation Protocol (LISP) for control planes and Virtual Extensible LAN (VXLAN) for overlay data planes
- maintains wireless control planes through Control and Provisioning of Wireless Access Points (CAPWAP) tunnels connecting AP with controllers.

Benefits of the controller

- Centralized control simplifies configuration, troubleshooting, and upgrades.
- The controller provides high throughput and distributed forwarding for enhanced network services.
- The controller facilitates scalable services such as Application Visibility and Control (AVC) without splitting control planes into MC and MA.

Install and activate sub-packages

1. Identify the network nodes that need enhanced wireless capabilities.
2. Deploy the sub-package on top of the base image on the relevant switches.
3. Reload the system to activate the newly installed sub-package.

> **Note**
>
> The sub-package is an optional binary that contains the entire Cisco Catalyst 9800 Series Wireless Controller software.

Deployment considerations

- Provision and deploy Cisco Catalyst 9800 Wireless Controller software on Cisco Catalyst 9300 switches using Cisco Catalyst Center. Standalone controller configuration is not supported. For additional information, see Cisco Catalyst 9300 Series Switches Hardware Installation Guide.
- SNMP Compatibility: SNMP is not supported on Catalyst 9800 Embedded Wireless Controllers.
- The sub-package contains the complete Cisco Catalyst 9800 software suite, optimally deployed using Cisco Catalyst Center. For more details, see the Cisco Catalyst 9300 Series Switches Hardware Installation Guide.
- Verify that network configurations and conditions meet operational requirements to achieve optimal performance.

Install wireless packages

How to install wireless package

Summary

Installing the wireless package is essential for enabling wireless capabilities on a switch. This process allows the switch to handle wireless access points and manage network traffic efficiently, thereby expanding its functionality within a network environment.

Workflow

The process involves these stages:

1. Install the base image: load the base image (excluding wireless) onto the switch to prepare it for further installations.
2. Install the wireless package: add the wireless package to the switch to enable wireless features.
3. Upgrade the AP image: ensure that the access point image is updated to the latest version for compatibility and performance.
4. Reload the switch: restart the switch to apply the new settings and updates.
5. Enable and configure wireless: use the **configuration command** to activate wireless on the switch and configure the necessary wireless features.

Result

The switch has wireless capabilities enabled, allowing it to manage wireless connections and integrate seamlessly into a wireless network architecture.

Remove wireless package

How to remove wireless package.

Summary

Removing a wireless package involves uninstalling it, reloading the switch, and clearing configurations.

Workflow

Perform these steps to remove wireless package:

1. Uninstall: Remove the wireless package from the switch.
2. Reload: Reboot the switch to refresh system settings.
3. Clear Configuration: Run the **write** command to delete wireless settings from startup configuration.

Result

The switch will no longer contain any wireless packages or settings.

Upgrading wireless packages

Summary

This process outlines how to upgrade the wireless software on a switch, ensuring the system runs the latest version.

Workflow

Perform these steps to upgrade wireless packages

1. Install the base image: Start by setting up the switch with the base software version.
2. Install the updated wireless package: Load the new wireless version onto the switch.
3. Reload: Restart the switch to apply changes.
4. Commit: Finalize the installation to activate the updated package.

Result

The switch successfully runs the upgraded wireless software, improving its capabilities and resolving any prior issues.

Boot in install mode

Boot in Install Mode using commands.

Switch the boot mode from bundle-mode to install-mode using commands.

Use this procedure to boot the switch in install-mode

Before you begin

The sub-package does not work in bundle-mode. Use the **show version** command to verify the boot mode.

Procedure

1. Move the switch from bundle-mode to install-mode.

```
install add file image.bin location activate commit
```

   Note that *image.bin* is the base image.

2. Click **yes** to all the prompts.
3. Reload the switch.

```
Device# reboot
```

   Boot from *flash:packages.conf*. After the reload, the switch will be in install-mode.

> **Note**
>
> For image upgrade or downgrade in Install mode, the **Install add file** command with flash: *<file_name>* command is not supported. Instead of that bootflash: *<filename>* command needs to be used.

```
Install add file bootflash:file_name  activate commit
```

After completing this task, the switch will be in install-mode.

What to do next

Verify the boot mode using the **show version** command.

Install sub-package in a single step (GUI)

Install sub-package in a single step using GUI.

The purpose of this task is to install a sub-package using the graphical user interface in a single step.

Procedure

1. Choose **Administration** > **Software Management** > **Software Upgrade**.
2. Choose the upgrade mode from the **Upgrade Mode** drop-down list, the transport type from the **Transport Type** drop-down list and enter the **Server IP Address (IPv4/IPv6)**, the **File System** and choose the location from the **Source File Path** drop-down list.
3. Click **Download & Install**.

The sub-package is successfully downloaded and installed, ensuring the software is updated with the latest version.

Install sub-package in a single step

Install sub-package in a single step using commands.

The purpose of this task is to efficiently install a sub-package onto the switch in a single step, streamlining the update or installation process.

Before you begin

- Ensure that the switch is in install mode.
- Ensure that you boot only from *flash:packages.conf*.

Use this procedure to install sub-package in a single step

Procedure

1. **install add file** *flash:<controller>.bin* **activate commit**

   Installs the Cisco Catalyst 9800 Wireless Controller for Switch sub-package.

   > **Note**
   >
   > The sub-package (flash:<controller>.bin) is available on www.cisco.com. You can also install the sub-package directly from TFTP server.

2. Click **yes** to all the prompts.

Upon completion, the sub-package will be successfully installed and activated on the switch, ready for its verification and use.

What to do next

Use the **show install summary** command to verify the installed image or package.

Install multi-step sub-package

Install multi-step sub-package using commands.

This task provides steps to install a multi-step sub-package on a switch.

Before you begin

Ensure that the switch is in install-mode and boot only from flash:packages.conf.

Procedure

1. Add sub-package to the flash and expand.

   install add file *flash:<controller>.bin*

2. Install the sub-package.

   install activate file *flash:<controller>.bin*

3. Complete the installation by writing the files.

   install commit

The sub-package installation is completed successfully, and the system is ready for operation.

What to do next

Use the **show install summary** command to verify the installed image or package.

Install the package on a stack

Install the package on a network stack using a structured method.

Use a structured method to install the package on a network stack.

Before you begin

- Ensure you have the correct permissions.
- Determine if auto-upgrade is enabled or disabled for any new members.

Procedure

1. Verify the current version running on the stack.
2. Choose the installation method:
   - Use Single-step Package Installation if applicable.
   - Refer to the multi-step installation guide as required.
3. If a new member joins, there are two possible scenarios.

if auto-upgrade is enabled	confirm the software automatically installs.
if auto-upgrade is disabled	run the **install autoupgrade** command in EXEC mode manually.

The package is installed, and new stack members are aligned with the current software version, maintaining consistency across the system.

Upgrade to new version of wireless package

Upgrade to new version of wireless package using commands.

Upgrade the wireless package to the latest version by executing specific commands.

Use this procedure to upgrade to the new version of wireless package

Procedure

1. Add the base image (without wireless) to the flash and expand it.

   install add file *flash:<base-image>.bin*

2. Add the sub-package to the flash and expand it.

   install add file *flash:<controller-sub-package>.bin*

3. The system executes the installation of the base image and sub-package, which causes a reload.

   install activate

   You can also rollback to the previous state after the reload.

4. Complete the installation by writing the files.

   install commit

The latest wireless package version runs on your system. You can rollback if necessary.

Deactivate wireless package

Deactivate the wireless package using commands.

You can deactivate the wireless package on a device to ensure it operates without wireless functions through this task.

Follow this procedure to deactivate the wireless sub-package:

Procedure

1. Remove the package and force the switch to reboot

   Example

   ```
   Device# install deactivate file flash:<controller>.bin
   ```

   Removes the package and forces the switch to reboot.

2. Commit the switch without wireless package

   Example

   ```
   Device# install commit
   ```

Commits the switch without wireless package.

The device reboots and ensures that only non-wireless functions are available by operating without the wireless package.

Enable or disable auto-upgrade

Enable or disable auto-upgrade using commands.

The purpose of this task is to manage the auto-upgrade capabilities of the software on the device, allowing you to either enable or disable automatic upgrades according to operational needs.

Use this procedure to enable or disable auto-upgrade

Procedure

---

1. Enable software auto-upgrade

   Example

   ```
   Device(config)# software auto-upgrade enable
   ```

   Enables software auto-upgrade.

2. Disable software auto-upgrade

   Example

   ```
   Device(config)# no software auto-upgrade enable
   ```

   Disables software auto-upgrade.

The device either automatically updates its software when new firmware is available or remains on its current software version without automatic upgrades, depending on the command executed.

Part III

Lightweight Access Points

Topics:

- Global Use Access Points
- Countries and Regulations
- Security
- AP Joining
- AP Crash File Upload
- LED States for Access Points
- AP Support Bundle
- Access Point Memory Information
- Access Point Tag Persistency
- AP Power Save
- Real-Time Access Points Statistics
- AP Configuration
- Secure Data Wipe
- Wi-Fi 7 Operations
- Troubleshooting Lightweight Access Points

21 Global Use Access Points

Topics:

- Global use access points
- AP mode of options
- Cisco regulatory domain
- E-Label display

About Global Use APs.

Global use access points

Introduces the concept of global use APs and highlights best practices, detailing their benefits and essential considerations for deployment and use in various regulatory environments.

Global use access points (APs) are network devices that

- operates in both Meraki cloud native networks and Catalyst on-premise controller-based networks
- uses a single product ID (PID) for simplified logistics, and
- decouple PID or SKU from regulatory domains and boot modes.

The transition to a unified PID model reduces SKUs from around 16 product ID (PIDs) to nine by introducing Rest of World (RoW) APs, which support country-specific configurations. In contrast, non-RoW APs remain restricted due to strict regional compliance policies.

Regulatory compliance and global use AP

The important aspects that the Global Use AP addresses for Catalyst and Meraki Cloud setups are the AP mode of operation and the adherence to Cisco Regulatory Domain requirements, ensuring compliance with regulatory standards on radio channel usage and power limits across different geographical bands.

Benefits of global use AP

Global use AP benefits.

Global use APs offer numerous advantages that enhance network management efficiency and flexibility. The benefits of global use APs are:

- One PID to order Cisco Catalyst or Meraki mode APs
- Seamless provisioning
- Easily transition from Catalyst mode to Meraki mode to meet network needs
- Reduce complexity with no reliance on country-specific SKUs

AP mode of options

Explains supported AP operating modes and workflows, covering Cloud ID, Day 0 onboarding, prioritized migration methods, fast offline migration steps, local status and offline migration details, and strategies to prevent false migration.

An AP mode option is a provisioning choice that

- allows the AP to operate in either Meraki Cloud mode or Catalyst controller mode
- can be set during the initial deployment or after deployment
- and enables the AP to join the appropriate management system.

Cloud IDs

A Cloud ID is an identifier that

- represents the Meraki Serial Number for a device
- appears on the device's physical label, packaging, QR code, and related materials, and

- is essential for the Meraki device claim workflow in networks using devices such as the Cisco Wireless 917x series AP.

Additional reference information

The Cloud ID enables the streamlined claiming and management of Meraki devices within cloud-managed workflows. From the Cisco Wireless 917x series AP onward, this identifier provides a consistent reference for inventory and setup operations.

- A network administrator uses the Cloud ID printed on the access point's label to claim the device in the Meraki dashboard.
- The QR code containing the Cloud ID simplifies device onboarding for IT staff.

A device serial number used in traditional on-premises workflows is not considered a Cloud ID unless it participates in the Meraki cloud claim process.

Onboarding process for Wi-Fi 7 APs

Describes the brief details of the Day 0 onboarding workflow.

This section describes how to connect Cisco Catalyst 917x Wi-Fi 7 APs (CW917x) to the Meraki Dashboard and the Cisco Catalyst 9800 Wireless LAN Controller.

Summary

This process involves connecting the APs to the Meraki Dashboard and migrating them to the Cisco Catalyst 9800 Wireless LAN Controller.

The key components involved in the process are

- Cisco Catalyst 917x Wi-Fi 7 APs (CW917x): The APs must be powered on and claimed on the Meraki dashboard.
- Meraki Dashboard: Cloud-based management platform for onboarding APs.
- Cisco Catalyst 9800 Wireless LAN Controller: Final migration destination for managing APs.

Workflow

These stages describe connecting the APs to the Meraki Dashboard and migrating to the Cisco Catalyst 9800 Wireless LAN Controller.

1. Stage 1: Connect to Meraki Dashboard
   - Power on APs.
   - Claim APs using order number, Cloud ID, or MAC address. You can see the claimed APs on the Meraki dashboard as they start up.

   > **Note**
   >
   > Ensure there are no DHCP, DNS, or PnP configurations that can lead the CW917x AP to the controller.

2. Stage 2: Migrate the APs to the Cisco Catalyst 9800 Wireless LAN Controller
   - **Option 1:** Cloud-assisted migration If internet is available and the Meraki Dashboard account is configured, allow APs to join the dashboard and then migrate to the controller.
   - **Option 2:** Offline or fast offline migration If no internet is available:

- Offline migration uses DHCP or DNS without changes to the existing network.
- Fast offline migration uses new DHCP or DNS settings and skips cloud verification.

- If fast offline migration is not used, the AP will
  - search for the Meraki cloud for eight minutes, and
  - after eight minutes, check DHCP or DNS for the controller and migrate via CAPWAP discovery.

Result

APs successfully transition from initial setup in the Meraki Dashboard to established operational management under the controller, ready for enhanced performance and control.

Priority order for AP migration methods

To ensure efficient and reliable AP migration, use the following methods in priority order. Select the highest-priority method available for your deployment.

The migration methods are prioritized in this order

- Fast Offline Migration
  - DHCPv4
  - DNSv4
  - DHCPv6
  - DNSv6
- Local Status Page
  - You can manually migrate the APs through the Local Status window of the AP when Fast Offline Migration options are not available.
- Offline Migration
  - DHCPv4
  - DHCPv6
  - DNSv4
  - DNSv6
  - Broadcast or Multicast Discovery

Fast offline migration

Follow these steps to perform a fast offline migration for DHCPv4, DHCPv6, DNSv4, and DNSv6.

This task guides you through the process of executing a fast offline migration using DHCPv4, DHCPv6, DNSv4, and DNSv6 configurations to ensure network stability and performance.

This procedure is essential for network environments requiring quick migration of APs to a new controller mode, leveraging DHCP and DNS configurations.

Procedure

1. DHCPv4: The fast offline migration for DHCPv4 is Option 43.

   The syntax for the option 43 string is `F3 <size> <IP array> Mode=<1|2>`, where Mode 1 is Meraki and Mode 2 is Catalyst.

   a) To change the existing option 43 string `f104.ac10.0118` to `f305.ac10.0118.02`, do the following:

      - Change type from `f1` to `f3`,
      - Change length from `04` to `05`,
      - Add the sub-option `01` for Meraki or `02` for Catalyst.

   b) For controller discovery using DHCPv4, enter the following IOS/IOS XE configuration example:

```
ip dhcp pool vlan192
                                    network 192.168.200.0 255.255.255.0
                                    default-router 192.168.200.1
                                    option 43 hex hex f305.ac10.0118.02
```

   c) In the Windows server configuration, choose **DHCP** > **IPv4** > **Scope Options**.
   d) From the list of option names, select **043 Vendor Specific Info** and click **Configure Options**. The **Scope Options** window is displayed.
   e) Choose the **General** tab and check the **043 Vendor Specific Info** checkbox from the **Available Options**.
   f) Click **OK**.

   Note

   Ensure that at least one IP from the IP array is reachable through Internet Control Message Protocol (ICMP) or CAPWAP. The AP tries to contact the controller through pings. If the controller responds, the AP migrates to the controller management mode, otherwise the AP tries a CAPWAP discovery.

2. DHCPv6: The fast offline migration for DHCPv6 involves configuring Option 52 and Option 17.

   Note

   Configure Option 17, where the Enterprise ID is 29671, Subcode is 1, Mode is either 1 (Meraki), or 2 (Catalyst vendor-specific 29671). Option 52 (standard) is IPv6 array.

   a) In the Windows server configuration, choose **DHCP** > **IPv6** > **Scope Options**.
   b) From the list of option names, select **00052 capwap ac-address** and click **Configure Options**. The **Scope Options** window is displayed.
   c) Choose the **General** tab and select the **00052 capwap ac-address** checkbox from the **Available Options**.
   d) Click **OK**.
   e) To configure Option 17, choose **DHCP** > right-click **IPv6**, and select **Define Vendor Classes**. The **DHCP Vendor Classes** window is displayed.
   f) In the **DHCP Vendor Classes** window, create a new class.

g) In the **New Class** window, enter the **Display Name**, enter the **Vendor ID** as `29671`, and enter the **ASCII** value as `Cisco Wireless AP CW9178I`, for the CW9178I platform.

> **Note**
>
> For CW9176I and CW9176D1, use the following ASCII strings, respectively, `Cisco Wireless AP CW9176I` and `Cisco Wireless AP CW9176D1`.

h) Click **OK**.
i) Choose **DHCP** > right-click **IPv6**, and select **Set Predefined Options**.
The **Predefined Options and Values for v6** window is displayed.
j) In the **Predefined Options and Values for v6** window, click **Option Type**.
The **Option Type** window is displayed.
k) In the **Option Type** window, in the **Code** field, enter `1` and then click **OK**.
l) Click **OK**.
m) Choose **DHCP** > **IPv6** > **Scope Options** > **Configure Options**.
The **Scope Options** window is displayed.
n) In the **Scope Options** window, click the **Advanced** tab.
o) From the **Vendor class** list, select the vendor class.
p) From the **Available Options**, check the **00001 fast-conversion** checkbox.
q) In the **Data entry** section, in the **Byte** field, enter `0x2`.
r) Click **OK**.

> **Note**
>
> Ensure at least one IP from the IP array is reachable via Internet Control Message Protocol (ICMP) or CAPWAP. The AP tries to reach the controller through pings. If the controller responds, the AP migrates to the controller management mode, otherwise the AP tries a CAPWAP discovery.

The **00001 fast-conversion** option is added to the list of **Scope Options**.

3. DNSv4 and DNSv6: Fast offline migration using the migration string.
   a) Add a DNS record **cisco-automigrate.<domain>** in the DNS server.

   The AP checks for the **cisco-automigrate.<domain>** DNS entry.

   If the IP pings from DNS are successful, then the APs migrate immediately to the controller mode. If the pings fail, the AP tries the CAPWAP discovery method.

   > **Note**
   >
   > For this method to work correctly, the DHCP server must return the domain-name option (option 15) to the AP.

Successfully configuring fast offline migration ensures that APs migrate quickly to the new controller mode, maintaining network stability and performance.

Local status page

Describes the local status page.

For more information on how to access the local status page, refer to *Using the Cisco Meraki Device Local Status Page*.

Offline migration

The AP uses the traditional discovery mechanisms of DHCP, DNS and broadcast or multicast, if it is not able to reach the Meraki Cloud, perform a Fast Offline Migration, or migrate using the Local Status page.

Prevent false or accidental migration

The purpose of this task is to guide users on preventing accidental or unauthorized migration of APs to a controller, ensuring network stability and security.

To prevent accidental or false migration, perform these steps:

Procedure

1. Ensure that at least one IP in the IP address array returned by the DHCP option or the IP address of the resolved DNS entry is CAPWAP reachable. This CAPWAP response must include the controller version image for the CW917x AP to validate it and join the controller.
2. Enter cisco-do-not-automigrate when setting the DNS entry to resolve the controller's IP address. When it is resolved, the AP does not migrate to the controller mode.
3. You can configure commands to prevent the controller from responding to Day 0 CAPWAP discovery requests by CW917x APs. You can fine tune its configuration based on its own deployment specification. Configuration is done in the AP Join Profile.

   To prevent accidental or false migration, you must perform these configurations.

```
Device(config)# ap profile onboarding-prof
Device(config-ap-profile)# no capwap-discovery onboarding
Device(config-ap-profile)# exit
```

   To respond to the CAPWAP discovery, use the following configuration:

```
Device(config)# ap profile onboarding-prof
Device(config-ap-profile)# capwap-discovery ?
onboarding  Configure CAPWAP onboarding related parameters
private     Include private IP in CAPWAP Discovery Response
public      Include public IP in CAPWAP Discovery Response

Device(config-ap-profile)# capwap-discovery onboarding ?
all      Configure automatic CAPWAP onboarding from Meraki based on both
unicast and broadcast discovery request
unicast  Configure automatic CAPWAP onboarding from Meraki based on unicast
discovery request only
```

   By default the controller accepts only unicast requests for onboarding.

> **Note**
>
> If the CW917x APs have to be in the same subnet as the controller and use Broadcast (IPv4), or Multicast (IPv6) for discovery, then CAPWAP discovery onboarding should be set to **all**. Otherwise, the controller does not respond to Broadcast or Multicast discovery requests.

Completion of this task ensures that the APs are properly configured and do not unintentionally migrate to controller mode, preventing unauthorized network changes and maintaining system integrity.

Cisco regulatory domain

Details regulatory domain management for APs, providing guidance on country code assignment, best practices, examples of country code resolution, and step-by-step instructions for configuring and verifying regulatory activation via GUI and CLI.

Determining the regulatory domain is very important for wireless AP deployments. Unlike the mode of operation, which is a choice that can be made by the customers, provisioning the regulatory domain follows some strict rules and guidelines. With single PID, regulatory domain is automatically determined from where the APs are deployed.

Country code and regulatory domain

A country code is an identifier that enables Wi-Fi 7 APs to operate in compliance with the regulations of the country where they are deployed. Regulatory domains ensure that APs adhere to local compliance requirements through methods such as

- Compliance through geo-location: For APs operating in Meraki mode, regulatory standards are enforced using geo-location and network-wide settings.
- Hardware-independent enforcement: With certain AP models, such as the CW917x series, compliance is achieved through software rather than hardware-specific SKUs allowing worldwide deployment.

Regulatory domain information is determined based on the location and country where APs are deployed.

Determining AP Location

Various methods exist for determining the location of APs, which affect regulatory domain information.

- **Meraki determined location:** APs use IP addresses, shipping destinations, or dashboard configurations to reveal their location, essential for regulatory enforcement.
- **GPS or GNSS-based location:** If an AP is unable to identify the country-of-presence through Cisco cloud, the AP makes use of GPS or GNSS-based location. When the AP is located close to the open sky and there are enough satellites to get a location fix, the AP tries to use the satellites to locate country of operation.
- **Proximity-based AP location:** The proximity-based AP location method is useful for determining AP location based on proximity to other APs, especially when the APs are unable to find their location using the standard location techniques.

  For example, Anchored APs are APs that are closer to the periphery of a building and may be able to receive GPS signals, whereas the Non-Anchored APs are APs located in the middle of building, which may not be able to receive signals. The controller learns about the neighboring APs using radio resource management (RRM). Using RRM, you can propagate the location from the Anchored APs to the Non-Anchored APs.

This method is effective when the Wi-Fi 7 APs hear the Neighbor Discovery Protocols (NDPs) over 2.4 GHz from any other AP (Wi-Fi 7 or other APs) joined to the same controller.

Proximity-based location determination is faster compared to the other approaches, such as GPS or GNSS-based location.

- **AP Regulatory Activation File**: When automated methods are unsuccessful, a manual configuration via the appropriate activation file is implemented.

Best practices for Cisco regulatory domain

- In some deployment scenarios, without a clear sky view or with thick windows and neighboring buildings, the AP may take a long time or fail to obtain a GPS/GNSS signal. In such cases, other methods like Proximity, Migration, or Regulatory Activation File can be used to obtain the country code and regulatory information for the controller management mode.

  For greenfield deployments, we recommend that you use GPS/GNSS. Use Migration through Meraki Dashboard if you encounter issues obtaining the GPS/GNSS signal.

- For Cisco Wireless 9176I APs, configure the country code in the AP join profile to match the country code assigned through the Regulatory Activation File, proximity-based discovery, or GPS/GNSS. If the AP join profile does not include the matching country code, AP radios may remain operationally down or operate with limited functionality.

Examples of country code resolution

Gives the examples of country code resolution.

This section aims to provide detailed examples and explanations of the methods used by Wi-Fi 7 APs to determine and resolve their country codes, including GPS or GNSS-based location, proximity-based location, migration, and regulatory activation file methods.

These examples show how Wi-Fi 7 APs determine their country code:

- GPS or GNSS-based location: APs obtain geolocation via the integrated GPS or GNSS antenna. The CW917x series AP have a built-in GPS or GNSS. They can obtain the geolocation co-ordinates, as long as they can get a clear sky view, typically placed near a window, to obtain the satellite signal. The controller maps geolocation coordinates to the country. Until the AP obtains the country code, it will have **-UN** as the regulatory domain.

  For APs joining without domain the output displays **-UN** as the regulatory domain.

  For example, this **show** command displays the details of a Wi-Fi 7 AP which has the regulatory domain as -UN:

```
Device# show ap summary
Number of APs: 1
CC = Country Code
RD = Regulatory Domain

AP Name Slots AP Model  Ethernet MAC   Radio MAC      CC   RD   IP Address
   State         Location
-------------------------------------------------------------------------------
AP-B2E0   4   CW9178I   c414.a26f.b2e0 c414.a26f.b2f0 --   -UN  20.20.20.52
  Registered default location
```

  This **show** commands display the results after the controller determines the country code and the AP reloads and joins back with a country code:

```
Device# show ap name Wi-Fi7-AP-B2E0 config general | inc Country
Country Code                                    : US
Regulatory Domain Allowed by Country            : 802.11bg:-A 802.11a:-AB
```

```
802.11 6GHz:-B
AP Country Code                                 : US  - United States
Country Code Resolution Method                  : GPS


Device# show ap summary
Number of APs: 1
CC = Country Code
RD = Regulatory Domain
AP Name Slots AP Model  Ethernet MAC   Radio MAC      CC   RD   IP Address
   State         Location
-------------------------------------------------------------------------------
AP-B2E0   4   CW9178I   c414.a26f.b2e0 c414.a26f.b2f0 US   -B  20.20.20.52
 Registered default location
```

- Proximity-based AP location: For brownfield deployments, if your APs cannot obtain the GPS/GNSS signal due to existing legacy APs, use proximity-based discovery.

```
Device# show ap summary
Number of APs: 2
CC = Country Code
RD = Regulatory Domain
AP Name  Slots AP Model  Ethernet MAC  Radio MAC       CC   RD   IP Address
     State        Location
-------------------------------------------------------------------------------
AP-B2E0    4     CW9178I c414.a26f.b2e0 c414.a26f.b2f0 --   -UN  20.20.20.52
  Registered   default location
AP-E040    3     CW9176I 8c88.814f.e040 ecf4.0caf.6a60 US   -B   20.20.20.51
  Registered   default location
```

In the example above, the AP named AP-E040 has a country code and regulatory domain and the AP named AP-B2E0 that does not have a country code and regulatory domain, are in close proximity. AP-B2E0 learns its country code from AP-E040. The APs reload, after they learn their country code from the neighboring APs.

This example displays the state of the AP after obtaining the country code:

```
Device# show ap summary
Number of APs: 2
CC = Country Code
RD = Regulatory Domain
AP Name  Slots AP Model  Ethernet MAC  Radio MAC       CC   RD   IP Address
     State        Location
-------------------------------------------------------------------------------
AP-B2E0    4     CW9178I c414.a26f.b2e0 c414.a26f.b2f0 US  -B  20.20.20.52
 Registered   default location
AP-E040    3     CW9176I 8c88.814f.e040 ecf4.0caf.6a60 US  -B  20.20.20.51
 Registered   default location
```

```
C9800-L# show ap name Wi-Fi7-AP-B2E0 config general | inc Country
Country Code                                    : US
Regulatory Domain Allowed by Country            : 802.11bg:-A 802.11a:-AB
 802.11 6GHz:-B
AP Country Code                                 : US  - United States
Country Code Resolution Method                  : Proximity
```

- Migration: When you migrate APs from Meraki Dashboard, the APs keep their existing country code.

  For greenfield users, where

  - there are no legacy deployments,

- the deployment restrictions make it difficult to obtain the GPS or GNSS signal, and
- there is easy workflow of migrating the APs to the controller mode for Day 0, and obtaining country information,

we recommend the migration method.

This example displays the country code resolution method as Installed through Meraki Dashboard

```
Device# show ap name Wi-Fi7-AP-B2E0 config general | inc Country
Country Code                                    : US
Regulatory Domain Allowed by Country            : 802.11bg:-A  802.11a:-AB
802.11 6GHz:-B
AP Country Code                                 : US  - United States
Country Code Resolution Method                  : Installed via Meraki
Dashboard
```

- Regulatory Activation File: Download a Regulatory Activation File (RAF) from the Meraki Dashboard, that can be installed on the controller. See the section on Regulatory Activation File method for more details.

Regulatory activation file

The Cisco Meraki dashboard creates a regulatory activation file that has AP details such as the AP location, serial numbers, MAC addresses, country code, and site tags that can be imported to the controller through the Cisco Meraki Dashboard or saved to an external server or device.

To download the file from the Meraki Dashboard, choose **Meraki Dashboard** > **Network-wide** > **General** > **Regulatory info** > **Download regulatory file**.

> **Note**
>
> When you apply a new Regulatory Activation File (RAF), the new AP MAC-to-country mapping replaces the existing mapping. The contents of the new RAF overwrite the contents of the previously applied RAF; the mappings are not merged. Ensure that the new RAF contains all required AP mappings before applying it.

Configure AP regulatory activation (GUI)

Complete this task to activate the country code and regulatory domain of APs, which are in the worldwide mode.

This task guides you through the process of activating the regulatory domain on APs that have not been factory-programmed with a specific regulatory domain, using the GUI.

Complete this task to activate the country code and regulatory domain of APs, which are in the worldwide mode.

Before you begin

To provision a specific regulatory domain on APs that are not factory-programmed with a regulatory domain, download the regulatory activation file from the Meraki dashboard.

Procedure

1. Choose **Administration** > **Regulatory Activation**.
2. Click the upload area to select the regulatory activation file from your local folder, or, drag and drop the file into the upload area.

> **Note**
>
> Ensure that you add a new file every time regulatory activation has to be performed.

3. Click **Upload** to upload and validate the regulatory activation file.

   A message indicates the activation file has been validated. The records display the **AP MAC** details, **Country Code**, **Serial Number**, and the following validation results:

   - **Invalid Domain**: In this mode, APs do not have a country code, resulting in an undefined (UN) regulatory domain. The country code from the activation file is applied to APs that match this condition.
   - **Regulatory Mismatch**: In this mode, APs have a country code, but the country code is not configured on the controller.

     > **Note**
     >
     > To resolve a regulatory mismatch issue, enable the country code on the controller.

   - **Not World Wide Mode**: The APs in this mode are currently not affected by the regulatory activation process.

4. Click **Apply** to start the activation process.

   The activation process resolves the country code and regulatory domain of APs in Invalid Domain.

   A warning message is displayed that the regulatory activation process will cause the joined APs to be non-operational and hence get disconnected.

5. Click **OK** to start the process.

   The process runs in the background. A message is displayed if none of the records is impacted by the uploaded regulatory activation file.

The result of this task is that the APs are configured with the appropriate country code and regulatory domain, ensuring compliance and operational readiness.

What to do next

Go to the **Access Point Configuration** page to monitor the results.

Configure AP regulatory domain (CLI)

Describes the CLI steps to configure AP regulatory activation.

The purpose of this task is to guide users through the CLI commands necessary to configure AP regulatory activation, ensuring compliance with regional regulations.

Procedure

---

1. Enable privileged EXEC mode

Example

```
Device# enable
```

Enables privileged EXEC mode. Enter your password if prompted.

2. Apply the regulatory domain configuration

Example

```
Device# ap regulatory activation apply
```

Applies the regulatory domain configuration.

3. Clear all the AP MAC to country mapping records

Example

```
Device# ap regulatory activation clear all
```

Clears all the AP MAC to country mapping records or a specific AP MAC to country mapping record.

4. Configure the regulatory domain activation file

Example

```
Device# ap regulatory activation file reg-ap-act.json
```

Configures the regulatory domain activation file.

This configures the activation file necessary for domain settings.

What to do next

Verify that the AP regulatory domain configurations are operational and comply with regional requirements.

Verifying regulatory activation

This document provides verification details for the regulatory activation feature.

Describes the verification details of the regulatory activation feature.

Verifying Regulatory Activation Details for All APs

To verify regulatory activation details for all APs, run the following command:

```
Device# show ap regulatory activation all
Regulatory Activation file Meta-data
------------------------------------
Date Created     : 2024-06-28T08:55:13Z
Created By       : ciscoap.user@meraki.net/SSO
Device count     : 10
Organization Id  : 1130091
AP MAC              Serial Number      Country code
-------------------------------------------------
00XX.0a0X.5dXX       Q2XX-LUXX-DDXX       US
00XX.0a0X.5fXX       Q2XX-MUXX-4RXX       US
```

```
00XX.0a0X.01XX       Q2XX-ALXX-4UXX       US
00XX.0a0X.63XX       Q2XX-G8XX-7WXX       US
00XX.0a1X.0dXX       Q2XX-BBXX-ELXX       US
00XX.0a1X.19XX       Q2XX-E8XX-ZZXX       US
00XX.0a1X.1dXX       Q2XX-2LXX-FSXX       US
00XX.0a1X.04XX       Q2XX-5RXX-AFXX
00XX.0a3X.08XX       Q2XX-3YXX-D3XX       US
00XX.0a1X.0aXX       Q2XX-5QXX-ERXX       US
.
.
.
```

Verifying Regulatory Activation Details for Specific APs

To verify the regulatory activation details for specific APs, run the following command:

```
Device# show ap regulatory activation 38XX.79XX.9fXX
AP MAC          Serial Number         Country Code
-----------------------------------------------------
38XX.79XX.9fXX  Q3XX-XKXX-F3XX
```

E-Label display

Describes E-Label display features for APs, including compliance status monitoring for device models and configuration procedures using the GUI to ensure adherence to electronic labeling requirements.

E-LABEL compliance status

This topic describes the e-label compliance status.

A regulatory requirement for wireless APs is a set of standards that:

- mandates compliance with evolving legal or formal regulations,
- covers aspects such as operational frequencies and power levels, and
- include the implementation of technologies like the "E-label" for compliance.

Starting with the Cisco IOS XE 17.18.1 version, the E-Label digital labeling feature became accessible via the controller's GUI. This feature enables seamless integration of new regulatory approvals concerning the E-Label compliance without necessitating physical modifications to the devices.

Feature history for access point upgrade management enhancements
Table 25: Feature history

Release	Feature Information
Cisco IOS XE 17.18.1	The E-Label digital labeling feature is accessible via the controller's GUI and enables seamless integration of new regulatory approvals for wireless APs.

Starting with the Cisco IOS XE 17.18.1 release, this digital labelling feature is available via the controller's GUI, enabling seamless integration of new regulatory approvals without requiring physical modifications to the device.

E-Label compliance status of AP models

E-Label compliance status of AP models.

These are the AP models that support E-Label compliance status

- Cisco Catalyst Wi-Fi 6 APs
- Cisco Catalyst Wi-Fi 6E APs
- Cisco Catalyst Wi-Fi 7 APs

Wi-Fi Technology	AP Model
Cisco Catalyst Wi-Fi 6 APs	Cisco Catalyst 9105 AXI/AXW APs
	Cisco Catalyst 9115 AXI/AXE APs
	Cisco Catalyst 9117 AXI APs
	Cisco Catalyst 9120 AXI/AXE/AXP APs
	Cisco Catalyst 9130 AXI/AXE/AXE-STA APs
	Cisco Catalyst 9124 AXI/AXE/AXD APs
Cisco Catalyst Wi-Fi 6E APs	Cisco Catalyst 9136I APs
	Cisco Catalyst Wireless 9166I/D1 APs
	Cisco Catalyst Wireless 9164I APs
	Cisco Catalyst Wireless 9162I APs
	Cisco Catalyst Wireless 9163E APs
Cisco Catalyst Wi-Fi 7 APs	Cisco Catalyst Wireless 9178I APs
	Cisco Catalyst Wireless 9176I/D1 APs
	Cisco Catalyst Wireless 9172I/H APs

Configure AP E-Label compliance status (GUI)

This topic describes how to configure an AP E-Label compliance status using the GUI.

Ensure regulatory compliance of the AP E-Label by configuring its status through the GUI.

This task involves verifying the compliance status of an AP E-Label via the GUI.

To configure the AP E-Label compliance status, perform these steps

Procedure

1. Choose **Monitoring** > **Wireless** > **AP Statistics**.

> **Note**
>
> All the APs connected to the controller are automatically displayed.

2. Click the AP to view the regulatory compliance status.
3. Go to **360 View**.
4. Click the **Click here for Legal and Regulatory information**.

> **Note**
>
> This option will be under the **AP Country Code** detail.

A pop-up is displayed with the regulatory information for the selected AP, showing compliance details across various countries.

22 Countries and Regulations

Topics:

- Country codes
- Regulatory compliance domain
- Remove 8 channel restriction on slot 1 of Cisco Catalyst 9166 series APs
- Ensure 6 GHz radio band persistence after CAPWAP reset

Country codes

Explains how country codes enable wireless devices to comply with regulatory requirements specific to each country.

A country code is a regulatory identifier that

- specifies the country of operation for wireless devices, such as controllers and APs which must comply with varying regulatory requirements, and
- ensures each radio's broadcast frequency bands, interfaces, channels, and transmit power levels comply with country-specific regulations.

Each AP radio receives a regulatory domain at the factory, such as -E for Europe. Set the country code to match the location—for example, FR for France or ES for Spain.

Information About Japanese Country Codes

Country codes define the legal channels for each country. These country codes are available for Japan:

- J2: which allows only -P radios to join the controller, and
- J4: which allows 2.4GHz JPQU and 5GHz PQU to join the controller.

Prerequisites for configuring country codes

Prerequisites for configuring country codes.

- Generally, you should configure one country code per device. Select a country code that matches where the device and its APs are physically located. You can configure up to 200 country codes per device. You can manage APs in different countries from one device using multiple-country support.
- If you use the multiple-country feature, configure all devices joining the same RF group with the same countries in the same order.
- Each AP uses all available legal frequencies, but it is assigned frequencies approved for its domain.
- The RF group leader's country list decides which channels are available for group members. The RF group leader's country list does not depend on the countries on group members.
- If your device is in the Japan regulatory domain, configure at least one Japan country code (JP, J2, or J3) before the most recent boot.
- For devices in the Japan regulatory domain, you should have one or more Japan country codes (J2, or J4) configured on your device at the time of the most recent boot.
- For devices in the Japan regulatory domain, you must have at least one access point with a -J regulatory domain joined to your device.
- You cannot delete a country code with the **wireless country country-code** command if you configured it with the **ap country list** command. You also cannot delete a country code using the **ap country list** command if you configured it with the **wireless country country-code** command.

Configure country codes (GUI)

Configure country codes using GUI.

Set the allowed country codes on wireless APs to comply with local regulations.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points** > **Country**.
2. On the **Country** page, check the check box for each country where your APs are installed. If you select more than one check box, a message displays that RRM channels and power levels are limited to common channels and power levels.
3. Click **Apply**.

Configure country codes (CLI)

Configure country codes using commands.

Assign and verify country codes on an AP or wireless controller to ensure regulatory compliance for wireless operation.

Procedure

1. Display the list of all the available country codes.

   Example

   ```
   Device# show wireless country supported
   ```

2. Enter the global configuration mode and disable the 802.11b/g network, if you use 24GHz, 5GHz or 6GHz.

   Example

   ```
   Device# configure terminal
   ```

   ```
   Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} shutdown
   ```

3. Assign APs to a specific country.

   Example

   ```
   Device(config)# ap country country_code IN
   ```

   Configure the country code on the controller so that APs joining the controller match the country code and the corresponding regulatory domain codes for the AP.

   Note

   You can configure more than one country code.

4. Configure 200 country codes per device.

   Example

   ```
   Device(config)# wireless country country_code IN
   ```

> **Note**
>
> Use this command for deployments with more than 20 countries.

5. Return to the privileged EXEC mode and display the configured countries.

   Example

```
Device(config)# exit
```

```
Device# show wireless country configured
```

6. Display the list of available channels for the country codes configured on your device and enter the global configuration device.

   Example

```
Device# show wireless country channels
```

> **Note**
>
> Complete the next three steps only if you have configured multiple country codes in step 6.

7. Enter the global configuration mode and enable the 802.11b/g network, if you use 24GHz, 5GHz or 6GHz. Return to the privileged EXEC mode and disable the AP.

   Example

```
Device# configure terminal
```

```
Device(config)# no ap dot11 {24ghz | 5ghz| 6ghz} shutdown
```

```
Device(config)# end
```

```
Device# ap name cisco-ap AP02 shutdown
```

> **Note**
>
> Make sure to disable only the AP that you are configuring with country codes.

8. Enter the global configuration mode and assign each AP with a country code from the controller country code list.

Example

```
Device# configure terminal
```

```
Device# ap name cisco-ap AP02 country country_code US
```

> **Note**
>
> - Ensure that the country code that you choose is compatible with the regulatory domain of at least one of the APs radios.
> - Disable the AP before changing its country code.

9. Return to privileged EXEC mode and enable the AP.

Example

```
Device(config)# end
```

```
Device# ap name cisco-ap AP02 no shutdown
```

Configuration examples for configuring country codes

Configuration examples for configuring country codes.

View channel list for country codes

View channel list for country codes details.

This example shows how to display the list of available channels for the country codes on your devicedevice:

```
DeviceDevice# show wireless country channels

Configured Country........................: US - United States
KEY: * = Channel is legal in this country and may be configured manually.
     A = Channel is the Auto-RF default in this country.
     . = Channel is not legal in this country.
     C = Channel has been configured for use by Auto-RF.
     x = Channel is available to be configured for use by Auto-RF.
   (-,-) = (indoor, outdoor) regulatory domain allowed by this country.
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-
802.11bg         :
Channels         :                  1 1 1 1 1
                  1 2 3 4 5 6 7 8 9 0 1 2 3 4
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-
(-A ,-AB ) US    : A * * * * A * * * * A . . .
Auto-RF          : . . . . . . . . . . . . . .
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
802.11a          :                        1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
Channels         :3 3 3 4 4 4 4 4 5 5 6 6 0 0 0 1 1 2 2 2 3 3 4 4 5 5 6 6
                  4 6 8 0 2 4 6 8 2 6 0 4 0 4 8 2 6 0 4 8 2 6 0 9 3 7 1 5
```

```
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
(-A ,-AB ) US    : . A . A . A . A A A A A * * * * * . . . * * * A A A A*
Auto-RF          : . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
4.9GHz 802.11a   :
Channels         :                   1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2
                  1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6
-----------------:+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-
US (-A ,-AB )    : * * * * * * * * * * * * * * * * * * * * A * * * * * A
Auto-RF          : . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
```

Regulatory compliance domain

Explains how regulatory compliance domains determine country-specific wireless operating parameters and streamline AP configuration.

A regulatory domain is a set of country-specific rules that

- are designed for use in multiple countries with varying regulatory requirements
- enables you to specify a particular country of operation, such as FR for France or ES for Spain, and
- ensures that each radio's broadcast frequency bands, interfaces, channels, and transmit power levels comply with country-specific regulations.

This feature reduces the number of regulatory domains by updating the pre-provisioned domains workflow. It determines the regulatory domain at runtime for each country code. The new Rest of World (RoW) domain includes the nine earlier domains. Each AP selects its regulatory domain from these options by referencing the regulated power table and allowed radio channels.

The transmission power value in the TPC IE of the beacon may differ from the AP's transmission power value by up to 2 dB. You can check this difference by using the **show controllers dot11radio** command. The maximum allowed deviation in the TPC IE of the beacon is two dB.

Global country-level domains

This topic lists detailed information about the power table and supported channels of countries in the global domain for 2.4GHz, 5GHz, 6GHz, and Rest of World domains.

- Use this information to determine the allowed configurations for each region.
- For additional technical details and Rest of World/6GHz support, see the Cisco Catalyst 9100AX Access Points Technical Reference.
- For more information about country phase and codes, refer to Cisco CW917x Wi-Fi 7 Access Point Country Codes and Phases.

Remove 8 channel restriction on slot 1 of Cisco Catalyst 9166 series APs

Explains channel restrictions on the XOR radio in Cisco Catalyst 9166 Series APs and how dynamic switching impacts channel availability based on region and device mode.

A channel restriction on an XOR radio is a regulatory and configuration constraint that

- limits the number of usable channels on the 5GHz or 6GHz bands based on region and device mode
- dynamically adjusts channel availability and power settings according to band selection and regional compliance, and

- enables optimized performance and access to all channels in supported configurations.

Starting with Cisco Catalyst IOS XE 17.12.5 release, you can restrict channels on slot 1 of Cisco Catalyst 9166 Series APs.

The Cisco 9166 Series AP includes dynamic XOR 5GHz or 6GHz radio band switching that optimizes performance and ensuring regulatory compliance. During startup, the XOR radio defaults to the 5GHz band in regions where 6GHz is not supported. Otherwise, it defaults to 6GHz. The AP actively adjusts and communicates channel power settings. It offers all full 5GHz channels when configured for 6GHz and restricted channels when in 5GHz mode.

The XOR radio infrastructure manages channel power payload adjustments when users switch between the 5GHz and 6GHz bands. The XOR radio can function as either a single 5GHz or a dual 5GHz setup. When operating in a single 5GHz mode, the AP provides a complete list of 5GHz channels available in the selected country. In dual 5GHz mode, the system configures and communicates channel power settings. It offers full 5GHz channels in one slot and restricted channels in the other.

Note

If 6GHz is not supported in your country, use slot 2 in either 5GHz or 6GHz. Configure the AP to Client serving mode to prevent resets.

For 6G unsupported countries, you must use slot 1 as a full band and configure slot 2 to 6GHz in Client serving mode.

Configure XOR 5/6GHz radio slot for 6GHz band (CLI)

Assign the 6GHz band to the XOR 5/6GHz radio slot on an access point to support Wi-Fi 6E clients.

Procedure

1. Enter the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Disable dual-band radio hosted on slot 2 for a specific AP.

   Example

   ```
   Device# ap name ap-name AP-SIDD-A06 dot11 dual-band slot 2 shutdown
   ```

   ```
   Device# ap name ap-name AP-SIDD-A06 [no] dot11 dual-band slot 2 shutdown
   ```

   Use the **no** form of this command to enable the dual-band radio.

3. Configure the dual-band role in client-serving mode.

   Example

   ```
   Device# ap name ap-name AP-SIDD-A06 dot11 dual-band slot 2 radio role manual
    client-serving
   ```

4. Configure the 802.11 dual-band radio's current band to 6Ghz.

   Example

```
Device# ap name ap-name AP-SIDD-A06 dot11 dual-band slot 2 band 6ghz
```

Ensure 6 GHz radio band persistence after CAPWAP reset

Configure the AP radio role to ensure the 6 GHz band remains active after a CAPWAP reset.

This procedure describes how to configure the AP radio role to manual for slot 2. This change ensures that the 6 GHz radio band remains active, even after a CAPWAP reset. If the radio role is set to auto, the AP might revert to the 5 GHz band upon reset.

Before you begin

Ensure you have CLI access to the AP.

Procedure

1. Verify the current radio role configuration. From the AP CLI, check the current radio configuration for slot 2.

   Example

```
show capwap client config | i Radio
```

   If the output shows the radio role for slot 2 as auto, proceed to Step 2.

   If the output shows the radio role for slot 2 as manual, no further action is required for this setting

2. Configure the radio role to manual and set the 6 GHz band. Execute these commands on the AP CLI to set the radio role for slot 2 to manual and configure it for the 6 GHz band. This step ensures the controller applies the manual radio role to the AP.

   Example

```
ap name NAME dot11 dual-band slot 2 shutdown
ap name NAME dot11 dual-band slot 2 radio role manual client-serving
ap name NAME dot11 dual-band slot 2 band 5ghz
ap name NAME dot11 no dual-band slot 2 shutdown
ap name NAME dot11 dual-band slot 2 shutdown
ap name NAME dot11 dual-band slot 2 band 6ghz
```

23 Security

Topics:

- Data Datagram Transport Layer Security
- 802.1X authentication
- Access point client ACL counters
- AP MAC authorization

Data Datagram Transport Layer Security

Introduces 802.1X authentication, outlining its security functions, supported EAP methods, integration with Cisco devices, feature history, and requirements for AP credential provisioning and authentication processes on Wave 2 and Wi-Fi 6 APs.

A data Datagram Transport Layer Security (DTLS) is a security protocol that

- encrypts CAPWAP data packets sent between an access point and a controller
- uses separate UDP ports for control (5246) and data (5247) packets,
- is a standards-track IETF protocol that can encrypt both control and data packets based on TLS, and
- supports v1.2 as the latest version available.

Feature history for data DTLS

Feature name	Release information	Feature description
Data Datagram Transport Layer Security	Cisco IOS XE Gibraltar 16.7.1	The data Datagram Transport Layer Security (DTLS) is a standards-track IETF protocol that can encrypt both control and data packets based on TLS.

CAPWAP control and data packets

CAPWAP control packets are management packets that are exchanged between a controller and an AP. CAPWAP data packets encapsulate forwarded wireless frames.

If an AP does not support DTLS data encryption, DTLS is enabled only for the control plane, and a DTLS session for the data plane is not established.

DTLS handshake

If an AP supports Data DTLS

- It enables data DTLS after receiving the new configuration from the controller
- The AP performs a DTLS handshake on port 5247 and after successfully establishing the DTLS session, and
- All the data traffic (from the access point to the controller and the controller to the access point) is encrypted.

Note

The throughput is affected for some APs that have data encryption enabled.

The controller does not perform a DTLS handshake immediately after processing client-hello with a cookie, if the following incorrect settings are configured:

- ECDHE-ECDSA cipher in **ap dtls-cipher** and RSA-based certificate in “wireless management trustpoint”.
- RSA cipher in **ap dtls-cipher** and EC-based certificate in “wireless management trustpoint”.

This is applicable when you move from CC > FIPS > non-FIPS mode.

> **Note**
>
> If the DHCP lease time of the AP is less and the DHCP pool is small, the AP join may fail or a failure in establishing the Data Datagram Transport Layer Security (DTLS) session may occur. In such scenarios, associate the AP with a named site-tag and increase the DHCP lease time for at least eight days.

Configure data DTLS (GUI)

Complete this task to enable DTLS data encryption for the APs on the controller.

Procedure

1. Click **Configuration** > **Tags and Profile** > **AP Join**.
2. Click **Add** to create a new **AP Join Profile** or click an existing profile to edit it.
3. Click **CAPWAP** > **Advanced**.
4. Check **Enable Data Encryption** check box to enable Datagram Transport Layer Security (DTLS) data encryption.
5. Click **Update & Apply to Device**.

The DTLS data encryption for the APs on the controller is enabled.

Configure data DTLS (CLI)

Configure Data DTLS using commands.

Complete this task to enable DTLS data encryption for the access points on the controller.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile-name
   ```

   > **Note**
   >
   > Use the default AP profile (default-ap-profile) or create a named AP profile, as shown in the example.

3. Enable link encryption on your profile.

Example

```
Device(config-ap-profile)# link-encryption
```

Answer **Yes**, when the system prompts you with this message:

```
Enabling link-encryption will reboot the APs with link-encryption.
                         Are you sure you want to continue? (y/n)[y]:
```

> **Note**
>
> If you set stats-timer as zero (0) under the AP profile, then the AP will not send the link encryption statistics.

4. Return to privileged EXEC mode.**end**

Example

```
Device(config-ap-profile)# end
```

5. Optional: Display the DTLS session established for the AP that has joined this controller.

Example

```
Device# show wireless dtls connections
```

6. Optional: Display the link encryption-related statistics (whether link encryption is enabled or disabled) counter received from the AP.

Example

```
Device# show ap link-encryption
```

The DTLS data encryption for the access points on the controller is now enabled.

```
Device# configure terminal
Device(config)# ap profile ap-profile-name
Device(config-ap-profile)# link-encryption
Device(config-ap-profile)# end
Device# show wireless dtls connections
Device# show ap link-encryption
```

802.1X authentication

Introduces 802.1X authentication, outlining its security functions, supported EAP methods, integration with Cisco devices, feature history, and requirements for AP credential provisioning and authentication processes on Wave 2 and Wi-Fi 6 APs.

IEEE 802.1X port-based authentication is a network security protocol that

- prevents unauthorized devices from accessing the network
- utilizes EAP authentication models to ensure secure communication, and
- integrates with devices like routers, switches, and access points based on configuration.

Feature history

Table 26: Feature history for 802.1X authentication

Feature name	Release information	Feature description
802.1X authentication	Cisco IOS XE 16.9.1	IEEE 802.1X port-based authentication is a network security protocol that utilizes EAP authentication models to ensure secure communication, and integrates with devices like routers, switches, and access points based on configuration.
Access ports with dual port authentication	Cisco IOS XE 17.17.1	The access ports with dual port authentication feature supports dual Ethernet ports on Cisco Catalyst 9136 APs and Cisco Wireless 9178I APs.

Currently, Cisco Wave 2 and Wi-Fi 6 (802.11AX) APs support 802.1X authentication with switch ports for EAP-FAST, EAP-TLS, and EAP-PEAP methods. Configuration and credential provision to APs can be done through the controller.

Note

If the AP is dot1x EAP-FAST, upon reboot, it should perform an anonymous PAC provision using ADH cipher suites to establish an authenticated tunnel. Authentication will fail if RADIUS servers do not support ADH cipher suites.

EAP-FAST protocol

Details of the EAP-FAST protocol.

In the EAP-FAST protocol developed by Cisco, to establish a secured TLS tunnel with RADIUS, the AP requires a strong shared key (PAC), provided via in-band or manual out-band provisioning.

- The EAP-FAST type configuration requires 802.1x credentials configuration for AP, since AP will use EAP-FAST with MSCHAP Version 2 method.
- Cisco 7925 phones do not support Local EAP.
- In Cisco Wave 2 APs, for 802.1x authentication using EAP-FAST after PAC provisioning (caused by the initial connection or after AP reload), ensure that you configure the switch port to trigger re-authentication using **authentication timer restart** *num* or **authentication timer reauthenticate** *num*.
- Starting from Cisco IOS XE Amsterdam 17.1.1, TLS 1.2 is supported in the EAP-FAST authentication protocol that requires strong security measures.

EAP-TLS/EAP-PEAP Protocol

Details about

The EAP-TLS protocol or EAP-PEAP protocol provides certificate based mutual EAP authentication.

In EAP-TLS, both the server and the client side certificates are required, where the secured shared key is derived for the particular session to encrypt or decrypt data. In EAP-PEAP, only the server-side certificate is required, and the client authenticates using a password-based protocol in a secured channel.

The EAP-PEAP type configuration requires Dot1x credentials configuration for the AP, and the AP also needs to go through LSC provisioning. AP uses the PEAP protocol with MSCHAP Version 2 method.

Access ports with dual port authentication

This section introduces the access ports that have dual port authentication.

The access ports with dual port authentication feature supports dual Ethernet ports on Cisco Catalyst 9136 APs and Cisco Wireless 9178I APs.

The access ports with dual port authentication feature

- Enhances network security and provides redundancy through 802.1x and MAC Authentication Bypass (MAB).
- Ensures high-availability and minimizes downtime.
- Supports authentication methods including EAP-FAST, EAP-PEAP, and EAP-TLS.
- Handles certificate distribution for secure connections on both ports.

Prerequisites for access ports with dual authentication

This topic describes the prerequisites for access ports with dual authentication.

Prerequisites for access ports with dual authentication support include:

- Enable dot1x on the switch port whenever dot1x credentials are configured on the controller. Ensure that dot1x credentials are configured on the controller and also on the switch port.
- Clear the user credentials to propagate the correct configuration on the AP,when dot1x is disabled.
- Modify the configuration on the controller, and then on the switch port, to avoid configuration mismatch on the AP when dot1x is enabled or disabled. Use the AP dot1x command for recovery.
- For APs with dual Ethernet ports, configure the connected switch port to allow authentication and authorization for both AP Ethernet MAC addresses. You can use one of these methods based on your site security policy:
  - Configure multi-host mode on the switch port.
  - Configure multi-auth mode and use MAC Authentication Bypass (MAB) for the AP MAC address on the Ethernet port that does not perform 802.1X authentication.
  - Configure multi-auth mode and use RADIUS authorization to apply a template that changes the switch port to multi-host mode after the AP successfully completes 802.1X authentication. The template can be pushed only for AP models that use dual Ethernet.
  - Starting with AP software Release 26.2.1, configure **access-session host-mode multi-host peer** on both switch ports connected to a dual Ethernet AP. The **peer** keyword is required for successful AP authentication. Update the switch-port configuration or the RADIUS authorization template before upgrading the AP.

802.1X authentication limitations

802.1X authentication limitations.

The 802.1X authentication limitations are:

- An AP loses its 802.1X credentials and configuration when migrating from a Cisco AireOS controller to a Cisco Catalyst 9800 controller. To rest ore them, the AP must join the new Catalyst 9800 controller. The controller pushes the necessary credentials and configuration for authentication on the switchport. You can either temporarily disable 802.1X authentication on the switchport to allow the AP to connect and receive its configuration, or use MAC Authentication Bypass (MAB) to provide network access to the AP for staging. After staging, you need to reload the AP or restart 802.1X authentication on the switch to complete the setup.
- 802.1X is not supported on dynamic ports or Ethernet Channel ports.
- 802.1X is not supported in a mesh AP scenario.
- There is no recovery from the controller on credential mismatch or the expiry/invalidity of the certificate on AP. The 802.1X authentication has to be disabled on the switch port to connect the AP back to fix the configurations.
- There are no certificate revocation checks implemented on the certificates installed in AP.
- You can provision only one Locally Significant Certificate (LSC) for the AP. Use the same certificate for both CAPWAP DTLS session establishment and 802.1X authentication with the switch. If you disable the global LSC configuration on the controller, the AP deletes the already provisioned LSC.
- If the AP has clear configurations applied, it will lose the 802.1X EAP Type configuration and the LSC Certificates. The AP should undergo the staging process again if 802.1X is required.
- 802.1X for trunk port APs on multi-host authentication mode is supported. Network Edge Authentication Topology (NEAT) is not supported on COS APs.
- The DHCP requests are sent in incremental periodic value of: "2, 3, 4, 6, 8, 11, 15, 20, 27, 30, 30, 30, 30, 30...". The Cisco Catalyst 9100 Access Points perform an interface reset after a 100-second timeout, which in turn resets the timers on the associated switch port to which they are connected.

Topology

Topology details

This topic explains how an AP acts as an 802.1X supplicant and how a switch uses a RADIUS server that supports EAP-FAST, EAP-TLS, and EAP-PEAP to authenticate it.

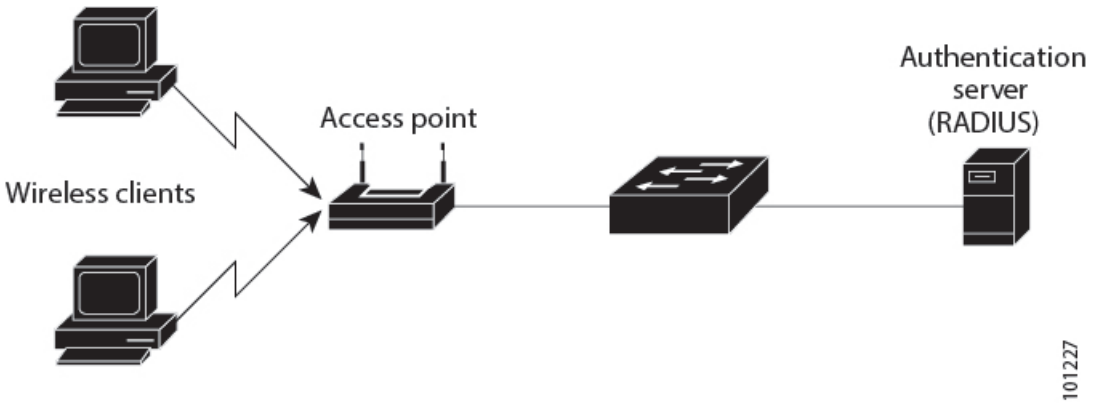
Summary

When dot1x authentication is enabled on a switch port, the connected device must authenticate itself to send and receive data other than 802.1X traffic.

- For EAP-FAST authentication, configure the RADIUS server's credentials at the controller and pass them to the AP through a configuration update request.
- For EAP-TLS or EAP-PEAP, the APs use certificates provided by the local CA server.

Workflow

Figure 4: Topology for 802.1X authentication



Configuring 802.1X authentication type and LSC AP authentication type (GUI)

Configuring 802.1X Authentication Type and LSC AP Authentication Type using GUI.

Complete this task to configure 802.1X authentication type and LSC AP authentication type.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. On the **AP Join Profile** page, click **Add**.

   The **Add AP Join Profile** page is displayed.
3. In the **AP** > **General** tab, navigate to **AP EAP Auth Configuration**.
4. From the **EAP Type** drop-down list, choose the EAP type as **EAP-FAST**, **EAP-TLS**, or **EAP-PEAP** to configure the dot1x authentication type.
5. From the **AP Authorization Type** drop-down list, choose either **CAPWAP DTLS +** or **CAPWAP DTLS**.
6. Click **Save & Apply to Device**.

The 802.1X authentication type and LSC AP authentication type are configured.

Configure 802.1X authentication type and LSC AP authentication type (CLI)

Configure 802.1X authentication type and LSC AP authentication type using commands.

Complete this task to configure 802.1X authentication type and LSC AP authentication type.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Specify a profile name.

Example

```
Device(config)# ap profile ap-profile-name
```

3. Configure the dot1x authentication type.

   Example

   ```
   Device(config-ap-profile)# dot1x {max-sessions | username | eap-type |
   lsc-ap-auth-state}
   ```

   Here

   - **max-sessions**: Configures the maximum 802.1X sessions initiated per AP.
   - **username**: Configures the 802.1X username for all APs.
   - **eap-type:**: Configures the dot1x authentication type with the switch port.
   - **lsc-ap-auth-state**: Configures the LSC authentication state on the AP.

4. Configure the dot1x authentication type as EAP-FAST, EAP-TLS, or EAP-PEAP.

   Example

   ```
   Device(config-ap-profile)# dot1x eap-type {EAP-FAST | EAP-TLS | EAP-PEAP}
   ```

5. Configure the LSC authentication state on the AP.

   Example

   ```
   Device(config-ap-profile)# dot1x lsc-ap-auth-state {CAPWAP-DTLS |
   Dot1x-port-auth | Both}
   ```

   Here

   - **CAPWAP-DTLS**: Uses LSC only for CAPWAP DTLS.
   - **Dot1x-port-auth**: Uses LSC only for dot1x authentication with port.
   - **Both**: Uses LSC for both CAPWAP-DTLS and Dot1x authentication with port.

6. Exit the AP profile configuration mode and enter the privileged EXEC mode.

   Example

   ```
   Device(config-ap-profile)# end
   ```

The 802.1X authentication type and LSC AP authentication type have been configured.

```
Device# configure terminal
Device(config)# ap profile ap-profile-name
Device(config-ap-profile)# dot1x eap-type
Device(config-ap-profile)# dot1x eap-type EAP-TLS
Device(config-ap-profile)# dot1x lsc-ap-auth-state Dot1x-port-auth
Device(config-ap-profile)# end
```

Configure the 802.1X username and password (GUI)

Describes the GUI steps to configure the 802.1X username and password.

Complete these steps to configure the 802.1X username and password.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. On the **AP Join** page, click the name of the AP Join profile or click **Add** to create a new one.
3. Click the **Management** tab and then click **Credentials**.
4. Enter the local username, password details, and choose the appropriate local password type.
5. Enter 802.1X username and password details.
6. Choose the appropriate 802.1X password type.
7. Enter the time in seconds after which the session should expire.
8. Enable local credentials or 802.1X credentials, or both, as required.
9. Click **Update & Apply to Device**.

The configuration of the 802.1X username and password is complete.

Configure the 802.1X username and password (CLI)

Describes the steps to configure the 802.1X username and password.

Complete these steps to configure the 802.1X username and password.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify a profile name for the AP.

   Example

   ```
   Device(config)# ap profile ap-profile-name
   ```

3. Configure the dot1x authentication type.

   Example

   ```
   Device(config-ap-profile)# dot1x {max-sessions | username | eap-type |
   lsc-ap-auth-state}
   ```

   Here

   - **max-sessions**: Configures the maximum 802.1X sessions initiated per AP.
   - **username**: Configures the 802.1X username for all APs

- **eap-type:**: Configures the dot1x authentication type with the switch port.
- **lsc-ap-auth-state**: Configures the LSC authentication state on the AP.

4. Configure the dot1x password for all the APs.

Example

```
Device(config-ap-profile)# dot1x username username password {0 | 8} password
```

Here

- **0**: Specifies an unencrypted password will follow.
- **8**: Specifies an AES encrypted password will follow.

The configuration of the 802.1X username and password is complete.

```
Device# configure terminal
Device(config)# ap profile ap-profile-name
Device(config-ap-profile)# dot1x eap-type
Device(config-ap-profile)# dot1x username username password 0 password
```

Enable 802.1X on the switch port (CLI)

Describes the steps to enable 802.1X on the switch port.

Complete these steps to enable 802.1X on the switch port.

Procedure

---

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Enable AAA.

Example

```
Device(config)# aaa new-model
```

3. Create authentication methods to determine user privilege for accessing the privileged command level, enabling the device to communicate with the AAA server.

Example

```
Device(config)# aaa authentication dot1x {default | listname}
method1[method2...]
```

4. Enable AAA authorization for network services on 802.1X.

Example

```
Device(config)# aaa authorization network group
```

5. Enable 802.1X port-based authentication, globally.

   Example

   ```
   Device(config)# dot1x system-auth-control
   ```

6. Enter the interface configuration mode and specify the interface to be enabled for 802.1X authentication.

   Example

   ```
   Device(config)# interface type slot/port
   ```

7. Enable 802.1X port-based authentication on the interface.

   Example

   ```
   Device(config-if)# authentication port-control {auto | force-authorized |
   force-unauthorized}
   ```

   Here are the options:

   - **auto**: Enables IEEE 802.1X authentication and causes the port to begin in the unauthorized state, allowing only EAPOL frames to be sent and received through the port. The authentication process begins when the link state of the port changes from down to up or when an EAPOL-start frame is received. The Device requests the identity of the supplicant and begins relaying authentication messages between the supplicant and the authentication server. Each supplicant attempting to access the network is uniquely identified by the Device by using the supplicant MAC address.
   - **force-authorized**: Disables IEEE 802.1X authentication and causes the port to change to the authorized state without any authentication exchange required. The port sends and receives normal traffic without IEEE 802.1X-based authentication of the client. This is the default setting.
   - **force-unauthorized**: Causes the port to remain in the unauthorized state, ignoring all attempts by the supplicant to authenticate. The Device cannot provide authentication services to the supplicant through the port.

8. Enable 802.1X authentication on the port with default parameters.

   Example

   ```
   Device(config-if)# dot1x pae [supplicant | authenticator | both]
   ```

9. Enter the privileged EXEC mode.

   Example

   ```
   Device(config-if)# end
   ```

802.1X is enabled on the switch port.

```
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authentication dot1x default group radius
Device(config)# aaa authorization network group
Device(config)# dot1x system-auth-control
Device(config)# interface fastethernet2/1
Device(config-if)# authentication port-control auto
Device(config-if)# dot1x pae authenticator
Device(config-if)# end
```

Verify 802.1X on the switch port

To display the authentication state of 802.1X on the switch port, use the following command:

```
Device# show dot1x all
Sysauthcontrol              Enabled
Dot1x Protocol Version      2
Dot1x Info for FastEthernet1
-----------------------------------
PAE                         = AUTHENTICATOR
PortControl                 = AUTO
ControlDirection            = Both
HostMode                    = MULTI_HOST
ReAuthentication            = Disabled
QuietPeriod                 = 60
ServerTimeout               = 30
SuppTimeout                 = 30
ReAuthPeriod                = 3600 (Locally configured)
ReAuthMax                   = 2
MaxReq                      = 2
TxPeriod                    = 30
RateLimitPeriod             = 0
```

Verify the authentication type

To display the authentication state of an AP profile, use the following command:

```
Device#show ap profile default-ap-profile detailed
        AP Profile Name        : default-ap-profile
        Description            : default ap profile
        …
        Dot1x EAP Method       : [EAP-FAST/EAP-TLS/EAP-PEAP/Not-Configured]
        LSC AP AUTH STATE      : [CAPWAP DTLS / DOT1x port auth / CAPWAP DTLS
 + DOT1x port auth
```

Verify access ports dual port authentication

This section gives the details about the verification of access ports dual port authentication.

Verify access ports dual port authentication

To verify access ports dual port authentication, use the following commands:

```
Device# show ap profile name default-ap-profile detailed
AP Profile Name               : default-ap-profile
Description                   : default ap profile
```

```
Country code                    : Not configured
Stats Timer                     : 180
Link Latency                    : DISABLED
Data Encryption                 : DISABLED
LED State                       : ENABLED
NTP server                      : 0.0.0.0
NTP Authentication              : DISABLED
Jumbo MTU                       : DISABLED
24ghz Report Interval           : 90
5ghz Report Interval            : 90
bssid stats status              : DISABLED
bssid stats frqncy interval     : 30
bssid neighbor stats status     : DISABLED
bssid neighbor stats interval : 180
CAPWAP Control Aggregation      : DISABLED
NSI Ports State                 : DISABLED
POE :
  PreStandard 802.3af Switch  : DISABLED
  Power Injector State        : DISABLED
  Power Injector Selection    : Unknown
  Injector Switch Mac         : Not Configured
Device Management :
  Telnet                      : DISABLED
  SSH                         : ENABLED
  Serial Console              : ENABLED
User Management :
  Username                    : lab
TCP MSS :
  Adjust MSS                  : ENABLED
  TCP Adjust MSS              : 1250
CAPWAP Timer :
  Heartbeat Timeout           : 30
  Discovery Timeout           : 10
  Fast Heartbeat Timeout      : 0
  Primary Discovery Timeout   : 120
  Primed Join Timeout         : 0
Retransmit Timer :
  Count                       : 5
  Interval                    : 3
Login Credentials :
  Local Username              :
  Dot1x Username              : cisco
Ap eap auth info :
  Dot1x EAP Method            : EAP-FAST
  LSC AP AUTH STATE           : CAPWAP DTLS
Syslog :
  Facility Value              : FACILITY_KERN
  Host                        : 255.255.255.255
  Log Level                   : SYSLOG_LEVEL_INFORMATION
  Secured                     : DISABLED
Proxy :
  Hostname                    : Not Configured
  Proxy Port                  : Not Configured
  NO_PROXY list               : Not Configured
  Username                    : Not Configured
Backup Controllers :
  Fallback                    : ENABLED
  Primary Backup Name         : Not Configured
  Primary Backup IP           :
  Secondary Backup Name       : Not Configured
  Secondary Backup IP         :
Hyperlocation :
```

```
  Admin State                  : DISABLED
  PAK RSSI Threshold Detection: -100
  PAK RSSI Threshold Trigger  : 10
  PAK RSSI Threshold Reset    : 8
Halo BLE Beacon :
  Interval                     :
  TX Power                     :
  Enabled                      : Unknown
  Apply Global                 : Unknown
Group NAS Id                   : Not Configured
CDP                            : ENABLED
TFTP Downgrade :
  IP Address                   : 0.0.0.0
  Filename                     : Not Configured
Time Limit                     : 0
Capwap window size             : 1
AP packet capture profile      : Not Configured
AP trace profile               : Not Configured
Mesh profile name              : default-mesh-profile
Urwb profile name              : Not Configured
Power profile name             : Not Configured
Calendar Profile
Method-list  name              : Not Configured
Packet Sequence Jump DELBA     : ENABLED
Lag status                     : DISABLED
Extended Module                : DISABLED
USB Module                     : ENABLED
Persistent SSID Broadcast      : DISABLED
DHCP server                    : DISABLED
Preferred Mode                 : DISABLED
CAPWAP UDP-Lite                : DISABLED
AWIPS                          : DISABLED
AWIPS Forensic                 : DISABLED
Fallback to DHCP               : ENABLED
DTLS server preference         : ENABLED
Maximum client limit           : 0
Location Configuration :
  FTM                          : ENABLED
  FTM initiator burst size     : 16
  FTM initiator burst duration: 32ms
  UWB                          : ENABLED
  UWB initiator burst size     : 32
  UWB initiator burst duration: 10
Kernel core dump :
  Kernel core dump limit        : 5
  Kernel core dump type         : Disabled
  Kernel core dump dir-limit    : 15
Client RSSI Statistics
  Reporting                    : ENABLED
  Reporting Interval           : 30 seconds
Traffic Distribution Stats     : ENABLED
Traffic Distribution Stats Interval(sec): 300
OEAP Mode Config
  Link Encryption               : ENABLED
  Rogue Detection               : DISABLED
  Local Access                  : ENABLED
  Provisioning SSID             : ENABLED
AP broken antenna detection :
  Status                       : DISABLED
RLAN Configurations :
  Fast Switching                : DISABLED
Onboarding configuration       : Unicast
```

```
Pressure sensor :
.
.
.
```

```
Device# show ap name cisco-ap-name config general
Cisco AP Name   : APXXXD.BXXC.1XXX
================================================

Cisco AP Identifier                              : 687d.b45f.2ae0
Country Code                                     : Multiple Countries : US,VN
Regulatory Domain Allowed by Country          : 802.11bg:-AE^   802.11a:-ABES^
   802.11 6GHz:-B
Radio Authority IDs                              : FCC:LDKMU6CR2417
AP Country Code                                  : US  - United States
AP Regulatory Domain
  802.11bg                                       : -A
  802.11a                                        : -B
  802.11 6GHz                                    : -B
License Type                                     : AIR
License State                                    : --
Non Compliance Reason                            : --
MAC Address                                      : 687d.b45c.1900
IP Address Configuration                         : DHCP
IP Address                                       : 198.51.100.1
IP Netmask                                       : 255.255.0.0
Gateway IP Address                               : 203.0.113.1
CAPWAP Path MTU                                  : 1485
Capwap Active Window Size                        : 1
Telnet State                                     : Disabled
CPU Type                                        :  ARMv8 Processor rev 4 (v8l)
Memory Type                                      : DDR4
Memory Size                                      : 1775616 KB
SSH State                                        : Enabled
Serial Console State                             : Enabled
Cisco AP Location                                : default location
AP Floor ID                                      : 0
AP Location Mode                                 : Unknown
Site Tag Name                                    : ST-Flex
RF Tag Name                                      : default-rf-tag
Policy Tag Name                                  : PT-Flex-Local
AP join Profile                                  : default-ap-profile
Flex Profile                                     : default-flex-profile
Primary Cisco Controller Name                    : Not Configured
Primary Cisco Controller IP Address              : 0.0.0.0
Secondary Cisco Controller Name                  : Not Configured
Secondary Cisco Controller IP Address            : 0.0.0.0
Tertiary Cisco Controller Name                   : Not Configured
Tertiary Cisco Controller IP Address             : 0.0.0.0
Administrative State                             : Enabled
Operation State                                  : Registered
NAT External IP Address                          : 198.51.100.1
AP Certificate type                              : Manufacturer Installed
Certificate
AP Certificate Expiry-time                       : 08/09/2099 20:58:26
AP Certificate issuer common-name                : High Assurance SUDI CA
AP Certificate Policy                            : Default
AP CAPWAP-DTLS LSC Status
    Certificate status                           : Not Available
AP 802.1x LSC Status
    Certificate status                           : Not Available
AP LSC authentication state                      : CAPWAP-DTLS
```

```
AP Mode                                          : FlexConnect
AP VLAN tagging state                            : Disabled
AP VLAN tag                                      : 0
CAPWAP Preferred mode                            : IPv4
CAPWAP UDP-Lite                                  : Not Configured
AP Submode                                       : Not Configured
Office Extend Mode                               : Disabled
Link-Encryption                                  : Disabled
Dhcp Server                                      : Disabled
Remote AP Debug                                  : Disabled
Logging Trap Severity Level                      : information
Logging Syslog facility                          : kern
Software Version                                 : 17.18.0.33
Boot Version                                     : 1.1.2.4
Mini IOS Version                                 : 0.0.0.0
Stats Reporting Period                           : 180
LED State                                        : Enabled
MDNS Group Id                                    : 0
MDNS Rule Name                                   :
MDNS Group Method                                : None
PoE Pre-Standard Switch                          : Disabled
PoE Power Injector MAC Address                   : Disabled
Power Type/Mode                                  : PoE/Full Power
Number of Slots                                  : 4
AP Model                                         : C9136I-B
IOS Version                                      : 17.18.0.33
Reset Button                                     : Disabled
AP Serial Number                                 : FOC25322JK8
Management Frame Validation                      : Capable
Management Frame Protection                      : Not capable
AP User Name                                     : lab
AP 802.1X User Mode                              : Global
AP 802.1X User Name                              : cisco
Cisco AP System Logging Host                     : 255.255.255.255
Cisco AP Secured Logging TLS mode                : Disabled
AP Up Time                                       : 4 days 21 hours 17 minutes
34 seconds
AP CAPWAP Up Time                                : 4 days 21 hours 14 minutes
42 seconds
Join Date and Time                               : 04/10/2025 20:13:36
Join Taken Time                                  : 2 minutes 51 seconds
Join Priority                                    : 1
AP Link Latency                                  : Disable
AP Lag Configuration Status                      : Disabled
Lag Support for AP                               : Yes
Rogue Detection                                  : Enabled
Rogue Containment auto-rate                      : Disabled
Rogue PMF Denial                                 : Disabled
Rogue Containment of standalone flexconnect APs  : Disabled
Rogue Detection Report Interval                  : 10
Rogue AP minimum RSSI                            : -90
Rogue AP minimum transient time                  : 0
AP TCP MSS Adjust                                : Enabled
AP TCP MSS Size                                  : 1250
AP IPv6 TCP MSS Adjust                           : Enabled
AP IPv6 TCP MSS Size                             : 1250
Hyperlocation Admin Status                       : Disabled
Retransmit count                                 : 5
Retransmit interval                              : 3
Kernel core dump :
  Configured limit                               : 5
  Kernel core dumps collected on AP              : 0
```

```
  Kernel core dump type                          : Disabled
  Kernel core dump directory limit               : 15
AP dual port authentication :
  Port                                           : 0
  Status                                         : Authorized
  EAP status code                                : Success
AP dual port authentication :
  Port                                           : 1
  Status                                         : Authorized
  EAP status code                                : Success
MLO Capability                                     : Not Capable
Fabric status                                      : Disabled
FIPS status                                        : Disabled
WLANCC status                                      : Disabled
USB Module Type                                    : USB Module
USB Module State                                   : Enabled
USB Module Connected                               : No
USB Operational State Reason                       : Not Detected
USB Override                                       : Disabled
GAS rate limit Admin status                        : Disabled
WPA3 Capability                                    : Enabled
EWC-AP Capability                                  : Disabled
AWIPS Capability                                   : Enabled
AID Management Capability                          : Enabled
Proxy Hostname                                     : Not Configured
Proxy Port                                         : Not Configured
Proxy NO_PROXY list                                : Not Configured
Proxy Username                                     : Not Configured
GRPC server status                                 : Disabled
Unencrypted Data Keep Alive                        : Enabled
Local DHCP Server                                  : Disabled
Traffic Distribution Statistics Capability         : Enabled
Dual DFS Statistics                                : Disabled
AP Upgrade Out-Of-Band Capability                  : Enabled
AP statistics                                      : Disabled
AP power derate Capability                         : Enabled
AP PMK Propagation Capability                      : Enabled
AP FTM Responder Capability                        : Enabled
AP FTM Initiator Capability                        : Enabled
AP UWB Responder Capability                        : Disabled
AP UWB Initiator Capability                        : Disabled
AP Client FTM capability                           : Enabled
URWB Capability                                    : Capable
6GHz Standard-Power mode                           : Not Allowed
Mesh DCA Run Status                                : N/A
Last Mesh DCA Run                                  :
Meraki Capable AP                                  : No
Radio Reset Reason Statistics
Slot 0
  Unknown Reason                                                    : 1

  AP software interface coming up                                   : 1

  State change of HE param to UP                                    : 1

  State change of HE param to DOWN                                  : 1

Slot 1
  Unknown Reason                                                    : 1

  AP software interface coming up                                   : 1
```

```
  State change of DFS channel to UP                                    : 1

  State change of DFS channel to DOWN                                  : 1

  State change of HE param to UP                                       : 2

  State change of HE param to DOWN                                     : 2

  Hardware mode change                                                 : 3

Slot 2
  Unknown Reason                                                       : 1

Slot 3
  Unknown Reason                                                       : 1

  AP software interface coming up                                      : 1

  State change of DFS channel to UP                                    : 1

  State change of DFS channel to DOWN                                  : 1

  State change of HE param to UP                                       : 1

  State change of HE param to DOWN                                     : 1


Radio Failure Reason Statistics
Slot 0
Slot 1
Slot 2
Slot 3
AP image integrity
  Time                                          : 04/10/2025 20:09:46
  Alterative image loaded                       : No
  Backup image status
    Version                                     : 17.18.0.33
    Partition                                   : part2
    Kernel                                      : Good
    Root FS                                     : Good
    IOX                                         : Good
  Primary image status
    Version                                     : 17.18.0.33
    Partition                                   : part1
    Kernel                                      : Good
    Root FS                                     : Good
    IOX                                         : Good
```

Access point client ACL counters

Explains the AP Client ACL Counter feature that provides statistical counts for client ACL rules.

An access point client ACL counter is a statistical tracking feature that

- provides a count for client ACL rules to determine which rule was passing or dropping packets
- enables counting of packets that hit a specific rule in the client ACL, and
- is supported in FlexConnect mode and local switching central authentication sub-mode.

AP commands for ACL counter management

Use these AP commands to enable and manage the counter in the AP:

- **[no] debug flexconnect access-list counter [all | vlan-ACL | client-ACL]**
- **[no] debug flexconnect access-list event [all | vlan-ACL | client-ACL]**

To clear ACL counters use this command:

- **clear counters access-list client** *<MAC>* **all**

> **Note**
>
> This feature is available from the Cisco IOS XE Dublin 17.13.1 release. Until the Cisco IOS XE Dublin 17.12.1 release, there was no per-rule counter to determine which rule was passing or dropping the packets.

Feature history for Access Point Client ACL Counter

Lists the release and related information for the Access Point Client ACL Counter feature.

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 27: Feature History for Access Point Client ACL Counter

Release	Feature	Feature Information
Cisco IOS XE Dublin 17.13.1	Access Point Client ACL Counter	The AP Client ACL Counter feature provides a statistical count for client ACL rules. This feature allows you to count the number of packets that hit a specific rule in the client ACL.

AP MAC authorization

This topic describes AP MAC authorization.

AP MAC authorization is a security feature that:

- ensures only authorized APs can associate with the controller,
- requires registration of the AP's Ethernet MAC address, and
- it is configured locally on the controller or on an external RADIUS server.

With the Cisco IOS-XE 17.18 release, APs support MAC address authorization using different delimiter formats to enhance the controller's support, which previously did not accept any delimiter.

IOS AAA config **mac-delimiter and subscriber mac-filtering security-mode** <> under AAA group server RADIUS is enabled for this feature for wireless AP join cases.

Note

To authenticate an AP using its MAC address:

- Set the *mac-filter* flag to **yes**. This setting configures AAA to send the username with the specified delimiter and sets the MAC filter flag.
- Configure AAA server groups that include the IP addresses of the selected server hosts. This configuration allows you to group existing server hosts, select a subset of the configured server hosts, and use them for a particular service.
- By arranging server groups and configuring them using the global server-host list, you can manage services with a more structured and secure approach.

Feature history for AP MAC authorization

AP MAC authorization feature history.

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 28: Feature history for access point upgrade management enhancements

Release	Feature Information
Cisco IOS XE 17.18.1	AP MAC authorization enables APs to gain support for MAC address authorization using various delimiter formats. The feature improves the controller's support for this security feature. To enable this functionality for wireless AP join cases, the **mac-delimiter** and **subscriber mac-filtering security-mode** configurations under the AAA group server RADIUS are utilized.

Create a AAA RADIUS server (GUI)

This topic describes how to create a AAA RADIUS server using the GUI.

Configure a AAA RADIUS server using the GUI.

Procedure

1. Choose **Configuration** > **Security** > **AAA** > **Server Groups** > **RADIUS**.
2. Click the **Add** button.
   The **Create AAA Radius Server Group** dialog box is displayed.
3. Enter a name for the RADIUS server in the **Name** field.
4. Enter the IPv4/IPv6 server address in the **Server Address**.
5. Optional: Choose **Clear Text** from the **Key Type** drop-down list.
6. Add **Key**.

> **Note**
>
> If global encryption is enabled, then all keys or passwords are encrypted.

7. Match the **Key** with the **Confirm Key** field.
8. Click **Apply to Device**.

Configure a AAA RADIUS server (CLI)

This topic describes how to configure a AAA RADIUS server using the CLI.

Configure a AAA RADIUS server using the CLI.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the RADIUS server name to enter the RADIUS server configuration mode.

   Example

   ```
   Device(config)# radius server ISE
   ```

   ISE is the user-defined string.

   a) Specify the RADIUS server parameters.

      Example

      ```
      Device(config-radius-server)# address ipv4 209.165.201.1 auth-port 1812
      acct-port 1813
      ```

      For **auth-port** port number, specify the UDP destination port for authentication requests. The range is from 0 to 65536 and the default is 1812.

      For **acct-port** port number, specify the UDP destination port for accounting requests. The default is 1813.

   b) Configure the RADIUS per-server encryption key.

      Example

      ```
      Device(config-radius-server)#  key rad123
      ```

3. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

Example

```
Device(config-sg-radius)# end
```

Create a AAA Group configuration (GUI)

This topic describes how to create a AAA Group configuration using the GUI.

Create a new AAA group using the GUI.

Procedure

1. Choose **Configuration** > **Security** > **AAA** > **Server Groups** > **RADIUS** > **Server Groups**.
2. Click the **Add** button.
   The **Create AAA Radius Server Group** dialog box is displayed.
3. Enter a name for the RADIUS server in the **Name** field.
4. Add RADIUS as **Group Type**.
5. Select a **MAC-Delimiter** from the drop-down list.

   The available options are **colon**, **hyphen**, **single-hyphen**, and **none**.
6. Select **MAC-Filtering** from the drop-down list.

   The available options are **mac**, **key**, and **none**. The option **none** is supported only when both **MAC-Delimiter** and **MAC-Filtering** are set to the same value.
7. Add the newly created AAA group from **Available Servers** to **Assigned Servers**.
8. Click **Apply to Device**.

Configure a AAA Group configuration (CLI)

This topic describes how to configure a AAA Group configuration using the CLI.

Configure a AAA group configuration using the CLI.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Configure the RADIUS server group name and server group configuration.

   Example

   ```
   Device(config)# aaa group server radius MY-SERVER-GROUP
   ```
3. Re-format the delimiter of the username attribute.

Example

```
Device(config-sg-radius)# mac-delimiter {colon | hyphen | none | single-hyphen}
```

The options are:

a. **colon**: Sets the delimiter to colon (for example, xx:xx:xx:xx:xx:xx).

b. **hyphen**: Sets the delimiter to hyphen (for example, xxxxxxxxxxxx).

c. **none**: Sets the delimiter to none (for example, xx-xx-xx-xx-xx-xx).

d. **single-hyphen**: Sets the delimiter to single-hyphen (for example, xxxxxx-xxxxxx).

> **Note**
>
> The default delimiter is **none**. If nothing is configured, **none** is set to the username, that is, the MAC address goes without any delimiter to the respective RADIUS server.

4. Set the MAC address as the password.

Example

```
Device(config-sg-radius)#subscriber mac-filtering security-mode {mac | none
| shared secret}
```

The options are:

a. **mac**: The delimiter format of the MAC address in the RADIUS user password attribute is same as that of the MAC address string in the RADIUS username attribute.

b. **none**: MAC address string without any delimiter is sent as RADIUS user password.

c. **shared-secret**: The shared-secret option is not supported for the AP join use case.

> **Note**
>
> The default delimiter is **none**. If nothing is configured, the user password attribute is sent as the MAC address without delimiter. If **mac** is configured as the delimiter, then the delimiter format of the username attribute is applied to the MAC address string of the user password attribute.

5. Associate the RADIUS server to the server group.

Example

```
Device(config-sg-radius)# server name ISE
```

6. Configure the authorization method list with the named RADIUS server group.

Example

```
Device(config)# aaa authorization credential-download MY-METHOD-LIST group
MY-SERVER-GROUP
```

7. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

Example

```
Device(config-sg-radius)# end
```

Create a AAA Group Server configuration (GUI)

Create a AAA Group My Server configuration using the GUI.

Procedure

1. Choose **Configuration** > **Security** > **AAA** > **AAA Method List** > **Authorization**.
2. Click the **Add** button.
   The **Quick Setup: AAA Authorization** dialog box is displayed.
3. Enter a name in the **Method List Name** field.
4. Select **Type** as **Credential-Download** from the drop-down list.
5. Select **Group Type** as **Group** from the drop-down list.
6. Add the newly created AAA group from **Available Server Groups** to **Assigned Server Groups**.
7. Click **Apply to Device**.

Create a AAA Group Method List configuration (GUI)

This topic describes how to create a AAA Group Method List configuration using the GUI.

Create a AAA group method Method List configuration using the GUI.

Procedure

1. Choose **Configuration** > **Security** > **AAA** > **AAA Advanced** > **AP Policy**.
2. Use the toggle button to enable **Authorize APs against MAC** and configure **ap auth-list authorize-mac** on your controller.
3. Select AP-MAC-Method for **Authorization Method List** from the drop-down list to configure **ap auth-list method-list MY-METHOD-LIST** on your controller.
4. Click **Apply**.

Configure AP MAC authorization (CLI)

This topic describes how to configure AP MAC authorization using the CLI.

Configure AP MAC authorization using the CLI.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure AP authorization list.

   Example

   ```
   Device(config)# ap auth-list {authorize-mac | method-list method-list-name}
   ```

   The options are:

   - **authorize-mac**: Sets AP authorization policy with MAC.
   - **method-list**: Sets AP authorization method-list.

3. Enter the method-list to be used for AP MAC authorization.

   Example

   ```
   Device(config)# ap auth-list method-list MY-METHOD-LIST mac
   ```

4. Save the configuration and return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-auth-list)# exit
   ```

Verify MAC authorization details

Verify the AP authentication list.

Verify AP authentication list

To verify if the controller has enabled the AP authentication list, run the **show ap auth-list** command.

```
Device# show ap auth-list

Authorize APs against MAC : Disabled
Authorize APs against Serial Num : Enabled
Authorization Method List : <auth-list-name>
```

Verify AP status summary

To verify the status summary of all Cisco lightweight APs attached to the device, run the **show ap summary** command.

```
Device# show ap summary

Number of APs: 1
```

```
Global AP User Name: Cisco
Global AP Dot1x User Name: Not configured

AP Name                          AP Model  Ethernet MAC    Radio MAC
State
-----------------------------------------------------------------------------------
3602a                            3502I     003a.99eb.3fa8  d0c2.8267.8b00
Registered
```

24 AP Joining

Topics:

- Failover priority for APs
- AP Plug-n-Play
- AP fallback to controllers using AP priming profile

Failover priority for APs

Explains how failover priorities for access points determine which devices maintain connectivity when controller capacity is limited.

A failover priority for APs is a configuration setting that:

- assigns priority levels to APs for controller association
- determines which APs will be favored when controller capacity is limited, and
- allows administrators to ensure high-priority devices maintain connectivity during failover scenarios

Each controller supports a set number of AP connections. If you deploy multiple controllers on the network and one fails, your APs can automatically connect to another available controller.

Guidelines to configure AP failover priority

Use these guidelines to configure AP failover priority:

- Failover priority does not impact normal wireless operation. It only applies when AP connection requests exceed the controller's available capacity.
- The system checks AP priority as each AP connects. When capacity is full, or the primary controller fails, APs connect to the secondary controller.
- Enable failover priority and set priority levels for APs that need them.
- By default, all APs have priority level one (lowest). Assign higher levels only where needed.

Set AP priority (GUI)

Set AP priority using GUI.

Assign a failover priority to an AP to ensure high availability in the event of a controller failure.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click the Access Point.
3. In the **Edit AP** dialog box, go to the **High Availability** tab.
4. Select the priority from the **AP failover priority** drop-down list.
5. Click **Update and Apply to Device**.

Set AP priority (CLI)

Set AP priority using commands.

Assign a priority level to an AP to determine the order in which APs connect and operate using commands.

> **Note**
>
> Priority of APs ranges from one to four, with four being the highest.

Procedure

1. Specify the priority of an AP.

   Example

   ```
   Device# ap name ap-name AP44d3.ca52.48b5 priority 1
   ```

2. Display common information for all APs.

   Example

   ```
   Device# show ap config general
   ```

3. Display the configuration of a particular AP.

   Example

   ```
   Device# show ap name ap-name AP44d3.ca52.48b5 config general
   ```

AP Plug-n-Play

Explains the capabilities and configuration options that AP Plug-n-Play features provide for deploying access points.

AP Plug-n-Play (PnP) features are configuration capabilities that

- deliver staging parameters to APs before they join a controller
- allow runtime configuration transfer when an AP joins the controller using the staging configuration, and
- enable the PnP server to provide all tag-related information. You provide this as part of the preconfigured information to the AP and then to the controller.

You can upload configuration to the PNP server in either **TXT** or **JSON** format. You can also add the AP details. The AP details are mapped to the information in the **TXT** or **JSON** configuration file. While provisioning the AP from the PnP server, the AP acquires these configuration details. Based on the configuration details, the AP joins the corresponding controller with the tag information.

Provision AP from PnP server

Provision AP from PnP server.

You can provision AP from PnP Server in either ways:

- Configure DHCP server or switch with **Option 43**. For example, you can refer to the code sample:

```
ip dhcp pool vlan10
network 9.10.10.0 255.255.255.0
default-router 9.10.10.1
option 43 ascii 5A1D;B2;K4;I9.10.60.5;J80
```

- Configure DHCP server with DNS with **domain-name dns.com**. For example, you can refer to the code sample:

```
ip dhcp pool vlan10
network 9.10.10.0 255.255.255.0
default-router 9.10.10.1
dns-server 9.8.65.5
domain-name dns.com
```

Verify AP tag configuration

Verify AP tag configuration.

The example shows how to verify the AP tag configuration:

```
Device# show ap tag summary
Number of APs: 5


AP Name                  AP Mac           Site Tag Name           Policy Tag
Name         RF Tag Name             Misconfigured    Tag Source

-------------------------------------------------------------------------------------------------

APd42c.4482.6102         d42c.4482.6102   default-site-tag
default-policy-tag       default-rf-tag          No               Default


AP00c1.64d8.6af0         00c1.64d8.6af0   named-site-tag
named-policy-tag         named-rf-tag            No               AP
```

Note

The tag source in the second row originates from the PNP server.

AP fallback to controllers using AP priming profile

Explains how AP priming profiles centralize controller assignments and automate AP fallback behavior in large wireless networks.

An AP priming profile is a controller configuration that

- defines primary, secondary, and tertiary controller assignments for APs
- enables centralized assignment of controllers for groups of APs or individual devices, and

- automates fallback and reconnection behavior when APs lose connection with their assigned controllers.

In large networks, configuring AP priming for primary, secondary, and tertiary controllers on each AP is difficult. Simplify AP priming by using the AP fallback to controllers using AP priming profile feature. Use profiles defined in the controller to assign controllers to APs.

To configure primary, secondary, and tertiary controllers, use the AP priming profile under the AP filter profile. Apply it to a group of APs matching regular expressions or to an individual AP by using its Ethernet MAC address. When the AP joins the controller, the controller sends the AP priming configuration to the AP if priming override is enabled.

Note

When the controller sends a priming profile to the AP, the AP disconnects from the current controller and joins the controller in the priming profile. A CAPWAP restart or device reboot is not required.

Feature History

This table provides release and related information for the feature explained in this module.

Table 29: Feature History for AP Fallback to controllers using AP priming profile

Release	Feature	Feature Information
Cisco IOS XE Cupertino 17.9.2	AP Fallback to controllers using AP priming profile	This feature helps to configure primary, secondary, and tertiary controllers for a group of APs matching regular expression or for an individual AP using priming profiles.

AP controller selection and onboarding behavior

Review these points to choose the best AP controller and onboard APs effectively.

- Ensure APs always reconnect to their primary controller as soon as it becomes available; APs automatically disassociate from backup controllers when the primary controller is online. The AP does not switch to secondary controllers even if they are available.
- Configure primary, secondary, and tertiary controllers so that an AP connects to the tertiary controller only if both the primary and secondary controllers are unresponsive. If only the secondary controller recovers while the primary is still down, keep the AP connected to the tertiary controller until the primary is available. If the tertiary controller fails and the primary is still down, switch the AP to the available secondary controller.
- Do not use the AP Onboarding VLAN for client onboarding, as it results in address overlap errors.
- AP scale limits specify the maximum number of APs supported once they have joined. This limit does not determine how many APs can join at the same time or guarantee a particular recovery time. When an N+1 failover or controller recovery affects many APs, they may rejoin in several waves. The time required for all APs to rejoin depends on the number of reconnecting APs and controller and network conditions.

Restrictions for AP fallback to controllers using AP priming profile

AP fallback to controllers using AP priming profile restrictions.

- You must create a filter to complete the AP priming process. Without a filter, the system cannot determine which APs to apply the priority to, and the priming operation will not proceed.
- Rolling AP upgrade does not work if priming override is enabled.
- The maximum number of priming profiles allowed is 128.
- Each profile can be up to 32 ASCII characters long.

Configure AP priming profile (CLI)

Configure AP priming profile using commands.

Set up an AP priming profile to specify primary, secondary, and tertiary controllers for AP fallback.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the profile to prime APs.

   Example

   ```
   Device(config)# wireless profile ap priming profile-name Prime-FX
   ```

3. Configure the name and IP address of the primary controller for AP fallback.

   Example

   ```
   Device(config-priming)# primary controller-name aaaa ip-address 209.165.201.2
   ```

4. Configure name and IP address of the secondary controller for AP fallback.

   Example

   ```
   Device(config-priming)# secondary controller-name bbbb ip-address 209.165.201.3
   ```

5. Configure name and IP address of the tertiary controller for AP fallback.

   Example

   ```
   Device(config-priming)# tertiary controller-name bbbb ip-address 209.165.201.4
   ```

6. Send the priming details to the AP.

   Example

   ```
   Device(config-priming)# priming-override
   ```

Priming override is disabled by default. If priming override is disabled, the AP does not receive the information stored in the AP priming profile.

> **Note**
>
> N+1 upgrade may not work as expected when priming override is enabled. Before using N+1 upgrade, ensure that priming override is disabled using the **no priming-override** command.

Configure AP priming using filters (CLI)

Configure AP priming using filters using commands.

Configure AP priming filters and assign profiles using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the AP filter and sets the type as priming.

   Example

   ```
   Device(config)# ap filter name filter-name test-filter type priming
   ```

   Set the filter type to priming. The default filter type is tag.

   > **Note**
   >
   > To modify the filter type, delete the existing filter and create a new one with the desired type. Delete a filter by using the **no ap filter name** command.

3. Configure the AP name regular expression match.

   Example

   ```
   Device(config-ap-pr-filter)# ap name-regex reg-ex BGL18
   ```

4. Map the priming profile to the filter.

Example

```
Device(config-ap-pr-filter)# profile profile-name Prime-FX
```

5. Return to the global configuration.

Example

```
Device(config-ap-pr-filter)# exit
```

6. Configure priority for a named AP filter.

Example

```
Device(config)# ap filter priority priority 12 filter-name test-filter
```

Configure multiple AP priming profiles, each with a unique priority level. This allows you to set different priority levels for each AP priming profile.

Configure Per-AP priming (CLI)

Configure Per-AP priming using commands.

Assign a priming profile to a specific AP using the commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Enter the AP profile configuration mode.

Example

```
Device(config)# ap mac-address 00:00:5e:00:53:af
```

3. Map a priming profile to an AP.

Example

```
Device(config-ap-tag)# priming profile-name Prime-FX
```

Verify priming filter configuration

Verify priming filter configuration.

To view the list of all the priming filters, use the command:

```
Device# show ap filters all type priming

Filter Name                        regex                                  Priming
 profile
-------------------------------------------------------------------------------------
FLR1                                *AP-FLOOR-1*
AP-PRIMING-1
FLR2                                *AP-FLOOR-2*
AP-PRIMING-2
```

To view the list of all the active priming filters, use the command:

```
Device# show ap filters active type priming

Priority   Filter Name              regex                                 Priming
profile
-------------------------------------------------------------------------------------

 0          FLR2                    *AP-FLOOR-2*
AP-PRIMING-2

 1          FLR1                    *AP-FLOOR-1*
AP-PRIMING-1
```

To view the summary of the priming profiles, use the command:

```
Device# show wireless profile ap priming summary

Number of AP Priming Profiles: 2
Priming profile
-------------------
AP-PRIMING-1
AP-PRIMING-2
```

To view the details of the priming profiles, use the command:

```
Device# show wireless profile ap priming all

Profile Name  Primary Controller Name  Primary Controller IP  Secondary
Controller Name  Secondary Controller IP  Tertiary Controller Name  Tertiary
Controller IP  Override
-------------------------------------------------------------------------------------
AP-PRIMING-1  BGL18-wlc               209.165.201.1          BGL17-wlc
            209.165.201.2                                      0.0.0.0
          Disabled
AP-PRIMING-2  BGL18-wlc               209.165.201.2          BGL17-wlc
            209.165.201.2            BGL12-wlc                 209.165.201.3
          Disabled
```

To view the priming information for each AP, use the command:

```
Device# show ap name ap1 config general | sec Priming

Priming Profile                  : AP-PRIMING-1
Priming Override                 : Disabled
Priming Source                   : MAC/FILTER/NONE
Filter Name                      : FLR1
```

25 AP Crash File Upload

Topics:

- AP crash files

AP crash files

Introduces AP crash file upload, detailing foundational concepts, outlines restrictions for secure and effective operation, and provides instructions for configuring AP crash file upload using CLI commands.

A crash file is a diagnostic file that

- is created when an AP experiences an unexpected reboot
- contains information about the state of the AP at the time of the crash, and
- is stored in the local flash memory of the AP for retrieval.

AP crash files provide data that is essential for diagnosing the causes of unexpected reboots in access points. This data aids in troubleshooting and maintenance.

> **Note**
>
> An unexpected reboot of the AP is considered a crash.

Process crash details

When a process crashes, the device collects various types of data. and compiles it into a single, compressed archive for analysis. This archive includes:

- Full process core
- Trace logs
- Cisco IOS syslogs (not guaranteed in case of nonactive crashes)
- System process information
- Bootup logs
- Reload logs
- Certain types of proc information

Except for the full core and tracelogs, everything else is a text file. This consolidated information provides a snapshot of the AP's state, which can be used for further diagnosis.

Restrictions for AP crash file upload

- Reports are not generated by the system in case of a reload.

Configure AP crash file upload process (CLI)

Set up and configure an access point to collect and upload crash files for diagnostics.

This involves configuring multiple data collection commands and ensuring files are available for troubleshooting access point failures.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Collect AP crash information.

   Example

   ```
   Device# ap name ap-name crash-file get-crash-data
   ```

   The crash file is uploaded automatically after the AP reloads to the ready state. As a result, this command does not have to be manually executed.

3. Collect the AP core dump file for slot 0 or slot 1.

   Example

   ```
   Device# ap name ap-name crash-file get-radio-core-dump slot {0 | 1}
   ```

4. Upload the AP crash coredump file to the given TFTP location.

   Example

   ```
   Device# ap name ap-name core-dump tftp-ip crash-file uncompress
   ```

5. Display your AP crash file and the radio crash file.
6. Display the crash file in bootflash with .crash extension.

   Example

   ```
   Device# dir bootflash
   ```

The AP crash files and core dumps are collected and available in specified locations for troubleshooting.

26 LED States for Access Points

Topics:

- LED states for APs

LED states for APs

Introduces LED state features for access points, guides configuration using GUI and CLI, details AP profile settings, and provides verification commands for monitoring LED status and ensuring proper operation.

A LED state for access points is a configuration that

- allows the identification of specific access points in a LAN network
- can be configured globally or for individual APs, and
- prioritizes global configurations over individual AP settings.

Note

When disabling the LED on an AP, the LED state that is controlled by the AP-Join profile.

To maintain the LED in a disabled state, it is recommended to create a separate AP-Join profile and Site Tag specifically for APs with the LED disabled.

Note

For APs that have Ethernet LEDs in addition to the main system LED, the Ethernet LEDs are enabled or disabled (switched ON or OFF) as per the system LED. For example, if the system LED is ON, the Ethernet LED will also be ON.

Configure the LED State on APs (GUI)

Configure LED state in access points using GUI.

Configure the LED state on access points to optimize visibility and performance using the GUI.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click an AP from the AP list.
   The **Edit AP** window is displayed.
3. In the **General** tab, under the General section, go to **LED Settings**.
   a) Click the box adjacent to the **LED State** field to enable or disable the LED state.
   b) From the **LED Brightness Level** drop-down list, choose a value from 1 to 8.
4. Under the **Flash Settings** section, perform the following steps:
   a) Click the box adjacent to the **Flash State** field to enable or disable the flash state.
   When the flash state is enabled, the current status of the flash and the flash duration fields are displayed.
   b) From the **Flash Duration** drop-down list, choose either **Always ON** or **Timed**.
   If you choose the **Timed** option, the **Time Duration** field is displayed.
   c) In the **Time Duration** field, specify the flash duration time, in seconds. The default value is one second. The range is between 1 and 3600 seconds.

5. Click **Update & Apply to Device**.

The LED state and related settings on the access points are configured according to your selections, enhancing network device visibility or conserving energy.

Configure LED state for APs (CLI)

Configure LED State for APs in the global configuration mode using commands.

Configure the LED state for APs in the global configuration mode using commands.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

   Enter your password if prompted.

2. Enable the LED state for APs, globally.

   Example

   ```
   Device# ap name Cisco-AP-name led
   ```

3. Configure the starting and the stopping of the AP LED flash.

   Example

   ```
   Device# ap name Cisco-AP-name led flash start duration 20
   ```

   ```
   Device# ap name Cisco-AP-name led flash stop
   ```

   The valid start duration range is between zero and 3600 seconds.

4. Configure the LED brightness level.

   Example

   ```
   Device# ap name Cisco-AP-name led-brightness-level 4
   ```

   The brightness value ranges from 1 to 8.

The LED state is successfully configured with adjusted brightness and flash settings according to commands executed.

Configure LED state in the AP profile

Configure LED state in the AP profile using commands.

The task enables the LED state for all APs, ensuring active visual indicators across the network.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the AP profile configuration mode.

   Example

   ```
   Device(config)#ap profile default-ap-profile
   ```

3. Enable the LED-state for all APs.

   Example

   ```
   Device(config-ap-profile)# led
   ```

Verify that the LED status is visible on the access point, confirming successful configuration.

Verify LED state for APs

Verify LED state for APs.

To verify the LED state of the APs, use this command

```
Device# show ap name AXXX-APXXXX.bdXX.f2XX config general
Cisco AP Name : AXXX-APXXXX.bdXX.f2XX
=====================================================
Cisco AP Identifier : 0cXX.bdXX.65XX
Country Code : Multiple Countries : FR,IN,US
Regulatory Domain Allowed by Country : 802.11bg:-AE 802.11a:-ABDEN
AP Country Code : US - United States
AP Regulatory Domain
802.11bg : -A
802.11a : -B
.
.
.
CAPWAP Preferred mode : IPv4
CAPWAP UDP-Lite : Not Configured
AP Submode : WIPS
Office Extend Mode : Disabled
Dhcp Server : Disabled
Remote AP Debug : Disabled
Logging Trap Severity Level : information
Logging Syslog facility : kern
Software Version : 17.X.0.XXX
Boot Version : 1.1.X.X
Mini IOS Version : 0.0.0.0
Stats Reporting Period : 180
LED State : Enabled
MDNS Group Id : 0
.
```

27 AP Support Bundle

Topics:

- AP support bundle

AP support bundle

Introduces Access Point Support Bundle functionality, guiding users through export processes using GUI and CLI methods, and detailing procedures for monitoring the status of support bundle exports for effective wireless issue analysis.

An AP support bundle is a data package that

- contains core files, crash files, show run configuration, configuration commands, message logs, and trap logs, and
- is supported only on Cisco Wave2 APs and Cisco Catalyst APs.

Feature history for AP support bundle
Table 30: Feature history for secure data wipe

Release	Feature	Feature Information
Cisco IOS XE 17.3.1	AP support bundle	You can now retrieve the support bundle information of an AP and export it to the controller or an external server.

The AP Support Bundle feature provides comprehensive diagnostic information for troubleshooting and configuring the AP.

Retrieving support bundle information

Retrieve the support bundle information of your AP and export it to the controller or an external server.

Note

The destination file path for a Cisco Catalyst 9800 controller-initiated SCP copy of an AP's support bundle must be a pre-existing folder name.

```
ap name APNAME export support-bundle mode scp target ip-address 10.11.12.13
 path /
ap name APNAME export support-bundle mode scp target ip-address 10.11.12.13
 path flash:
ap name APNAME export support-bundle mode scp target ip-address 10.11.12.13
 path /BUNDLE
```

Export an AP support bundle (GUI)

Export an access point support bundle using GUI.

You can export an AP support bundle using GUI for analysis or backup.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click the corresponding AP name.

The **Edit AP** window is displayed.

3. Click the **Support Bundle** tab.
4. From the **Destination** drop-down list, choose one of the following:
   - **This Device**: If you choose this, enter the values for the **Server IP**, **Destination File Path**, **Controller Username**, and **Password** fields.

     When you choose **This Device**, a bundle is sent through Secure Copy (SCP) to the controller (if you have configured the **ip scp server enable** command globally on the controller). You can easily retrieve the bundle later from your browser, using the controller file manager.
   - **External Server**: If you choose this, from the **Transfer Mode** drop-down list, choose either **scp** or **tftp**.

     If you choose the **scp** transfer mode, enter the values for the **Server IP**, **Destination File Path**, **Controller Username**, and **Password** fields.

     If you choose the **tftp** transfer mode, enter the values for the **Server IP**, and **Destination File Path** fields.

   Information about the **Last Export Status**, such as **State**, **Transfer Mode**, **Server IP**, **File Path**, and **Time of Export**, is displayed on the right-hand side of the window.
5. Click **Start Transfer** to initiate the export.

Once you complete the transfer, the AP support bundle is ready at your chosen destination for analysis or backup.

Export an AP support bundle (CLI)

Export an AP support bundle using commands.

This task guides you to export a support bundle from an access point using CLI commands. This helps in troubleshooting and support.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

   Enter your password, if prompted.
2. Export the AP support bundle through the SCP or TFTP transfer modes.

   Example

   ```
   Device> ap name Cisco-AP-name export support-bundle mode scp target ip-address
    10.1.1.1 path file-path
   ```

   If you select the **scp**, you will be prompted to provide your username and password.

   For **tftp**, username and password are not required.

The AP support bundle is successfully exported to the specified file path using the selected transfer mode.

Monitor the status of support bundle export

Monitor the status of support bundle export.

To monitor the status of a support bundle export, run this command:

```
Device# show ap support-bundle summary
AP Name    Server-IP   Status        Last Successful Time      Path File-name
──────────────────────────────────────────────────────────────────────────────
AP_28XXX   10.1.1.10   Copy Success  04/24/2020 07:27:38 UTC
AP_28XXX_support.17.4.0.2.2020.07XXXX.tgz
```

28 Access Point Memory Information

Topics:

- Access point memory information

Access point memory information

Introduces key concepts of access point memory, describing the types, organization, and roles of memory within APs, and provides verification steps to review and assess AP memory information.

Access point memory information is a feature that

- allows viewing the memory type of an AP
- provides details about the CPU type of the AP, and
- displays the memory size per AP after single sign-on authentication.

This feature enables APs to report hardware specifications to the controller during the join phase. This enhancing network visibility and aids hardware-related troubleshooting.

Verify AP memory information

Verify access point memory information.

To verify the memory information of a specified AP, including the CPU type, memory type and memory size, use thia command:

```
Device# show ap name AP-NAME config general
Cisco AP Name    : AP-NAME
================================================
Cisco AP Identifier                              : 00XX.f1XX.e0XX
Country Code                                    : Multiple Countries : FR,IN,US
Regulatory Domain Allowed by Country            : 802.11bg:-AE   802.11a:-ABDEN
AP Country Code                                  : US  - United States
AP Regulatory Domain
  802.11bg                                       : -A
  802.11a                                        : -B
.
.
.
CPU Type                                         : ARMv7 Processor rev 1 (v7l)
Memory Type                                      : DDR4
Memory Size                                      : 1028096 KB
.
.
.
```

29 Access Point Tag Persistency

Topics:

- AP tag persistency

AP tag persistency

Introduces AP tag persistency concepts, outlines recommendations for AP rejoin after enabling tag persistency, details requirements for tag consistency across controllers, describes redundancy scenarios, and provides step-by-step configuration and verification procedures in both GUI and CLI environments.

AP tag persistency is a configuration feature that

- allows mapped tags to be saved locally on APs
- enables APs to retain tag configurations when moving between controllers, and
- removes the need for reapplying tags individually on each AP.

Feature history for AP tag persistency
Table 31: Feature history for secure data wipe

Release	Feature	Feature Information
Cisco IOS XE 17.6.1	AP tag persistency	Global tag persistency configuration on the controller that automatically write tags on the APs.

Supporting reference information

- The feature is disabled by default and must be explicitly enabled on the controller.
- When enabled, any AP that joins a controller stores the mapped tags locally.

Recommendation: Rejoin APs after enabling tag persistency

Recommendation for rejoining APs after enabling tag persistency.

After enabling AP tag persistency on a controller, ensure that all existing APs rejoin the controller for the configuration to take effect.

Requirement: Match tags across controllers

To retain tag configuration across different controllers:

- Ensure that the policy tag, site tag, and RF tag exist on all controllers an AP might connect to.
- If all three tags do not exist on a new controller, the AP will use default tags.
- Even if one or two tags exist, tag persistency can still apply using available tags.

Tag persistency for redundancy scenarios

Tag persistency helps in N+1 redundancy scenarios by ensuring that an AP retains its previously assigned tags when moving between controllers. This process assumes that the tags (policy, site, RF) exist on the secondary controller. If all required tags do not exist, the AP reverts to default tags. For more information, see "*New Configuration Model*" chapter in *Cisco Catalyst 9800 Series Wireless Controller Software Configuration Guide*.

Configure access point tag persistency (GUI)

Configure access point tag persistency using GUI.

The purpose of this task is to enable tag persistency on APs. It allows them to retain their tag mapping information without requiring individual configuration after reconnection.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. Click the **AP** tab.
3. In the **Tag Source** tab, check the **Enable AP Tag Persistency** check box to configure AP Tag persistency globally.

   When APs join a controller with the tag persistency enabled, the mapped tags are saved on the AP without having to write the tag configurations on each AP individually.
4. Click **Apply to Device**.

The AP tag persistency is enabled, allowing access points to retain their tag configurations even after disconnections.

Save tags on an access point (GUI)

Save tags on an AP using the GUI.

This task guides you on how to save tags, including policy, site, and RF tags, on an access point using the graphical user interface (GUI). Saving tags ensures the access point remembers its configuration across different controllers.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click an AP from the list.
   The **Edit AP** page is displayed.
3. Click the **General** tab.
4. In the **Tags** section, specify the appropriate policy, site, and RF tags that you created in the **Configuration** > **Tags & Profiles** > **Tags** page.
5. From the **Policy** drop-down list, select a value.
6. From the **Site** drop-down list, select a value.
7. From the **RF** drop-down list, select a value.
8. Check the **Write Tag Config to AP** check box to push the tags to the AP so that the AP can save and remember this information even when the AP is moved from one controller to another.
9. Click **Update & Apply to Device**.

The AP saves the specified tags and configurations, allowing seamless operation even if the AP is transferred between controllers.

Delete saved tags on an access point (GUI)

Delete saved tags on an access point.

The purpose of this task is to guide you to delete saved tags on an access point through the GUI.

Use these steps to delete saved tags on an access point:

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click an AP from the list of APs.

   The **Edit AP** window is displayed.
3. In the **Edit AP** window, choose the **Advanced** tab.
4. In the **Set to Factory Default** section, check the **Clear Resolved Tag Config** check box to clear the saved tags on an AP.
5. Click **Update & Apply to Device**.

The access point is updated and all saved tags are cleared.

Configure AP tag persistency (CLI)

Configure access point tag persistency using commands.

Configure AP tag persistency to ensure APs maintain their assigned tags when connecting to different controllers, which aids in redundancy scenarios.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Configure AP tag persistency

   Example

   ```
   Device(config)# ap tag persistency enable
   ```
3. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

After completing these steps, the AP tag persistency feature will be enabled on the controller, ensuring APs keep their tag assignments when moving to different controllers.

Verify AP tag persistency

Verify access point tag persistency.

To verify AP tag persistency in the primary controller, use this command:

```
Device# show ap tag summary
Number of APs: 1

AP Name         AP Mac           Site Tag Name         Policy Tag Name
 RF Tag Name            Misconfigured    Tag Source
-------------------------------------------------------------------------------
Cisco01_AP      xxxx.xxxx.xxxx   default-site-tag      OpenRoaming
 default-rf-tag         No               Static
```

If the Tag Source displays **Static** or **Filter**, it means that the AP tag mappings were configured on the primary controller. If the source displays **Default**, it means that the AP received the default tags when joining the controller.

To verify the AP tag persistency in the secondary controller, use this command:

```
Device# show ap tag summary
Number of APs: 1

AP Name       AP Mac           Site Tag Name        Policy Tag Name     RF Tag
 Name       Misconfigured   Tag Source
-------------------------------------------------------------------------------
Cisco01_AP   xxxx.xxxx.xxxx    default-site-tag     OpenRoaming
default-rf-tag     No              AP
```

If the Tag Source displays **AP**, it means that the policy tag, site tag, and RF tag match what was configured on the primary controller, indicating that the AP tags have persisted across controllers.

30 AP Power Save

Topics:

- Access point power save

Access point power save

AIntroduces access point power save capabilities, outlines configuration options and supported features across Cisco IOS XE releases, and details enhancements for energy efficiency, flexible policies, and supported access point models.

Access point power save is a power management capability that

- enables APs to reduce energy consumption during periods of inactivity
- supports flexible scheduling and policies for low-power operation, and
- helps optimize network efficiency without compromising performance.

This feature is critical in modern enterprise environments where reducing operational costs and improving environmental impact are priorities. Cisco APs offer multiple configurations to enable power save modes, such as calendar-based scheduling and client threshold triggers

Feature history for access point power save
Table 32: Feature History

Release	Feature Information
Cisco IOS XE 17.18.1	Multi-Link Operation (MLO) is now supported even when radios are operating in low-power mode. This enhancement ensures seamless multi-link operations under constrained power conditions, providing improved flexibility and efficiency for network operations. This feature is supported on the following access points only: • Cisco Wireless 9178 Series Access Points. • Cisco Wireless 9176 Series Access Points.
Cisco IOS XE 17.13.1	AP Power Distribution support in Cisco Catalyst 9124 Series APs.
Cisco IOS XE Dublin 17.10.1	These features are supported: • Radio spatial streams • Flexible Power over Ethernet (PoE) profiles
Cisco IOS XE Cupertino 17.9.1	Feature support for the following APs: • Cisco Catalyst 9164 Series Access Points • Cisco Catalyst 9166 Series Access Points
Cisco IOS XE Cupertino 17.8.1	This feature allows a network administrator to force APs to operate in low-power mode to reduce power consumption.

For more information about the APs that support the AP Power Save feature, see https://www.cisco.com/c/en/us/td/docs/wireless/access_point/feature-matrix/ap-feature-matrix.html.

Tips for AP power save configuration

- Use spatial stream adjustments to optimize power: From Cisco IOS-XE Cupertino 17.10.1 onwards, you can shut down AP radios or lower the radio spatial streams, to reduce usage of power. You can enforce radio speed by configuring the number of spatial streams on the radios. The combinations for radio spatial stream policies are: 1X1, 2X2, 3X3 (only for Cisco Catalyst 9130 Series Access Points), 4X4, and 8X8.
- Use calendar profiles for predictable power reduction: You can define regular sleep intervals using calendar profiles to maximize energy savings without disrupting business hours.

Requirements for AP power save configuration

- Enable RLAN port before using PoE-out: PoE-out functionality requires the RLAN port to be enabled. Ensure this is configured before applying any PoE-out power profile.

Best practice for AP power save configuration

- Setting client thresholds: Set the client wakeup threshold to a realistic minimum (e.g. 3–5 clients) to avoid unnecessary power cycling and improve user experience.

Warnings for AP power save configuration

- Do not disable uplink interfaces: Do not configure the AP uplink Ethernet port for shutdown in any power profile. The uplink is required for controller communication and management

AP power policies

Access point power policy details.

Access point power policies are a set of policies that

- define power budget utilization for an AP
- allow management of different interfaces on an AP, and
- support different modes for Cisco Catalyst 9124 APs under insufficient power conditions.

These policies help adapt AP behavior to different input power scenarios (for example, 802.3af, 802.3at, 802.3bt, DC) and ensure reliable operation even under limited power supply.

Additional reference information

- Cisco Catalyst 9124 AXI Series APs and Cisco Catalyst 9124D Series APs support up to two radio interfaces (single 5 GHz).
- Cisco Catalyst 9124 AXE APs support up to three (dual 5 GHz) radio interfaces.
- Operation modes under 8802.3, PoE, and 30 W insufficent power conditions conditions include dual or tri-radio modes.

Use case for AP power policy

These are the use cases of an AP power policy:

- Define a power policy for available power inputs, such as 802.3af, 802.3at, 802.3bt, and DC power.
- Predetermine AP operations for non-802.3bt power situations with tri-radio or quad-radio APs.

Power-save modes

Power-Save Mode details.

A power-save mode is a wireless AP power management feature that

- enables the AP to switch to a low-power state when no clients are associated
- reduces energy consumption by minimizing AP activity during off-peak hours, and
- allows organizations to improve energy efficiency and support environmentally conscious initiatives.

Additional reference information

When power-save mode is enabled, the AP can "fall asleep" after hours in workspaces or other deployment scenarios where client activity is absent, thereby saving power throughout the night.

These are the advantages of power save mode:

- Increases the energy saving per AP: By reducing AP functions during off-peak hours, power-save mode can save an additional 20 percent in energy costs compared to the regular idle mode.
- Enables environmentally conscious purchases: Large enterprises and companies often track environmental performance as a key index. Power-save mode helps support these goals by enabling centralized monitoring of energy efficiency by energy teams.

Use case for power save mode

During after hours in a workspace, when no wireless clients are connected, the AP enters power-save mode and reduces its power usage automatically for the duration of the night.

Release-specific feature support

- In Cisco IOS XE Release 17.15.x, Multi-Link Operation (MLO) is not supported if any radio operates in low power mode.
- However, from Cisco IOS XE Release 17.18.1 onwards, MLO support is enabled even when any of the radios are in low power mode. This feature is applicable on Cisco Wireless 9178 Series Access Points and Cisco Wireless 9176 Series Access Points only.

PoE profiles

PoE Profile details.

A PoE profile is a power allocation framework that

- facilitates negotiation and adaptation of power levels based on AP requirements
- comprises fixed and flexible options for interface configuration, and
- supports policy-based profiles customized for both normal and power-saving operational modes.

Types of PoE profiles

- Fixed PoE profiles: predefined settings per model, typically for full-power mode.
  - Fixed PoE Profiles enable APs to negotiate the power required from the switches they are connected to. The power needed can vary between different AP models.
  - If an AP does not receive the power it requests, it operates within the power budget. It can cause certain interfaces to function under degraded conditions. For instance, radios might operate at 2SS instead of their full 4SS capability.

    - These profiles are used when APs are in normal operation mode (nonpower-save mode). During power-save mode, the configured PoE power policies are applied instead.

- PoE Power Policy: schedule-based profiles for different times of day.
    - PoE Power Policies allow for the configuration of interfaces to specific speeds or settings according to a schedule.
    - These policies can push a defined profile to APs based on timing or calendar settings, such as turning off all but the 2.4-GHz radio and the multigigabit Ethernet at 100 MB at specific times, for example, from 7 p.m. to 7 a.m. on a group of APs located on a second floor.

- Flexible PoE Profiles: customizable profiles for interface-level power control.
    - Flexible PoE Profiles allow you to configure different interfaces with specific parameter values and states, instead of fixed profile rules.
    - In case an AP does not obtain the necessary power, it operates under the power budget, following the guidelines of the flexible PoE profile.

Additional Reference Information

- Operational parameters for each AP interface may be modified based on the hardware specifications, as detailed in the accompanying tables.

AP power draw specifications

AP power draw specifications.

This section provides detailed specifications on the power draw requirements for various AP models to enable accurate planning and deployment in network setups.

Table 33: AP Power Draw Specifications

AP	Power Draw Specification
Cisco Wireless 9172 Series Access Points	See the *Input power requirements* in the *Product specifications* section of Cisco Wireless 9172 Series Access Points Data Sheet
Cisco Wireless 9176 Series Access Points	See the *Input power requirements* in the *Product specifications* section of Cisco Wireless 9176 Series Access Points Data Sheet
Cisco Wireless 9178 Series Access Points	See the *Input power requirements* in the *Power over Ethernet* section of Cisco Wireless 9178 Series Access Points Data Sheet

Table 34: AP Power Draw Specifications: Cisco Catalyst 9115, 9120, 9130 Series APs

Access Points	Power Mode	Consumption @ Power Device	Consumption @ Power Source Equipment	Feature Mode						
		AP	Worst-Case Cable	Radio 1	Radio 2	Radio 3	Ethernet	USB	Module	PoE-Out
Cisco Catalyst 9115AXI Access Points	.3af	13.0	15.4	2X2	2X2	–	1G	N	–	–
	.3at	16.0	18.9	4X4	4X4	–	2.5G	N	–	–
	.3at	20.4	24.1	4X4	4X4	–	2.5G	Y(3.75W)	–	–
Cisco Catalyst 9115AXE Access Points	.3af	13.0	15.4	2X2	2X2	–	1G	N	–	–
	.3at	17.0	20.1	4X4	4X4	–	2.5G	N	–	–
	.3at	21.4	25.3	4X4	4X4	–	2.5G	Y(3.75W)	–	–
Cisco Catalyst 9120AXI/E Access Points	.3af	13.8	15.4	1X1	1X1	Enabled	1G	N	–	–
	.3at	20.5	23.2	4X4	4X4	Enabled	2.5G	N	–	–
	.3at	25.5	30.0	4X4	4X4	Enabled	2.5G	Y(4.5W)	–	–
Cisco Catalyst 9130AXI/E Access Points	.3af	13.8	15.4	1X1	1X1	Enabled	1G	N	–	–
	.3at	25.5	30.0	8X8	4X4	Enabled	5G	N	–	–
	.3at	25.5	30.0	Primary 4X4 Secondary Off	4X4	Enabled	5G	Y(4.5W)	–	–
	.3at	25.5	30.0	Primary 4X4 Secondary 4X4	Disabled	Enabled	5G	Y(4.5W)	–	–
	.3bt	30.5	33.3	8X8	4X4	Enabled	5G	Y(4.5W)	–	–

Table 35: AP Power Draw Specifications: Cisco Catalyst 9136 Series APs

Access Points	Power Mode	Consumption @ Power Device	Consumption @ Power Source Equipment	Feature Mode								
		at AP	With 2m Cable	5G Radio	2G Radio	6G Radio	AUX Radio	Mgig0	Mgig1	USB	Module	PoE-Out
Cisco Catalyst 9136 Series Access Points	.3af - Fixed	13.9	15.4	Disabled	Disabled	Disabled	Enabled	1G	Disabled	Disabled	–	–
	.3at - Fixed	24.0	27.90	Primary - 4X4 Secondary - Disabled	2X2	2X2	Enabled	2.5G	2.5G (hitless failover standby)	Disabled	–	–
	.3bt - Fixed	43.4	54.81	8X8 or Dual 4X4	4X4	4X4	Enabled	5G	5G	Y(9W)	–	–
	.3bt - PoE Policy 1	37.3	41.63	8X8 or Dual 4X4	4X4	4X4	Enabled	5G	5G	Disabled	–	–

Table 36: AP Power Draw Specifications: Cisco Catalyst 9166 Series APs

Access Points	Power Mode	Consumption @ Power Device	Consumption @ Power Source Equipment	Feature Mode								
		at AP	With 2m Cable	5G Radio	2G Radio	6G Radio	AUX Radio	Mgig0	Mgig1	USB	Module	PoE-Out
Cisco Catalyst 9166 Series Access Points	.3af - Fixed	13.9	15.4	Disabled	Disabled	Disabled	Enabled	1G	–	Disabled	–	–
	.3at - Fixed Policy	25.5	30.0	4X4	4X4	4X4	Enabled	5G	–	Disabled	–	–
	.3bt - Fixed	30.5	32.8	4X4	4X4	4X4	Enabled	5G	–	Y (4.5 W)	–	–
	DC Jack - Fixed	30.5	–	4X4	4X4	4X4	Enabled	5G	–	Y (4.5 W)	–	–

Table 37: AP Power Draw Specifications: Cisco Catalyst 9164 Series APs

Access Points	Power Mode	Power Consumption @Power Sourcing Device	Power Consumption @Power Source Equipment	Feature Mode								
		at AP	With 2m Cable	5G Radio	2G Radio	6G Radio	AUX Radio	Mgig0	Mgig1	USB	Module	PoE-Out
Cisco Catalyst 9164 Series Access Points	.3af - Fixed	13.9	15.4	Disabled	Disabled	Disabled	Enabled	1G	–	Disabled	–	–
	.3at - Fixed	25.5	30.0	4X4	2X2	4X4	Enabled	2.5G	–	Disabled	–	–
	.3bt - Fixed	30.1	32.8	4X4	2X2	4X4	Enabled	2.5G	–	Y (4.5 W)	–	–
	DC Jack - Fixed	30.1	–	4X4	2X2	4X4	Enabled	2.5G	–	Y (4.5 W)	–	–

Table 38: AP Power Draw Specifications: Cisco Catalyst 9162 Series APs

Access Points	Power Mode	Power Consumption @Power Sourcing Device	Power Consumption @Power Source Equipment	Feature Mode								
		at AP	With 2m Cable	5G Radio	2G Radio	6G Radio	AUX Radio	Mgig0	Mgig1	USB	Module	PoE-Out
Cisco Catalyst 9162 Series Access Points	.3af	13.3	14.32	1X1	Disabled	1X1	Enabled	1G	–	Disabled	–	–
	.3at	20.1	22.67	2X2	2X2	2X2	Enabled	2.5G	–	Disabled	–	–
	.3at	25.5	30	2X2	2X2	2X2	Enabled	2.5G	–	Y (4.5W)	–	–

Table 39: AP Power Draw Specifications: Cisco Catalyst 9124 Series APs

Access Points	Power Mode	Compliance @ Power Device	Compliance @ Power Source Equipment	Feature Mode										
		at AP	With Cable	Radio 0	R0 dBm Per Path	Radio 1	R1 dBm Per Path	Radio 2	R02 dBm Per Path	AUX Radio	Ethernet Mgig	SFP Module	GbE PHY	PoE-Out
				2.4 GHz Radio		5 GHz Primary Radio		5 GHz Secondary Radio		Chirwave				
Cisco Catalyst 9124 AXI	.3af	13.8	15.4	Disabled	–	Disabled	–	NA	NA	Enabled	1G	N	N	N
	.3at	25.5	30	2X2	23	2X2	23	NA	NA	Enabled	1G	N	Y	N
Cisco Catalyst 9124 AXD	.3bt / UPOE / DC	33.6	39.5	4X4	24	4X4	24	NA	NA	Enabled	2.5G	Y	Y	N
Cisco Catalyst 9124 AXE Dual Radio Mode	.3bt / UPOE / DC	51	60	4X4	24	4X4	24	NA	NA	Enabled	2.5G	Y	Y	Y
Cisco Catalyst 9124 AXI	.3at	25.5	30	Shutdown	–	4X4	24	Shutdown	–	Enabled	2.5G	Y	N	N
Cisco Catalyst 9124 AXD														
Single 5G Radio														

Access Points	PoE Mode	Compliant @Power Device	Compliant @Power Source Ethernet	Feature Mode										
		at AP	Wall Cable	Radio 0	R0 dBm Per Path	Radio 1	R1 dBm Per Path	Radio 2	R02 dBm Per Path	AUX Radio	Ethernet Mgig	SFP Module	GbE PHY	PoE-Out
				2.4 GHz Radio		5 GHz Primary Radio		5 GHz Secondary Radio		Chwave				
Cisco Catalyst 9124 AXE TriRadio Mode	.3af	13.8	15.4	Disabled	–	Disabled	–	Disabled	–	Enabled	1G	N	N	N
	.3at	25.5	30	Disabled	–	Disabled	–	Disabled	–	Enabled	1G	N	N	N
	.3bt / UPOE / DC	33.6	39.5	2X2	24	2X2	24	2X2	24	Enabled	2.5G	Y	Y	N
	.3bt / UPOE / DC	51	60	2X2	24	2X2	24	2X2	24	Enabled	2.5G	Y	Y	Y
Cisco Catalyst 9124 AXE Dual 5G Radio	.3at	25.5	30	Shutdown	–	2X2	23	2X2	23	Enabled	1G	N	Y	N

Client wakeup threshold

Wakeup threshold for access point power save mode.

A wakeup threshold is a power management feature in access points that

- defines the minimum number of client associations required to exit power-save mode
- is used to re-enable radios and interfaces when demand returns, and
- helps optimize energy efficiency in low-use environments.

Load-based wakeup threshold for AP power save mode

When the AP applies a power profile associated with an active calendar, the number of connected clients is compared to the wakeup threshold:

- If the number of connected clients reaches or exceeds the wakeup threshold, the AP exits power save mode and becomes active to ensure adequate performance.
- If the number of connected clients is less than the wakeup threshold, the AP either shuts down the interface or reduces interface speed to save power, maintaining energy efficiency.

This feature dynamically adjusts the AP's operation based on client demand, balancing performance and energy savings.

> **Note**
>
> Spatial stream changes are inhibited on any radio interface with connected clients, except when the calendar profile expires.

Smart energy scenarios

Access point power save scenarios.

A smart energy AP scenario is a situation where

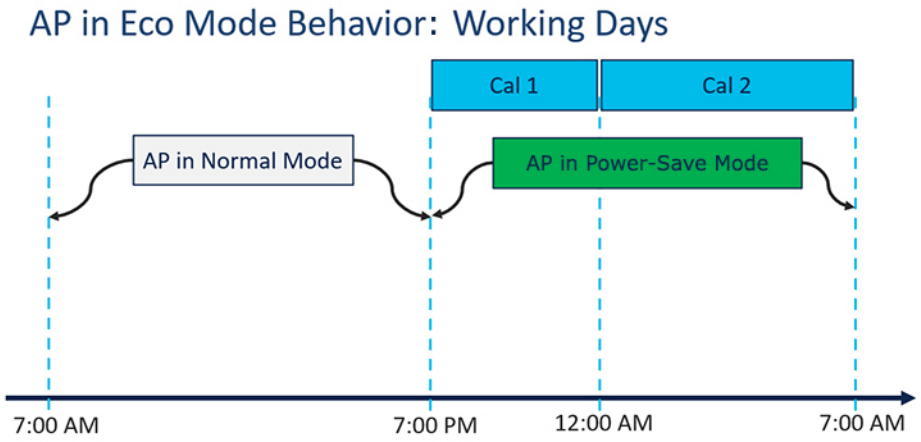
- APs implement energy-saving modes
- adjust power modes to optimize energy efficiency, and
- apply PoE power policies based on calendar profiles.

The power Save feature helps APs enter a power-save mode or low-power mode by associating a calendar with the corresponding power profile. This feature is aimed at conserving energy during periods with lower activity, such as after hours.

Examples

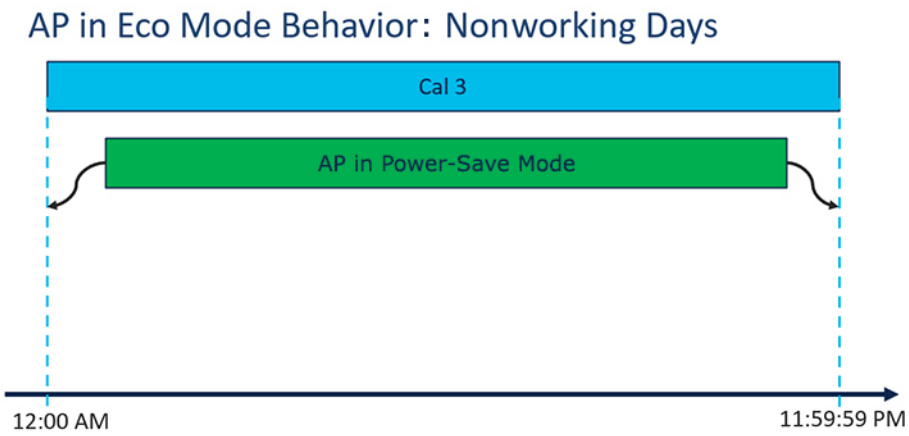
- AP in eco mode behavior on working days: On working days, from 7:00 a.m. to 7:00 p.m, the AP functions in normal mode or fixed mode when the maximum number of clients connect to the AP. From 7:00 p.m. to 12:00 midnight., the **Cal1** calendar profile timer starts to put the AP in the power-save mode. Likewise, the **Cal2** calendar profile timer starts, and extends the power-save mode from 12:00 a.m. to 7:00 a.m. Again, at 7:00 a.m., the AP goes into normal mode.

  Figure 5: AP in Eco Mode Behavior: Working Days



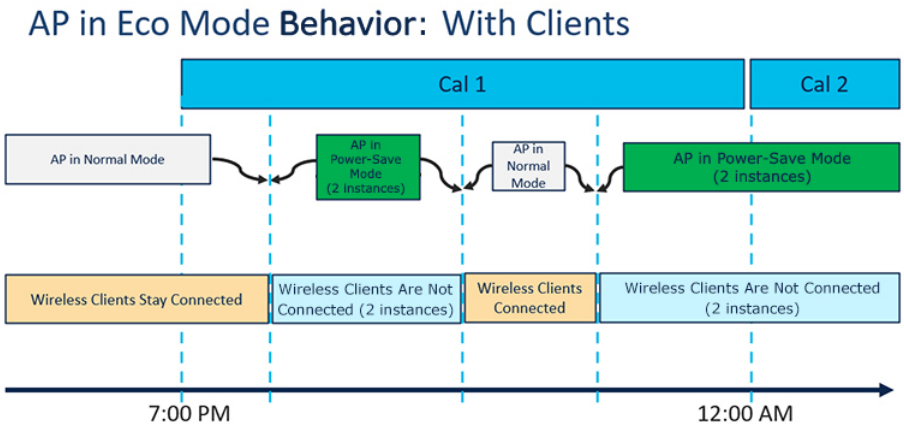
- AP in eco mode behavior on nonworking days: On nonworking days, the AP enters power-saving mode from 12:00 a.m. to 11:59:59 p.m. The **Cal3** calendar profile is applied here. This profile defines the timer for the power-save mode. This means there are no clients connected to the AP, and the AP is asleep.

  Figure 6: AP in Eco Mode Behavior: Nonworking Days



- AP behavior when clients connect: When clients are connected to the AP, the AP switches to the normal mode based on client connections. For example, in the calendar profile **Cal1**, the AP is in normal mode, because wireless clients are connected to the AP. At 8:00 p.m., clients dissociate from the AP, and the AP goes into power-save mode. When clients enter the AP coverage area at 9:30 p.m., the AP automatically switches from power-save mode to normal mode of operation.

Figure 7: AP in Eco Mode Behavior: With Clients



How AP power save policies work

Summary

Access point power save policies dynamically manage power states using profiles and schedules. The key components involved in the process are:

- Power profile: Defines what interfaces or features are disabled or derated.
- Calendar profile: Specifies the timing to activate power profiles.
- Client threshold: Defines when to override calendar and re-enable services.

Workflow

The process involves these following stages:

1. Admin defines and configures power and calendar profiles.
2. Profiles are mapped in the AP join profile.

3. AP follows calendar-based transitions between power-save and normal mode.
4. Client presence can override power-save mode if above threshold.

Result

The AP maintains optimal power usage while ensuring service availability when clients connect.

Configure power policy profile (GUI)

Configure power policy profile using GUI.

Use the GUI to configure a power policy profile, enabling you to manage power settings for network components effectively.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Power Profile**.
2. Click **Add**.
   The **Add Power Profile** window is displayed.
3. Enter a name and description for your power profile. Ensure the name contains only ASCII characters, up to 128 characters, without leading or trailing spaces.
4. Click **Add** to add rules for the power profile.
5. Enter a unique sequence number in the **Sequence number** field to designate the priority for power disablement of the component. A sequence number of zero indicates the component is disabled first.
6. From the **Interface** and **Interface ID** drop-down list, choose interface and interface ID to designate to the component for which the power derating rule applies.
7. From the **Parameter** and **Parameter value** drop-down list, choose the values depending on the interface you have previously chosen.

   For example, if you chose **Ethernet** as an interface, you can further customize the rule for the interface by choosing the associated speed. This rule ensures that the AP disables power for the Ethernet interface that is operating at a higher speed, and thereby consuming more power.
8. Click the check mark to save and then click **Apply to Device**.

After configuration, you can effectively manage power for network components.

Configure a power policy profile (CLI)

Configure a power policy profile using commands.

Set up a power policy profile to manage AP energy by configuring Ethernet and USB components through the CLI.

Before you begin

Make sure you activate at least one radio interface before configuring a power policy profile.

Procedure

1. Enter global configuration mode.

   Example

```
Device# configure terminal
```

2. Configure the power policy profile.

   Example

   ```
   Device(config)# wireless profile power power-profile-name
   ```

3. Configure the power policy for Ethernet.

   Example

   ```
   Device(config-wireless-power-profile)# 10 ethernet gigabitethernet1 speed
   1000mbps
   ```

   sequence-number: You can derate the AP based on the sequence number you enter. The same combination of interface identifiers and parameter values does not appear in another sequence number. The same interface with the same parameter can appear multiple times with different parameter values, however, the parameter value that yields the lowest power consumption is the one that gets selected, irrespective of the sequence number if there is active calendar.

   Note

   - The Ethernet interface is used to join the controller. The uplink interface is not disabled even if it is defined in the power policy.
   - Ethernet speed configuration is not operational in Cisco IOS XE 17.8.1 and later releases.

4. Configure the power policy for USB.

   Example

   ```
   Device(config-wireless-power-profile)# 60 usb 0 state disable
   ```

The power policy profile is successfully configured, allowing efficient management of power resources through defined configurations for Ethernet and USB interfaces.

Configure a calendar profile (GUI)

Configure a calendar profile using the GUI.

Set up calendar profiles to define daily, weekly, or monthly recurrence schedules applied to configurations like power management on APs.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Calendar**.
2. Click **Add**.

   The **Add Calendar Profile** window is displayed.

3. Enter a name for the calendar profile. Names must include ASCII characters up to 32 in length, with no leading or trailing spaces.
4. From the **Recurrence** drop-down list, choose the schedule for which you want to create a profile.
5. Select the **Start Time** and the **End Time** for the recurrence schedule.

> **Note**
>
> - For daily recurrences, you can select the start time and end time. For example, if you want the AP to derate the power on certain interfaces between 7 p.m. to 7 a.m. daily, or if you want the controller to not allow any clients to be associated during this period, you can set up this daily recurrence schedule.
>
> To cover this timespan, you must create two calendar profiles, one for 7 p.m. till 23:59:59, and another one from midnight to 7 a.m. of the next calendar day, and map it to the same power profile. Assign it to the AP Join profile after completing this step.
> - For weekly recurrences, select the specific days of the week along with the start and end time.
> - For monthly recurrence, select the specific days of the month along with the start and end time.

6. Click **Apply** to save the configuration.

The calendar profile is applied to manage schedules as defined to influence device behavior based on the recurrence schedule.

Configure a calendar profile (CLI)

Configure a calendar profile using commands.

Configure a calendar profile to automatically schedule device operations based on defined timing parameters.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a calendar profile. Enter the calendar profile configuration mode.

   Example

   ```
   Device# wireless profile calendar-profile name <i>ap_power_calendar</i>
   ```

   Here, **name** refers to the name of the calendar profile.

3. Configure daily recurrence for daily profile.

   Example

   ```
   Device(config-calendar-profile)# recurrence daily
   ```

4. Configure the start time and end time for the calendar profile.

   Example

   ```
   Device(config-calendar-profile)# start 16:00:00 end 20:00:00
   ```

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config-calendar-profile)# end
   ```

The device uses the specified calendar profile to manage operations according to the configured timings.

Configure a power policy in an access point join profile (GUI)

Configure a power policy in an AP join profile using GUI.

Configure a power policy using a power profile or a mapped configuration of a power profile with a calendar profile in an AP join profile.

During an AP join session, you can use a power profile or a mapped configuration of a power profile and a calendar profile. Map up to five combination profiles (calendar and power) per AP profile.

Before you begin

Ensure that the power profile and calendar profile are created and displayed in the respective drop-down lists in the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.

   The **Add AP Join Profile** window is displayed.

2. Click the **AP** tab.
3. Under the AP tab, click the **Power Management** tab.
4. From the **Regular Power Profile** drop-down list, choose the power profile.

   The AP derates power based on the priority list that you configured.

   > **Note**
   >
   > If you want the AP to apply the power profile configuration during a specific time period, choose the **Calendar Profile** and map it to the power profile from the drop-down list.

5. Click the check mark to associate the mapping.
6. Click **Apply to Device** to save the configuration.

The AP adjusts its power settings based on designated profiles and according to set priorities and calendar schedules.

Map a power profile under an AP profile (CLI)

Map a Power Profile Under an AP Profile using commands.

Configure a power profile mapping in an AP profile using CLI commands.

Before you begin

Ensure that you have defined a calendar profile in the wireless profile, before you map the calendar profile to an AP join profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)#  ap profile  ap-profile-name
   ```

3. Configure the AP power profile.

   Example

   ```
   Device(config-ap-profile)#  power-profile  power-profile-name
   ```

   This power profile is used during non-calendar hours to meet the power budget provided by the switch connected to the AP.

4. Map a calendar profile to the AP profile. Enter the AP profile calendar configuration mode.

   Example

   ```
   Device(config-ap-profile)#  calendar-profile  ap-calendar-profile
   ```

5. Map a specific power profile to a specific calendar profile.

   Example

   ```
   Device(config-ap-profile-calendar)# action power-saving-mode
   power-profile power-profile1
   ```

   Maps the power-saving mode action for the calendar profile. Use the **no** form of this command to disable the command.

> **Note**
>
> You can have more than one mapping of calendar profile to power profile.

The selected power profile is now mapped to the calendar profile within the specified AP profile.

Configure client wakeup threshold (CLI)

Configure client wakeup threshold using commands.

Configure the client wakeup threshold on devices to manage power save mode effectively.

This task involves setting a threshold for clients that influences the power save mode of APs.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the power policy profile.

   Example

   ```
   Device(config)# wireless profile power power-profile1
   ```

3. Configure the client threshold below which the AP can stay in the power save mode.

   Example

   ```
   Device(config-wireless-power-profile)# power-save-client-threshold 5
   ```

   The valid range is between one and 32 clients. The default value is one.

The power save mode configuration is completed successfully for the specified client threshold.

Configure PoE-Out Interface in Power Profile using GUI

Configure PoE-Out Interface in Power Profile through the GUI.

Configure the PoE-Out interface in a power profile through the GUI. This setup is necessary to ensure proper power management of connected devices through an AP.

Before you begin

Enable the RLAN port for the AP.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Power Profile**.
2. Click **Add**.

   The **Add Power Profile** window is displayed.
3. Enter a name and description for the power profile. The name can contain up to 128 ASCII characters, without leading or trailing spaces.
4. Optional: From the **Power Save Client Threshold** counter, select a value to set a limit to the number of client associations with the AP. The default value is 1. The range is between 1 and 32.
5. Click **Add** to create a rule for the PoE-Out interface.

   The **Rule** section is displayed in the window.
6. In the **Rule** section, complete the following steps:
   a) In the **Sequence number** field, enter a unique sequence number to assign the priority in which power should be disabled for the component. A sequence number of 0 indicates that the component should be disabled first.
   b) From the **Interface** drop-down list, choose **Ethernet** as the interface.
   c) From the **Interface ID** drop-down list, choose one of the following interfaces: **LAN1**, **LAN2**, or **LAN3**.
   d) From the **Parameter** drop-down list, choose **POE-out**.

      An AP uses a POE-out port to provide power to another device, like a camera. This selection ensures that the power usage of the port is reduced or shut down at the specified sequence.
7. Click **Apply to Device**.

The power profile is configured, setting the PoE-Out interface according to specified parameters, effectively managing power to connected devices.

Configure PoE-out interface in the power profile

Configure PoE-out interface in power profile using commands.

Configure PoE-out interfaces in your network within a specified power profile using CLI commands.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Configure the power policy profile. Enter the wireless power profile configuration mode.

   Example

   ```
   Device(config)# wireless profile power poe-out-power-profile
   ```
3. Disable the PoE-out state.

Example

```
Device(config-wireless-power-profile)# 1 ethernet LAN1 poe-out disable
```

The interface disables the PoE-out state according to the specified power profile settings for optimized power management.

Configuration example of power profile

Configuration example of power profile.

The example shows how to define a power save policy:

```
wireless profile power power-save
            10 radio 5ghz state shutdown
            20 radio secondary-5ghz state shutdown
            30 radio 6ghz state shutdown
            40 usb 0 state disable
```

The example shows how to define a calendar profile:

```
wireless profile calender-profile name eve-to-midnight
            recurrence daily
            start 19:00:00 end 23:59:59
wireless profile calender-profile name midnight-to-morning
            recurrence daily
            start 00:00:00 end 07:00:00
wireless profile calender-profile name weekends
            recurrence weekly
            day Saturday
            day Sunday
            start 00:00:00 end 23:59:59
```

The example shows how to define an AP join profile and map a calendar profile to a power profile:

```
ap profile wireless-prof-site1
                 calendar-profile eve-to-midnight
                  action power-saving-mode power-profile power-save
                 calendar-profile midnight-to-morning
                  action power-saving-mode power-profile power-save
                 calendar-profile weekends
                  action power-saving-mode power-profile power-save
```

Verify access point power policy (GUI)

Verify the AP Power Policy using GUI.

The purpose of this task is to verify the AP Power Policy applied on the Cisco Catalyst 9136 series using the GUI.

To verify the applied configuration on the GUI, use these steps:

Procedure

1. Choose **Monitoring** > **AP Statistics**.
2. Click a Cisco Catalyst 9136 series AP from the list of APs.

   The **General** window is displayed.

3. Click the **Power** tab.

   The **Power Operational Status** and the **AP Fixed Power Policy** details are displayed.

4. Click **OK**.

These steps help verify whether the power policy settings are correctly applied to the AP.

Verify the access point fixed power policy details

Ensure that the access points are operating with the correct power settings by verifying and updating the AP fixed power policy details, which helps maintain optimal network performance and coverage.

To verify the AP fixed power policy details from the list of configured APs, use these steps:

Procedure

---

1. Choose **Configuration** > **Access Points**.
2. Click a Cisco Catalyst 9136 series AP from the list of APs.
   The **Edit AP** window is displayed.
3. Click the **Interfaces** tab.
   The **AP Fixed Power Policy** details are displayed.
4. Click **Update & Apply**.

The AP fixed power policy details will be displayed.

Verify the access point power profile

Verify the AP Power Profile.

To view the calendar profile and its mapping, use this command:

```
Device# show ap profile name default-ap-profile detailed
AP Profile Name                      : default-ap-profile
Description                          : default ap profile
Power profile name                   : power_prof_day
AP packet capture profile            : Not Configured
AP trace profile                     : Not Configured
Mesh profile name                    : default-mesh-profile
Power profile name                   : Not Configured
Calendar Profile
    Profile Name                    : cal47
    Power saving mode profile name : pow_da
    ------------------------------------------------
    Profile Name                    : cal48
    Power saving mode profile name : pow23
    ------------------------------------------------
```

To view the operational details of the AP, use this command:

```
Device# show ap name cisco-ap power-profile summary
AP power derate Capability      : Capable

Power saving mode
Power saving mode profile        : pow2
Associated calendar profile      : cal1
```

```
AP power profile status         : Insufficient De-rating

Interface    Interface-ID       Parameter        Parameter value    Status
-------------------------------------------------------------------------------
Radio        5 GHz              State            DISABLED           Success

Radio        6 GHz              State            DISABLED           Not
Applicable
Ethernet     LAN1               State            DISABLED           Not
Applicable
Radio        2.4 GHz            State            DISABLED           Success

Ethernet     Gig0               Speed            5000 MBPS          Fixed
Policy
```

AP power derate capability is displayed in the output as **Capable** only for those APs that support power policy. For the other APs, it is displayed as **Not Capable**.

In the **show ap name cisco-ap power-profile summary** output, in the power saving mode, the status of the interface configured in the power profile (for example, **pow2**) is applied on the AP, and the AP sends the details (that are displayed in the show command) such as, the name of the power saving profile and the associated calendar profile.

The table that is displayed shows the interfaces and the parameter status of the power saving profile. The AP sends the information as to which of the interfaces are disabled. For example, if the AP does not have a 6-GHz radio interface, the **Status** is displayed as **Not Applicable**. If the interfaces are applied without any errors, then **Success** is displayed.

Note

When the AP uses the fixed power policy, due to inactive calendar or client connectivity, the interfaces are not displayed in the power profile summary if their status is UP on the AP.

Verify radio spatial streams

Verify radio spatial streams.

To view the configuration and operational details of radio spatial stream rules in the power profile, run these commands:

```
Device# show wireless profile power detailed wireless_pow_profile_name
Power profile name              : wireless_pow_profile_name
-----------------------------------------------
Description                     :
.
.
Seq No       Interface    Interface-id       Parameter        Parameter value
-------------------------------------------------------------------------------
100          Radio        6 GHz              Spatial Stream   2 x 2

200          Radio        5 GHz              Spatial Stream   8 x 8

400          USB          USB0               State            DISABLED

500          Ethernet     Gig0               Speed            100 MBPS

600          Radio        6 GHz              State            DISABLED

700          Radio        2.4 GHz            State            DISABLED
```

```
900          Radio        5 GHz              State            DISABLED
```

```
Device# show ap name cisco-ap-name power-profile summary
AP power derate Capability      : Capable

AP fixed power policy
---------------------

Interface    Interface-ID       Parameter        Parameter value    Status
-------------------------------------------------------------------------------
Ethernet     Gig0               Speed            5000 MBPS          Fixed
Policy
Radio        2.4 GHz            Spatial Stream   4 x 4              Fixed
Policy
Radio        5 GHz              Spatial Stream   8 x 8              Fixed
Policy
Radio        Sec 5 GHz          Spatial Stream   4 x 4              Fixed
Policy
USB          USB0               State            DISABLED           Fixed
Policy
```

Verify client threshold

Verify client threshold.

To view the client threshold details, run this command:

```
Device# show wireless profile power detailed profile1
Power profile name             : profile1
-----------------------------------------------
Description                    : Power profile 1
Power save client threshold    : 5
Seq No       Interface    Interface-id       Parameter        Parameter value
-------------------------------------------------------------------------------
0             Radio        6 GHz             State              DISABLED
1             Radio        5 GHz             Spatial Stream     1 x 1
2             Radio        2.4 GHz           Spatial Stream     2 x 2
3             USB          USB0              State              DISABLED
```

Verify PoE-out details

Verify PoE-out details.

To view the PoE-out details in the wireless power profile, run these commands:

```
Device# show wireless profile power detailed poe-out_profile_name
Power profile name             : poe-out_profile_name
-----------------------------------------------
Description                    : profile-description
Seq No       Interface    Interface-id       Parameter        Parameter value
-------------------------------------------------------------------------------
10 ...
20           Ethernet     LAN 1              POE_OUT          DISABLED
30 ...
```

```
Device# show ap name Cisco-Ap1 power-profile summary
AP power derate Capability      : Capable
```

```
AP fixed power policy
---------------------

Interface    Interface-ID       Parameter        Parameter value    Status
-------------------------------------------------------------------------------
...
Ethernet     LAN 1              POE_OUT          DISABLED           Fixed
Policy
...
```

31 Real-Time Access Points Statistics

Topics:

- Access point real-time statistics

Access point real-time statistics

Introduces access point real-time statistics, covering SNMP traps, monitoring restrictions, and configuration procedures via GUI and CLI. Details radio monitoring, verification commands, and guides AP performance management for optimal wireless network operations.

An access point real-time statistic is a tool that

- allows tracking of CPU utilization and memory usage of an AP,
- monitors the health of an AP, and
- generates real-time statistics from an AP.

Feature history

Release	Feature	Feature information
Cisco IOS XE 17.7.1	Real time access point statistics	The system implements this feature with AP threshold values between 0 and 50. From Cisco IOS XE Cupertino 17.7.1 onwards, you can reset radios based on the AP's real-time statistics. If Tx or Rx counters do not increment during a sampling period, a radio reset is triggered. The following commands are introduced: • **show wireless stats ap join summary** • **show wireless stats ap history**
Cisco IOS XE 17.5.1	Real time access point statistics	Starting from Cisco IOS XE 17.5.1, you can monitor both CPU utilization and memory usage to assess the health of an AP through generated real-time statistics.

Supporting reference information

SNMP traps are sent out when the threshold is crossed. The sampling period and statistics interval can be configured using SNMP, YANG, and CLI.

From Cisco IOS XE 17.7.1 release onwards, for radio monitoring, you can reset the radios based on the statistics sent by the AP for a sampling period. When you configure the radios in the controller, if there is no increment in the Tx or Rx statistics when the radio is up, then the radio reset is triggered.

SNMP traps and intervals for APs

AP real-time statistics support these capabilities:

- SNMP Traps: Triggers notifications when CPU or memory usage crosses configured thresholds for APs and controllers.

- Sampling Period and Statistics Interval: Can be configured using SNMP, YANG, and CLI to control how often statistics are collected and reported.

Restrictions for AP radio monitoring statistics

AP radio monitoring statistics restrictions.

You cannot reset the radio firmware from the controller. The controller will shut and then restart the radio if the Rx or Tx count does not increase for a radio slot within a specified period.

Configure access point real time statistics (GUI)

Configure access point real time statistics using GUI.

This task helps you configure real-time statistics for access points using a graphical user interface, allowing you to effectively monitor and manage AP behavior.

Procedure

1. Choose **Configuration > Tags & Profiles > AP Join**.
2. Click **Add**. The **Add AP Join Profile** page is displayed.
3. Under the **AP** tab, click the **AP Statistics** tab.
4. In the **System Monitoring** section:
   - Enable **Monitor Real Time Statistics** to get calculated statistics and alarms of the AP.
   - To receive an alarm when the upper threshold is surpassed for parameters such as CPU utilization and memory, enable **Trigger Alarm for AP**.
   - Enter the threshold percentage for CPU and memory usage in the **CPU Threshold to Trigger Alarm** field and **Memory Threshold to Trigger Alarm** fields, respectively. The valid range is between 0 and 50. An SNMP trap is sent out when this threshold is crossed.
   - In the **Interval to Hold Alarm** field, enter the time for which the alarm is held before it gets triggered. The range is between 0 and 3600 seconds.
   - In the **Trap Retransmission Time** field, enter the time between retransmissions of the alarm. The range is between 0 and 65,535 seconds.
   - To define how often data should be collected from the AP, enter a value in the **Sampling Interval** field. The range is between 720 and 3600 seconds.
   - To define the interval at which AP statistics are to be calculated, enter a value in the **Statistics Interval** field. The range is between 2 and 900 seconds.
   - To automatically reload the AP when there is high CPU and memory usage in the defined sampling interval, select the **Reload the AP** check box.
5. Under the **Radio Monitoring** section:
   - Select the **Monitoring of AP Radio stuck** check box to verify that the Tx and Rx statistics of the AP are updated each time the payloads are coming in from the AP to the controller.
   - To generate an alarm for the radio of the AP when there is no increment in the Tx and RX statistics for the payloads, select the **Alarms for AP Radio stuck** check box.
   - Select the **Reset the stuck AP Radio** check box to recover the radio from the bad state. A radio admin state payload will be sent from the controller to toggle the radio and the radio will be shut when there is no increment in the Tx and Rx statistics.

- To define how often data should be collected from the radio, enter a value in the **Sampling Interval** field. The valid range is between 720 and 3600 seconds.

6. Click **Apply to Device** to save the configuration.

The AP monitors real-time statistics and triggers alarms based on defined thresholds to aid you in efficiently managing AP resources.

Configure access point real-time statistics (CLI)

Configure real-time access point statistics using commands.

This task provides instructions on how to configure real-time statistics for an access point using CLI commands.

This task is intended for network administrators who need to monitor and configure various system parameters on a Cisco access point, particularly real-time statistics related to performance monitoring.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the AP profile. The default AP join profile name is *default-ap-profile*.

   Example

   ```
   Device(config)# ap profile doc-test
   ```

3. Optional: Configure the statistics timer.

   Example

   ```
   Device(config-ap-profile)# stats-timer 60
   ```

   This step configures the interval for the statistics reports coming from the AP, with values ranging from 0 to 65535 seconds.

4. Optional: Enable monitoring for AP system.

   Example

   ```
   Device(config-ap-profile)# statistics ap-system-monitoring enable
   ```

5. Enable alarms for AP system monitoring.

   Example

   ```
   Device(config-ap-profile)# statistics ap-system-monitoring alarm-enable
   ```

6. Define the hold time for system monitoring alarms on the AP.

Example

```
Device(config-ap-profile)# statistics ap-system-monitoring alarm-hold-time
400
```

The values range from 0 to 3600 seconds.

7. Define the retransmission interval for the trap alarms.

Example

```
Device(config-ap-profile)# statistics ap-system-monitoring
alarm-retransmit-time 100
```

Set the interval, with values from 0 to 65535 seconds, for retransmitting trap alarms when conditions persist.

8. Define the CPU usage threshold for the AP to trigger alarms.

Example

```
Device(config-ap-profile)# statistics ap-system-monitoring cpu-threshold 90
```

> **Note**
>
> From Cisco IOS XE 17.7.1 release onwards, the threshold value for triggering alarms is from 0 to 100%.

9. Define the memory usage threshold on the AP to trigger alarms.

Example

```
Device(config-ap-profile)# statistics ap-system-monitoring mem-threshold 90
```

The percentage threshold for memory usage on the AP to trigger is from 0 to 100%.

> **Note**
>
> From Cisco IOS XE 17.7.1 release onwards, the memory usage threshold to trigger alarms ranges from 0 to 100%.

10. Optional: Configure the statistics sampling interval.

Example

```
Device(config-ap-profile)# statistics ap-system-monitoring sampling-interval
 600
```

The interval can range from 2 to 65535 seconds.

11. Exit from AP profile configuration mode and return to global configuration mode.

Example

```
Device(config-ap-profile)# exit
```

12. Enable sending AP-related trap flags for statistics that exceed set thresholds.

Example

```
Device(config)# trapflags ap ap-stats
```

Traps are sent when statistics exceed the configured thresholds, enabling alerts and automated responses.

The statistics of the access point have been configured, enabling real-time monitoring and alarm triggering for system performance parameters.

```
                Device(config)# ap profile default-policy-profile
                Device(config-ap-profile)# statistics ap-system-monitoring
enable
                Device(config-ap-profile)#statistics ap-system-monitoring
sampling-interval 600
                Device(config-ap-profile)#statistics ap-system-monitoring
alarm-enable
                Device(config-ap-profile)#statistics ap-system-monitoring
alarm-hold-time 3
                Device(config-ap-profile)#statistics ap-system-monitoring
alarm-retransmit-time 10
                Device(config-ap-profile)#statistics ap-system-monitoring
cpu-threshold 90
                Device(config-ap-profile)#statistics ap-system-monitoring
mem-threshold 90
                Device(config)# trapflags ap ap-stats
```

Configure the access point radio monitoring statistics (CLI)

Configure access point radio monitoring statistics using commands.

Learn how to configure monitoring statistics for AP radios to enable stuck radio statistics, alarms, and actions for optimal performance.

These steps help configure AP radio monitoring statistics:

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure an AP profile and enter the AP profile configuration mode.

Example

```
Device(config)# ap profile test1
```

3. Enable the monitoring of AP radio stuck statistics.

   Example

   ```
   (config-ap-profile)#statistic ap-radio-monitoring enable
   ```

4. Enable the alarm for AP radio stuck statistics.

   Example

   ```
   (config-ap-profile)#statistic ap-radio-monitoring alarm-enable
   ```

5. Specify the sampling interval in seconds.

   Example

   ```
   (config-ap-profile)# statistic ap-radio-monitoring action reload-ap interval850
   ```

   The values range from 720 to 3,600 seconds.

6. Generate an alarm and reset the radio if the radio is stuck.

   Example

   ```
   (config-ap-profile)# statistic ap-radio-monitoring action radio-reset
   ```

7. Reload the AP.

   Example

   ```
   (config-ap-profile)# statistic ap-system-monitoring action reload-ap
   ```

The AP radio monitoring statistics help identify stuck radios and automatically take corrective actions to ensure optimal performance.

```
Device(config)# ap profile test1
Device(config-ap-profile)# statistics ap-radio-monitoring enable
Device(config-ap-profile)#statistic ap-radio-monitoring alarm-enable
Device(config-ap-profile)#statistic ap-radio-monitoring sampling-interval 750
Device(config-ap-profile)# statistic ap-radio-monitoring action radio-reset
Device(config-ap-profile)#statistic ap-system-monitoring action reload-ap
```

Monitor access point real-time statistics (GUI)

Monitor AP real-time statistics using GUI.

This task enables network administrators to monitor access point performance and diagnose potential issues by viewing real-time statistics.

Procedure

---

1. Choose **Monitoring** > **Wireless** > **AP Statistics**.

2. Click the **General** tab.
3. Click an AP name. The **General** window is displayed.
4. To view the AP Statistics data, click the **AP Statistics** tab.

   These types of information are displayed:

   - **Memory alarm last send time**: Displays the time of the last memory trap sent.
   - **Memory Alarm Status**: Displays the state of the memory alarm. An alarm can be **ACTIVE**, **INACTIVE**, **INACTIVE_SOAKING**, **ACTIVE_SOAKING**. An alarm is soaked until the configured hold time has passed.
   - **Memory alarm raise time**: Displays the last time the memory alarm was active.
   - **Memory alarm clear time**: Displays the last time the memory alarm was inactive.
   - **Last statistics received**: Displays the time of the last statistics report received from the AP.
   - **Current CPU Usage**: Displays the latest percentage of CPU usage reported.
   - **Average CPU Usage**: Displays the average CPU usage calculated.
   - **Current Memory Usage**: Displays the latest percentage of memory usage reported.
   - **Average Memory Usage**: Displays the average memory usage calculated.
   - **Current window size**: Displays the window size. The window size is calculated by dividing the statistics interval by the sampling interval. The average CPU and memory usage is calculated by the window size.
   - **CPU alarm last send time**: Displays the time of the last CPU trap sent.
   - **CPU Alarm Status**: Displays the state of the CPU alarm. An alarm can be **ACTIVE**, **INACTIVE**, **INACTIVE_SOAKING**, **ACTIVE_SOAKING**. An alarm is soaked until the configured hold time has passed.
   - **CPU alarm raise time**: Displays the last time the CPU alarm was active.
   - **CPU alarm clear time**: Displays the last time the CPU alarm was inactive.

5. Click **OK**.

The current statistics for the selected access point are displayed, allowing for comprehensive monitoring and analysis of AP performance metrics.

Verify access point real-time statistics

Verify access point real-time statistics.

To verify AP real-time statistics, run this command:

```
Device# show ap config general | section AP statistics
!Last Statistics
AP statistics : Enabled
Current CPU usage : 4
Average CPU usage : 49
Current memory usage : 35
Average memory usage : 35
Last statistics received : 03/09/2021 15:25:08
!Statistics Configuration
Current window size : 1
Sampling interval : 30
Statistics interval : 300
AP statistics alarms : Enabled
!Alarm State - Active, Inactive, Inactive_Soaking, Inactive_Soaking
```

```
Memory alarm status : Active
Memory alarm raise time : 03/09/2021 15:24:29
Memory alarm clear time : NA
Memory alarm last send time : 03/09/2021 15:24:59
CPU alarm status : Inactive
CPU alarm raise time : 03/09/2021 15:24:25
CPU alarm clear time : 03/09/2021 15:25:05
CPU alarm last send time : 03/09/2021 15:25:05
!Alarm Configuration
Alarm hold time : 6
Alarm retransmission time : 30
Alarm threshold cpu : 30
Alarm threshold memory : 32
```

To verify that the statistics reporting period is accurate, run this command:

```
Device# show ap config general | i Stats Reporting Period
Stats Reporting Period : 10
```

32 AP Configuration

Topics:

- AP console configurations
- AP audit configuration
- Radio support for specified slot number
- XOR radio support
- Receiver only dual-band radio support
- OFDMA support for 802.11ax APs
- Cisco flexible antenna port
- Environmental sensors in APs
- CAPWAP LAG support
- CAPWAP message aggregation

AP console configurations

Explains how AP console configurations enable or restrict console access to access points and outlines their relation to security modes and configuration methods.

An AP console configuration is a controller-managed setting that

- allows administrators to enable or disable console access for APs from the controller
- operates independently from the FIPS or high-security (CC) mode in certain software versions, and
- can be configured through both the controller GUI and CLI.

Enable the console independently of FIPS mode by configuring the AP join profile. If you enable CC mode, console and SSH access are disabled. Any AP console settings configured in the AP profile are overridden by CC mode.

Feature history

Table 40: Feature history for AP console configurations

Feature Name	Release Information	Feature Description
AP console configurations	Cisco IOS XE 17.9.1	This feature allows you to configure the AP console from the controller. In Cisco IOS XE Cupertino 17.8.x and earlier releases, the AP console could be disabled from the controller, only by enabling the Federal Information Processing Standard (FIPS) mode or the Common Criteria (CC) mode.

Configure the AP console (GUI)

Configure the AP console using GUI.

Enable serial console access for an AP via the management interface.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. In the **Management** tab, go to the **Telnet/SSH Configuration** section. Check the **Serial Console** check box.
3. Set the **Console Speed**. The available options are: **9600**, **115200** and **None**.

   None sets the **no console speed**.
4. Click **Apply to Device**.

Configure the AP console (CLI)

Configure the AP console using commands.

Enable or disable the AP serial console port by configuring the AP profile.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile-name
   ```

3. Enable the AP serial console port.

   Example

   ```
   Device(config-ap-profile)# console
   ```

   Use the **no** form of this command to disable the AP serial console port.

4. Configure the AP console speed.

   Example

   ```
   Device(config-ap-profile)# console speed { 9600 | 115200}
   ```

   The supported speeds are 9600 and 115200 bps.

   Use the **no** form of the command to maintain the current configuration and retain the AP's console speed.

Verify the AP console status

Verify the AP console status.

To verify the AP console status from the controller, run the command:

```
Device# show ap config general | include ap-Name | console
Cisco AP Name : CiscoAP
================================================
Cisco AP Identifier : 6XXX.bXXX.aXXX
Country Code : US
Regulatory Domain Allowed by Country : 802.11bg:-A 802.11a:-AB 802.11 6GHz:-B
AP Country Code : US - United States
AP Regulatory Domain
802.11bg : -A
802.11a : -B
MAC Address : 6XXX.bXXX.0XXX
IP Address Configuration : DHCP
IP Address : 10.30.30.30
IP Netmask : 255.255.255.0
Gateway IP Address : 10.30.30.30
Fallback IP Address Being Used :
Domain :
```

```
Name Server :
CAPWAP Path MTU : 1485
Capwap Active Window Size : 1
Telnet State : Disabled
CPU Type : ARMv8 Processor rev 4 (v8l)
Memory Type : DDR3
Memory Size : 1752064 KB
SSH State : Enabled
Serial Console State : Enabled
```

AP audit configuration

Explains how AP audit configuration enables real-time and periodic detection and reporting of synchronization issues between wireless controllers and access points.

An AP audit configuration is a wireless network management feature that

- detects and reports synchronization issues between the controller and an AP
- provides both real-time and periodic comparison of configuration and operational states, and
- supports automated error reporting using syslog for discrepancies.

In Cisco IOS XE Amsterdam, Release 17.3.1, two methods are implemented to support AP audit configuration.

- Config Checker
- Config Audit

Config Checker audits the application of wireless policies during the AP join phase. Any discrepancies detected at this stage are reported to the controller. This function is built-in and cannot be disabled. When you configure AP attributes such as name, IP address, controller information, tag, mode, radio mode, and radio admin state, the AP parses the CAPWAP payload configuration from the controller. It then reports any detected errors to the controller using the proper code. If a discrepancy is detected, the controller flags errors using the syslog.

Config Audit periodically compares operational states between an AP and the controller after the AP joins and while it remains connected. When discrepancies are found, Config Audit reports them immediately on the controller. You can view a consolidated report at any time. By default, this functionality is disabled. You can configure the periodic auditing interval.

Use the **ap audit-report** command to enable and configure audit report parameters. When you trigger the audit, the AP sends its configurations to the controller. The controller compares these configurations with the current settings. If there is a mismatch, the controller reports the error using syslog.

Feature history

Table 41: Feature history for AP audit configuration

Feature Name	Release Information	Feature Description
AP audit configuration	Cisco IOS XE 17.3.1	This feature enables real-time and periodic detection, reporting, and automated syslog alerting of synchronization discrepancies between the wireless controller and APs.

Restrictions for AP audit configuration

AP audit configuration restriction.

- Config checker alerts are available only through the syslog.
- IOS AP is not supported.
- Audit reports do not synchronize from the active controller to the standby controller. After SSO, you cannot access audit reports until the next reporting interval for already connected APs.
- Audit reports are not available when an AP operates in standalone mode.
- This feature is supported only on APs in FlexConnect mode.

Configure AP audit parameters (CLI)

Configure AP audit parameters using commands.

Enable and customize AP audit reporting to monitor and compare AP operational states against controller expectations.

The AP sends state view details to the controller, and the controller compares it with what it perceives as the AP state. This feature is disabled by default.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable audit reporting.

   Example

   ```
   Device(config)# ap audit-report enable
   ```

3. Configure the AP audit reporting interval.

   Example

   ```
   Device(config)# ap audit-report interval interval 1300
   ```

   The default value for interval is 1440 minutes. The valid range is from 10 to 43200.

Verify AP audit report summary

Verify AP audit report summary.

To verify the AP audit report summary, use the **show ap audit-report summary** command:

```
Device# show ap audit-report summary
WTP Mac                    Radio                   Wlan                 IPv4
Acl          IPv6 Acl       Last Report Time
```

```
-------------------------------------------------------------------------------------------------
1880.90fd.6b40   OUT_OF_SYNC    OUT_OF_SYNC    IN_SYNC        IN_SYNC
01/01/1970 05:30:00 IST
```

Verify AP audit report detail

Verify AP audit report details.

To verify an AP audit report's details, use the **show ap name ap-name audit-report detail** command:

```
Device# show ap name Cisco-AP audit-report detail
Cisco AP Name   : Cisco-AP
================================================
IPV4 ACL Audit Report Status     : IN_SYNC
IPV6 ACL Audit Report Status     : IN_SYNC
Radio Audit Report Status        : IN_SYNC
WLAN Audit Report Status         :
Slot-id  Wlan-id  Vlan           State          SSID           Auth-Type
 Other-Flag
-------------------------------------------------------------------------------------
0        4        IN_SYNC        IN_SYNC        IN_SYNC        IN_SYNC
 IN_SYNC
1        4        IN_SYNC        IN_SYNC        IN_SYNC        IN_SYNC
 IN_SYNC

bh-csr1# show ap name audit-report summary
WTP-Mac          Radio          Wlan           IPv4-Acl       IPv6-Acl
Last-Report-Time
-------------------------------------------------------------------------------------------------
4001.7aca.5140   IN_SYNC        IN_SYNC        IN_SYNC        IN_SYNC
06/22/2020 13:17:39 IST
4001.7aca.5a60   IN_SYNC        IN_SYNC        IN_SYNC        IN_SYNC
06/22/2020 13:18:25 IST
7070.8b23.a1a0   IN_SYNC        IN_SYNC        IN_SYNC        IN_SYNC
06/22/2020 13:18:29 IST
a0f8.49dc.9460   IN_SYNC        IN_SYNC        IN_SYNC        IN_SYNC
06/22/2020 13:16:43 IST
a0f8.49dc.96e0   IN_SYNC        IN_SYNC        IN_SYNC        IN_SYNC
06/22/2020 13:17:55 IST
```

Radio support for specified slot number

Explains which radio modules are supported in each slot number.

Configure 2.4-GHz radio support for the specified slot number (CLI)

Configure 2.4-GHz radio support for the specified slot number.

Enable and configure 2.4-GHz radio features for a specified AP slot using commands.

> **Note**
>
> The term **802.11b radio** or **2.4-GHz radio** are used interchangeably.

Procedure

1. Enter the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Enable Spectrum Intelligence (SI) for the dedicated 2.4-GHz radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 SI
   ```

   For more information, see the *Spectrum Intelligence* section in this guide. In this context, ***0*** refers to the Slot ID.

3. Configure 802.11b antenna hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 antenna {ext-ant-gain
   antenna_gain_value | selection [ internal | external ]}
   ```

   - **ext-ant-gain** : configures the 802.11b external antenna gain. *antenna_gain_value* : the external antenna gain value, specified in multiples of 0.5 dBi units. The valid range is from zero to 40, with a maximum gain of 20 dBi.
   - **selection** : configures the 802.11b antenna to use either the internal or external option.

   > **Note**
   >
   > - For APs that support self-identifying antennas (SIA), the gain depends on the antenna and not the AP model. The AP automatically learns the gain, so no controller configuration is necessary.
   > - For APs that do not support SIA, the AP sends the antenna gain in the configuration payload. The default antenna gain depends on the AP model.
   > - Cisco Catalyst 9120E and 9130E APs support self-identifying antennas (SIA). Cisco Catalyst 9115E APs do not support SIA antennas. Although Cisco Catalyst 9115E APs work with SIA antennas, they do not automatically detect SIA antennas or apply the correct external gain.

4. Configure beamforming for the 2.4-GHz radio hosted on slot 0 for a specific AP.

Example

```
Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 beamforming
```

5. Configure advanced 802.11 channel assignment parameters for the 2.4-GHz radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 channel {channel_number | auto}
   ```

6. Enable CleanAir for 802.11b radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 cleanair
   ```

7. Configure 802.11n antenna for 2.4-GHz radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 dot11n antenna {A | B | C |
   D}
   ```

   Here, are the descriptions of various options:

   A : A is the antenna port A.

   B : B is the antenna port B.

   C : C is the antenna port C.

   D : D is the antenna port D.

8. Disable 802.11b radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 shutdown
   ```

9. Configure transmit power level for 802.11b radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 24ghz slot 0 txpower {tx_power_level | auto}
   ```

   - *tx_power_level* : is the transmit power level in dBm. The valid range is from one to eight.
   - **auto** : enables auto-RF.

Configure 5-GHz radio support for the specified slot number (CLI)

Configure 5-GHz Radio Support for the Specified Slot Number.

Enable and configure 5-GHz radio features on a selected AP slot.

> **Note**
>
> The term **802.11a radio** or **5-GHz radio** are used interchangeably in this document.

Procedure

1. Enable Spectrum Intelligence (SI) for the dedicated 5-GHz radio hosted on slot 1 for a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 SI
   ```

   Here, ***1*** refers to the Slot ID.

2. Configure external antenna gain for 802.11a radios for a specific AP hosted on slot 1.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 antenna ext-ant-gain
   antenna_gain_value
   ```

   antenna_gain_value : refers to the external antenna gain value in multiples of .5 dBi units. The valid range is from zero to 40, with the maximum gain of 20 dBi.

   > **Note**
   >
   > - For APs supporting self-identifying antennas (SIA), the gain depends on the antenna, not on the AP model. The AP learns the gain, so controller configuration is not required.
   > - For APs that do not support SIA, the AP sends the antenna gain in the configuration payload. The default antenna gain depends on the AP model.
   > - Cisco Catalyst 9120E and 9130E APs support self-identifying antennas (SIA). Cisco Catalyst 9115E APs do not support SIA antennas. Although Cisco Catalyst 9115E APs work with SIA antennas, these APs do not automatically detect SIA antennas or add the correct external gain.

3. Configure the antenna mode for 802.11a radios for a specific AP hosted on slot 1.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 antenna mode {omni | sectorA |
    sectorB}
   ```

4. Configure the antenna selection for 802.11a radios for a specific AP hosted on slot 1.

Example

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 antenna selection {internal |
external}
```

5. Configure beamforming for the 5-GHz radio hosted on slot 1 for a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 beamforming
   ```

6. Configure advanced 802.11 channel assignment parameters for the 5-GHz radio hosted on slot 1 for a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 channel {channel_number | auto
    | width [20 | 40 | 80 | 160]}
   ```

   Here,

   channel_number : refers to the channel number. The valid range is from one to 173.

7. Enable CleanAir for 802.11a radio hosted on slot 1 for a given or specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 cleanair
   ```

8. Configure 802.11n for 5-GHz radio hosted on slot 1 for a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 dot11n antenna {A | B | C | D}
   ```

   Here, are the descriptions of various options:

   A : A is the antenna port A.

   B : B is the antenna port B.

   C : C is the antenna port C.

   D : D is the antenna port D.

   Change the channel hosted on slot 1 for a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 rrm channel 2
   ```

   Here, *channel*: refers to the new channel created using 802.11h channel announcement. The valid range is from one to 173, provided 173 is a valid channel in the country where the AP is deployed.

9. Disable 802.11a radio hosted on slot 1 for a specific AP and configure 802.11a radio hosted on slot 1 for a specific AP.

Example

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 shutdown
```

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 txpower {tx_power_level | auto}
```

- *tx_power_level* : is the transmit power level in dBm. The valid range is from one to eight.
- **auto** : enables auto-RF.

Configure 6-GHz radio support for the specified slot number

Configure 6-GHz radio support for the specified slot number.

Enable and configure 6-GHz wireless radios on a Cisco AP to operate optimally within regulatory requirements.

Before you begin
Set the static channel before you change the channel width. As there are no external antenna APs, antennas must be captive (always internal) for 6-GHz, in accordance with regulatory requirements.

Procedure

1. Enter the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Configure the antenna port for 802.11 6-Ghz radios for a specific AP.

   Example

   ```
   Device# ap name Cisco-AP dot11 6ghz slot 3 antenna port {A | B | C | D}
   ```

   Here, are the descriptions of various options:

   A : A is the antenna port A.

   B : B is the antenna port B.

   C : C is the antenna port C.

   D : D is the antenna port D.

3. Configure the antenna selection, either internal or external, for 802.11 6-Ghz radios for a specific AP.

   Example

   ```
   Device# ap name Cisco-AP dot11 6ghz slot 1 antenna selection {internal |
   external}
   ```

> **Note**
>
> - For APs supporting self-identifying antennas (SIA), the antenna determines the gain. The AP learns the gain automatically; you do not need to configure the controller.
> - For APs that do not support SIA, the APs send the antenna gain in the configuration payload, where the default antenna gain depends on the AP model.
> - Cisco Catalyst 9120E and 9130E APs support self-identifying antennas (SIA). Cisco Catalyst 9115E APs do not support SIA antennas. Although Cisco Catalyst 9115E APs work with SIA antennas, these APs do not automatically detect SIA antennas or add the correct external gain.

4. Configure advanced 802.11 channel assignment parameters for the 6-GHz radio hosted on slot 3 for a specific AP.

   Example

   ```
   Device# ap name Cisco-AP dot11 6ghz slot 3 channel channel_number { auto |
   width [160 | 20 | 40 | 80]}
   ```

   Here, *channel_number* : refers to the channel number. The valid range is from one to 233.

5. Enable basic service set (BSS) color for 802.11 6-Ghz radio for a given or specific AP.

   Example

   ```
   Device# ap name Cisco-AP dot11 6ghz slot 3 dot11ax bss-color {bss-color-number
    | auto}
   ```

   Here, *bss-color-number*: refers to the BSS color number. The valid range is from one to 63.

6. Configure the 802.11 6-Ghz radio role

   Example

   ```
   Device# ap name Cisco-AP dot11 6ghz slot 3 radio role {auto | manual
   [client-serving | monitor | sniffer]}
   ```

7. Configure a new channel using 802.11h channel announcement.

   Example

   ```
   Device# ap name Cisco-AP dot11 6ghz slot 3 rrm channel 1
   ```

   Here, *channel* : refers to the new channel created using 802.11h channel announcement. The valid range is from one to 233.

8. Disable the 802.11 6-Ghz radio on the Cisco AP.

   Example

   ```
   Device# ap name Cisco-AP dot11 6ghz slot 3 shutdown
   ```

9. Configure 802.11 6-Ghz Tx power level.

Example

```
Device# ap name AP-SIDD-A06 dot11 5ghz slot 1 txpower { tx_power_level | auto}
```

- *tx_power_level* : is the transmit power level in dBm. The valid range is from one to eight.
- **auto** : enables auto-RF.

XOR radio support

Explains the concept of XOR radios in wireless networking and how they enable flexible use of radio hardware for improved network performance.

Dual-band radios in Cisco AP models

Details on dual-band radio support.

A dual-band radio is a device category that

- offers connectivity on more than one frequency band (such as 2.4 GHz and 5 GHz)
- provides flexibility in network configuration, and
- is used in multiple Cisco AP models like the 2800, 3800, 4800, and 9120 series.

Feature history table
Table 42: Feature history table for dual-band radios in Cisco AP models

Feature name	Release information	Feature description
Dual-band (XOR) radio support	Cisco IOS XE 16.10.1	The Dual-band (XOR) radio support offers the ability to serve 2.4 GHz or 5 GHz bands or passively monitor both the bands on the same AP. These APs can be configured to serve clients in 2.4 GHz and 5 GHz bands, or serially scan both 2.4 GHz and 5 GHz bands on the flexible radio while the main 5 GHz radio serves clients.

Key features of dual-band radios in Cisco APs

The XOR radio support can be steered manually or automatically.

- Manual steering of a band on a radio: The band on the XOR radio can only be changed manually.
- Automatic client and band steering on radios: This is managed by the Flexible Radio Assignment (FRA) feature that monitors and changes the band configurations as per site requirements.

Client steering

When a radio moves between bands (from 2.4 GHz to 5 GHz and vice versa), clients need to be steered to get an optimal distribution across radios. When an AP has two radios in the 5 GHz band, client steering algorithms contained in the FRA algorithms are used to steer a client between the same band co-resident radios

Cisco APs and dual-band radios

Cisco 2800, 3800, 4800, and 9120 series AP models are equipped with dual-band (XOR) radios. These models have

- radios that operate on either 2.4 GHz or 5 GHz bands, or
- the ability to passively monitor both the bands on the same AP.

These APs can be configured to serve clients in 2.4 GHz and 5 GHz bands, or serially scan both 2.4 GHz and 5 GHz bands on the flexible radio while the main 5 GHz radio serves clients.

Cisco AP models up to the Cisco 9120 APs are designed to support dual 5 GHz band operations with the *i* model supporting a dedicated Macro or Micro architecture and the *e* and *p* models supporting Macro or Macro. The Cisco 9130AXI APs support dual 5 GHz operations as Macro or Micro cell.The CW9166I supports Macro or Macro cell.

Cisco Catalyst Wireless 9166 AP (CW9166) has XOR function for a dual 5 GHz 4x4 or 5 GHz 4x4 and 6 GHz 4x4 radios. These radios can also be configured as client serving, monitor or as a sniffer interface like the earlier XOR radios.

> **Note**
>
> For all countries that do not support 6 GHz spectrum for use of Wi-Fi, when the Cisco Catalyst Wireless 9166I AP operates as dual 5 GHz, the 5 GHz channels will be locked on both the radios even if slot 2 is disabled or set up for monitoring.

Wi-Fi 7 APs compatibility

For Wi-Fi 7 APs, the XOR radio operates in 2.4 GHz or 5 GHz low band mode (UNII 1-2A). By default, the XOR radio is brought up in 2.4 GHz band. The existing controller dual-band commands are used to switch between 2.4 GHz and 5 GHz bands. When XOR radio is switched to 5 GHz band, the Wi-Fi 7 APs operate in dual 5 GHz mode. To provide enough RF isolation between both 5 GHz radios, both the 5 GHz radios operate in band-locked mode and radio frequency.

Limitations

Limitations for XOR radio support.

These are the limitations for XOR radio support:

- RF measurement is disabled when a static channel is configured on slot 1. As a result, the dual-band radio slot 0 operates only with 5 GHz radios and not in the monitor mode. When slot 1 radio is disabled, RF measurement will not run, and the dual band radio slot 0 will be only on 2.4 GHz radio.
- Only one of the 5 GHz radios can operate in the UNII band (100 to 144), due to an AP limitation to maintain the power budget within the regulatory limit.

Configure default XOR radio support (CLI)

Configure Default XOR Radio Support using commands.

Complete this task to configure the basic settings for XOR radio support.

Before you begin

> **Note**
>
> The default radio points to the XOR radio hosted on slot 0.

Procedure

1. Enable the priviledged EXEC mode.

   Example

   ```
   Device> enable
   ```

2. Configure the 802.11 dual-band external antenna gain on the AP.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band antenna ext-ant-gain
   antenna_gain_value
   ```

   Enter external antenna gain value in multiple of .5 dBi units (that is, an integer value 4 means 4 x 0.5 = 2 dBi of gain).

3. Shut down the default dual-band radio.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band shutdown
   ```

   Use the **no** form of the command to enable the radio.

4. Switch to the client-serving mode of the AP.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band role manual client-serving
   ```

5. Switch to the 2.4 GHz radio band.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band band 24ghz
   ```

The basic configuration of the XOR radio is completed.

Example

```
Device> enable
Device# ap name ap-name  dot11 dual-band antenna ext-ant-gain 2
Device# ap name ap-name  dot11 dual-band shutdown
```

```
Device# Device# ap name ap-name  dot11 dual-band role manual client-serving
Device# Device# ap name ap-name  dot11 dual-band band 24ghz
```

What to do next

Continue with the configuration steps listed in the "Configure advanced settings" to complete the configuration.

Configure advanced settings (CLI)

Complete this task to configure the advanced settings for XOR radio support, such as, transmit power, channel, channel width, CleanAir settings and so on.

Procedure

1. Configure the transmit power for the radio on the AP.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band txpower {transmit_power_level | auto}
   ```

   Note

   When an FRA-capable radio (slot 0 on 9120 AP[for instance]) is set to Auto, you cannot configure static channel and Txpower on this radio.

   If you want to configure static channel and Txpower on this radio, you will need to change the radio role to Manual Client-Serving mode.

   This note is not applicable for Cisco Catalyst Wireless 9166 AP (CW9166).

2. Configure the channel number for the dual band.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band channel 40
   ```

   The range of the *channel-number* is from 1 to 173.

3. Optional: Enable the auto channel assignment for the dual-band.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band channel auto
   ```

4. Configure the channel width for the dual band.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band channel width {20 MHz | 40 MHz | 80
    MHz | 160 MHz}
   ```

5. Enable the Cisco CleanAir management on the dual-band radio.

Example

```
Device# ap name ap-name  dot11 dual-band cleanair
```

6. Configure a band for the Cisco CleanAir feature.

   Example

   ```
   Device# ap name ap-name [no] dot11 dual-band cleanair band 5 GHz
   ```

   Selects a band for the Cisco CleanAir feature.

   Use the **no** form of this command to disable the Cisco CleanAir feature.

7. Configure the 802.11n dual-band antenna and antenna ports selection for a specific AP.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band dot11n antenna {A | B | C | D}
   ```

   Here, A, B, C, and D are the antenna ports.

The configuration of the additional features for dual-band (XOR) radios are completed.

Example

```
Device# ap name ap-name  dot11 dual-band txpower auto
Device# ap name ap-name  dot11 dual-band channel 40
Device# ap name ap-name  dot11 dual-band channel width 40 MHz
Device# ap name ap-name  dot11 dual-band cleanair
Device# ap name ap-name [no] dot11 dual-band cleanair band 5 GHz
Device# ap name ap-name  dot11 dual-band dot11n antenna A
```

Configure XOR radio support for the specified slot number (GUI)

Configure XOR radio support for the specified slot number using GUI.

Complete this task to configure XOR radio for the specified slot number.

Procedure

---

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. In the **Dual-Band Radios** section, select the AP for which you want to configure dual-band radios.

   The AP name, MAC address, CleanAir capability and slot information for the AP are displayed. If the Hyperlocation method is HALO, it displays the antenna PID and antenna design specifics.

3. Click **Configure**.
4. In the **General** tab, set the **Admin Status** as required.
5. Set the **CleanAir Admin Status** field to Enable or Disable.
6. Click **Update & Apply to Device**.

The XOR radio support for the specified slot number has been configured.

Configure XOR radio support for the specified slot number (CLI)

Configure XOR Radio Support for the Specified Slot Number using commands.

Complete this task to configure XOR radio for a specified slot number.

Procedure

1. Enable the priviledged EXEC mode.

   Example

   ```
   Device> enable
   ```

2. Configure dual-band antenna for the XOR radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name ap-name dot11 dual-band slot 0 antenna ext-ant-gain
   external_antenna_gain_value
   ```

   Here, *external_antenna_gain_value* is the external antenna gain value in multiples of .5 dBi unit. The range is from 0 to 40.

   Note

   - For APs supporting self-identifying antennas (SIA), the gain depends on the antenna, and not on the AP model. The gain is learned by the AP and there is no need for controller configuration.
   - For APs that do not support SIA, the APs send the antenna gain in the configuration payload, where the default antenna gain depends on the AP model.

3. Configure the current band for XOR radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name ap-name dot11 dual-band slot 0 band {24ghz | 5ghz}
   ```

4. Configures dual-band channel for the XOR radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name ap-name dot11 dual-band slot 0 channel {channel-number | auto
    | width [160 | 20 | 40 | 80]}
   ```

   The range of the *channel_number* is from 1 to 165.

5. Enable the Cisco CleanAir feature for dual-band radios hosted on slot 0 for a specific AP.

Example

```
Device# ap name ap-name dot11 dual-band slot 0 cleanair band {24Ghz | 5ghz}
```

6. Configure 802.11n dual-band parameters hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name ap-name dot11 dual-band slot 0 dot11n antenna {A | B | C |
   D}
   ```

   Here are the antenna configurations.

   A : Enables antenna port A.

   B : Enables antenna port B.

   C : Enables antenna port C.

   D : Enables antenna port D.

7. Configure dual-band role for the XOR radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name ap-name dot11 dual-band slot 0 role {auto | manual
   [client-serving | monitor]}
   ```

   Here are the dual-band roles.

   - **auto** : refers to the automatic radio role selection.
   - **manual** : refers to the manual radio role selection.

> **Note**
>
> When Flexible Radio Assignment (FRA) is disabled, XOR radios whose roles were changed by Radio Resource Management (RRM) might not automatically revert to their default radio role. To restore an individual AP to the default client-serving role, use the these commands:
>
> ```
> Device# ap name ap-name dot11 dual-band slot 0 shutdown
> ```
>
> ```
> Device# ap name ap-name dot11 dual-band slot 0 role manual client-serving
> ```
>
> ```
> Device# ap name ap-name dot11 dual-band slot 0 band 24ghz
> ```
>
> ```
> Device# ap name ap-name no dot11 dual-band slot 0 shutdown
> ```

8. Disable dual-band radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name ap-name [no] dot11 dual-band slot 0 shutdown
   ```

   Use the **no** form of this command to enable the dual-band radio.

9. Configure dual-band transmit power for XOR radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name ap-name dot11 dual-band slot 0 txpower {tx-power-level | auto}
   ```

   Here,

   - *tx_power_level* : is the transmit power level in dBm. The range is from 1 to 8.
   - **auto** : enables auto-RF.

The XOR radio for the specified slot number is configured.

Example

```
Device> enable
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 antenna
ext-ant-gain 2
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 band 24ghz

                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 channel 3
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 cleanair
```

```
band 24Ghz
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 dot11n
antenna A
              Device# ap name AP-SIDD-A06 [no] dot11 dual-band slot 0 shutdown
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 txpower 2
```

Verify XOR radio support

To verify the XOR radio support parameters, use these **show** commands:

Verify auto-RF information

To display the auto-RF information for a Cisco AP, use the **show ap name** *ap-name* **auto-rf dot11 dual-band** command.

```
Device# show ap name cisco-ap-name auto-rf dot11 dual-band
```

Verify the list of BSSIDs

To display the list of BSSIDs for a Cisco AP, use the **show ap name** *ap-name* **wlan dot11 dual-band** command.

Receiver only dual-band radio support

Explains the features, functions, and operational constraints of receiver only dual-band radios in WLAN deployments.

A receiver only dual-band radio is a WLAN radio type that

- configures the dual-band Rx-only radio features for an AP with dual-band radios
- serves dedicated functions such as analytics, hyperlocation, wireless security monitoring, and BLE Angle of Arrival (AoA), and
- continually remains in monitor mode, preventing any *tx-rx* configuration for client or data traffic.

Enable CleanAir with receiver Only dual-band radio on a Cisco AP (GUI)

Enable CleanAir with Receiver Only Dual-Band Radio on a Cisco Access Point using GUI.

Enable CleanAir functionality on a Cisco AP that is configured with a receiver only dual-band radio through the GUI.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. In the **Dual-Band Radios** settings, click the AP for which you want to configure the dual-band radios.
3. In the **General** tab, enable the **CleanAir** toggle button.
4. Click **Update & Apply to Device**.

Enabling CleanAir with receiver only dual-band radio on a Cisco AP (CLI)

Enable CleanAir with receiver only dual-band radio on a Cisco AP using commands.

Enable CleanAir spectrum analysis for a dual-band receiver-only radio on a specified Cisco AP.

Procedure

1. Enter the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Enable CleanAir with receiver only (Rx-only) dual-band radio on a specific AP.

   Example

   ```
   Device# ap name AP-SIDD-A06 dot11 rx-dual-band slot 2 cleanair band 24Ghz
   ```

   ```
   Device# ap name AP-SIDD-A06 [no] dot11 rx-dual-band slot 2 cleanair band 24Ghz
   ```

   Here, 2 refers to the slot ID. Use the ***no*** form of this command to disable CleanAir.

Disable receiver only dual-band radio on a Cisco AP (GUI)

Disable receiver only dual-band radio on a Cisco AP using GUI.

Prevent the AP from operating a receiver-only dual-band radio by using the GUI.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. In the **Dual-Band Radios** settings, click the AP for which you want to configure the dual-band radios.
3. In the **General** tab, disable the **CleanAir Status** toggle button.
4. Click **Update & Apply to Device**.

Disable receiver only dual-band radio on a Cisco AP (CLI)

Disable receiver only dual-band radio on a Cisco AP using commands.

Disable the receiver only dual-band radio function on a specific Cisco AP using commands.

Procedure

1. Enter the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Disable receiver only dual-band radio on a specific Cisco AP.

Example

```
Device# ap name AP-SIDD-A06 dot11 rx-dual-band slot 2 shutdown
```

```
Device# ap name AP-SIDD-A06 [no] dot11 rx-dual-band slot 2 shutdown
```

Here, 2 refers to the slot ID. Use the ***no*** form of this command to enable receiver only dual-band radio.

Configure client steering (CLI)

Configure client steering using commands.

Enable and customize client steering on a wireless device to optimize client distribution between macro (5 GHz) and micro (2.4 GHz) radios, improving wireless network performance and load balancing.

Before you begin
Enable Cisco CleanAir on the corresponding dual-band radio.

Procedure

1. Configure the macro-micro client load-balancing window for a set number of clients.

   Example

   ```
   Device(config)# wireless macro-micro steering transition-threshold
   balancing-window number-of-clients(0-65535) 10
   ```

2. Configure the macro-micro client parameters for a minimum client count for transition.

   Example

   ```
   Device(config)# wireless macro-micro steering transition-threshold client
   count number-of-clients(0-65535) 10
   ```

3. Configure the macro-to-micro transition RSSI.

   Example

   ```
   Device(config)# wireless macro-micro steering transition-threshold
   macro-to-micro RSSI-in-dBm( -128—0) -100
   ```

4. Configure the micro-to-macro transition RSSI.

   Example

   ```
   Device(config)# wireless macro-micro steering transition-threshold
   micro-to-macro RSSI-in-dBm(-128—0) -110
   ```

5. Configure the number of probe cycles to be suppressed.

Example

```
Device(config)# wireless macro-micro steering probe-suppression aggressiveness
 number-of-cycles(-128—0) -110
```

6. Configure the macro-to-micro probe in RSSI.

   Example

   ```
   Device(config)# wireless macro-micro steering probe-suppression hysteresis
   RSSI-in-dBm -5
   ```

   The range is between -6 to -3.

7. Enable probe suppression mode.

   Example

   ```
   Device(config)# wireless macro-micro steering probe-suppression probe-only
   ```

8. Enable probe and single authentication suppression mode.

   Example

   ```
   Device(config)# wireless macro-micro steering probe-suppression probe-auth
   ```

9. Display the wireless client steering information.

   Example

   ```
   Device# show wireless client steering
   ```

Verify Cisco APs with dual-band radios

Verify Cisco APs with dual-band radios.

To verify the APs with dual-band radios, use the command:

```
Device# show ap dot11 dual-band summary


AP Name Subband Radio      Mac   Status Channel Power Level Slot ID Mode
-------------------------------------------------------------------------
4800   All 3890.a5e6.f360 Enabled (40)* *1/8     (22 dBm)      0   Sensor
4800   All 3890.a5e6.f360 Enabled N/A    N/A       2               Monitor
```

OFDMA support for 802.11ax APs

Explains the features, functions, and operational constraints of receiver only dual-band radios in WLAN deployments.

OFDMA support for 802.11ax (Wi-Fi 6) APs is a set of wireless networking capabilities that

- improves performance in dense deployments using both 2.4 GHz and 5 GHz bands

- enables simultaneous data transmissions with technologies such as Orthogonal Frequency-Division Multiple Access (OFDMA) and MU-MIMO (Multi-User, Multiple-Input, Multiple-Output), and
- supports uplink and downlink multi-user operations across multiple client devices.

Cisco Catalyst 9100 series APs support IEEE 802.11ax and are recommended for high-performance environments.

The OFDMA mechanism ensures all users transmit or receive in synchronized time windows, ending simultaneously. This coordination supports independent data streams on the same radio frequencies for increased efficiency.

Feature history

Table 43: Feature history for OFDMA support for 802.11ax APs

Feature Name	Release Information	Feature Description
OFDMA support for 802.11ax APs	Cisco IOS XE 16.12.1	This feature enhances wireless network efficiency and user experience in high-density environments by enabling simultaneous multi-user transmissions with OFDMA and MU-MIMO across both 2.4 GHz and 5 GHz bands.

Supported modes on 802.11ax APs

802.11ax APs supported modes details.

The following AP modes are supported:

- Local mode
- FlexConnect mode
- Bridge mode
- FlexConnect+Mesh mode

Configure 802.11AX (GUI)

Configure 11AX using GUI.

Enable 802.11ax features for 5 GHz and 2.4 GHz frequency bands to optimize high throughput wireless performance.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **High Throughput**.
2. Click the **5 GHz Band** tab.
   a) Expand the **11ax** section.
   b) Check the **Enable 11ax** and **Multiple Bssid** check boxes, if required.
   c) Check either the **Select All** check box to configure all the data rates or select the desired options from the available data rates list.
3. Click the **2.4 GHz Band** tab.
   a) Expand the **11ax** section.
   b) Check the **Enable 11ax** and **Multiple Bssid** check boxes, if required.

c) Check either the **Select All** check box to configure all the data rates or select the desired options from the available data rates list.

Configure channel width (CLI)

Configure channel width with commands.

Set the 802.11 radio channel width for wireless performance tuning

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure channel width for 802.11 radios as 160.

   Example

   ```
   Device(config)# ap dot11 { 24ghz | 5ghz} rrm channel dca chan-width 160
   ```

   Use the **no** form of the command to disable the configuration.

   > **Note**
   >
   > Cisco Catalyst 9115 and C9120 series APs do not support 80+80 channel width. Cisco Catalyst 9117 series APs do not support OFDMA in 160 channel width.

3. Configure an RF profile and enters RF profile configuration mode.

   Example

   ```
   Device(config)# ap dot11 { 24ghz | 5ghz} rf-profile ax-profile
   ```

4. Configure the RF profile DCA channel width.

   Example

   ```
   Device(config-rf-profile)# channel chan-width 160
   ```

Configure 802.11ax radio parameters (GUI)

Configure 802.11ax Radio Parameters using GUI.

Configure high throughput settings and 802.11ax (Wi-Fi 6) radio parameters for the 2.4 GHz and 5 GHz bands on APs using the GUI.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **High Throughput** > **5 GHz Band** > **11ax**.
2. Check or uncheck the **Enable 11 n** check box.
3. Check the check boxes for the desired MCS/(data rate) or to select all of them, check the **Select All** check box. Click **Apply**.
4. Choose **Configuration** > **Radio Configurations** > **High Throughput** > **2.4 GHz Band** > **11ax**.
5. Check or uncheck the **Enable 11 n** check box.
6. Check the check boxes for the desired MCS/(data rate) or to select all of them, check the **Select All** check box. Click **Apply**.
7. Choose **Configuration** > **Wireless** > **Access Points**. Click the Access Point.
8. In the **Edit AP** dialog box, enable the **LED State** toggle button and select the LED brightness level from the **LED Brightness Level** drop-down list.
9. Click **Update & Apply to Device**.

Configure 802.11ax radio parameters (CLI)

Configure 802.11ax radio parameters using commands.

Set up and manage 802.11ax radio parameters on your Cisco device using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the 802.11 6 GHz dot11ax parameters.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz |6ghz} dot11ax
   ```

   Use the **no** form of the command to disable the configuration.

3. Enable the 802.11ax 2.4-Ghz, 5-Ghz, or 6-Ghz band modulation and coding scheme (MCS) transmission rates.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} dot11ax mcs tx index 11
   spatial-stream 8
   ```

4. Optional: Configure the led brightness level.

Example

```
Device(config)# ap led-brightness 6
```

Set up the 802.11ax radio parameters (CLI)

Set up the 802.11ax radio parameters using commands.

Configure advanced wireless radio settings to optimize performance and coverage on Cisco APs supporting 802.11ax.

Procedure

1. Enter the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Configure the led brightness level.

   Example

   ```
   Device# ap name ax-ap led-brightness-level 6
   ```

3. Configure the 802.11n - 5 GHz antenna selection.

   Example

   ```
   Device# ap name ap1 dot11 {24ghz | 5ghz} dot11n antenna A
   ```

   Use the **no** form of the command to disable the configuration.

4. Configure 802.11 channel width.

   Example

   ```
   Device# ap name ap1 dot11 {24ghz | 5ghz} channel width 160
   ```

5. Configure the advanced 802.11 secondary 80Mhz channel assignment parameters.

   Example

   ```
   Device# ap name ap1 dot11 {24ghz | 5ghz} secondary-80 12
   ```

Configure OFDMA on a WLAN (CLI)

Configure OFDMA on a WLAN using commands.

Enable OFDMA and related 802.11ax features on a WLAN interface.

> **Note**
>
> For Cisco Catalyst 9115 and 9120 series APs, the configuration is applied per radio, not per WLAN. The controller keeps this feature enabled if it is enabled on any WLAN.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration mode.

   Example

   ```
   Device(config)# wlan wlan1
   ```

3. Enable the downlink connection that uses the OFDMA technology.

   Example

   ```
   Device(config-wlan)# dot11ax downlink-ofdma
   ```

   Use the **no** form of the command to disable the configuration.

4. Enable the uplink connection that uses the OFDMA technology.

   Example

   ```
   Device(config-wlan)# dot11ax uplink-ofdma
   ```

5. Enable the downlink connection that uses the MUMIMO technology.

   Example

   ```
   Device(config-wlan)# dot11ax downlink-mumimo
   ```

6. Enable the uplink connection that uses the MUMIMO technology.

   Example

   ```
   Device(config-wlan)# dot11ax uplink-mumimo
   ```

7. Enable the TWT broadcast support operation.

Example

```
Device (config-wlan)# dot11ax twt-broadcast-support
```

UNII-3 channels in the –E domain

UNII-3 channels in the -E domain details

UNII-3 channels are wireless frequency channels that

- operate in the -E domain with low power
- are disabled by default starting from Cisco IOS XE 17.17.1, and
- may cause coverage gaps for APs when selected by Radio Resource Management (RRM), if the client devices do not support those channels.

> **Note**
>
> Verify that your AP is listed among the Supported APs. If your AP is not listed, UNII-3 channels is disabled to meet regulatory requirements. For more information, see CSCwq22634.

Configure UNII-3 channels for –E domain (CLI)

Enable UNII-3 low-power channels for all 802.11a radios operating in the 5 GHz band in the -E domain.

Before you begin

UNII-3 low-power channels is applicable only for 5 GHz band.

Procedure

1. In global configuration mode, configure either of the following:
   a) Configure the UNII-3 channels for all 802.11a radios in the 5 GHz band for the default RF profile.

   Example

   ```
   Device# configure terminal
   ```

   ```
   Device(config)# ap dot11 5ghz rrm channel unii3-low-power-channels
   ```

   b) Configure the UNII-3 channels for all 802.11a radios in the 5 GHz band for a custom RF profile.

   Example

   ```
   Device(config)# ap dot11 5ghz rf-profile ax-profile
   ```

2. Enable UNII-3 low power channels.

Example

```
Device(config-rf-profile)# channel unii3-low-power-channels
```

3. Return to the privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Example

```
Device(config-rf-profile)# end
```

Restrictions and supported APs

Restrictions for UNII-3 channels in the -E domain

Channel availability depends on both the configured country code and hardware certification status to guarantee lawful operation and prevent interference.

- Ensure that your configured country code is correct to guarantee legal operation and prevent interference. See CSCwd27071.
- Check whether your AP is included in the Supported APs list. If it is not, UNII-3 channels are disabled to comply with regulatory requirements. See CSCwq22634.
- If your AP is not on this list of Supported APs, UNII-3 channels are disabled or unavailable to ensure compliance with regulatory requirements.

Supported APs

- Cisco Catalyst 9136 Access Point
- Cisco Catalyst Wireless 9162I Access Point
- Cisco Catalyst Wireless 9164I Access Point
- Cisco Catalyst Wireless 9166I Access Point
- Cisco Catalyst Wireless 9178I Access Point
- Cisco Catalyst Wireless 9176I Access Point
- Cisco Catalyst Wireless 9176D1 Access Point

Verify UNII-3 channel configurations

To verify the status of the UNII-3 channel configuration in the AP RF profile details, use the command:

```
Device# show ap rf-profile name ax-profile detail

A-MPDU Window                    :  255
Zero Wait DFS                    :  Disabled
UNII-3 Low Power Channels        : Disabled
Preamble Puncturing              : Disabled
Packets Retries Settings:
Management Retries               : -1
Aggregate Retries                : -1
Non_Aggregate Retries            : -1
```

To verify the UNII-3 channel status in the 5-GHz radio configuration details, use the command:

```
Device# show ap dot11 5ghz channel

Leader Automatic Channel Assignment
Channel Assignment Mode                       : AUTO
Channel Update Interval                       : 600 seconds
Anchor time (Hour of the day)                 : 0
Channel Update Contribution
Noise                                         : Enable
Interference                                  : Enable
Load                                          : Disable
Device Aware                                  : Disable
CleanAir Event-driven RRM option              : Disabled
Zero Wait DFS                                 : Disabled
Channel Assignment Leader                     : EWLC_RRM_DURGA (9.9.38.103)
(2009:300:38::103)
Last Run                                      : 300 seconds ago

DCA Sensitivity Level                         : MEDIUM : 15 dB
DCA Channel Width                             : 80 MHz
DCA Minimum Energy Limit                      : -95 dBm
DCA Aggressive on startup                     : Enabled
UNII-3 Low Power Channels                     : Disabled
Channel Energy Levels
Minimum                                       : -86 dBm
Average                                      : -84 dBm
Maximum                                      : -81 dBm
Channel Dwell Times
Minimum                                      : 53 minutes 47 seconds
Average                                      : 19 hours 35 minutes 47 seconds
Maximum                                     : 1 day 4 hours 51 minutes 54 seconds

802.11a 5 GHz Auto-RF Channel List
Allowed Channel List                         :
44,48,52,60,64,100,104,108,112,116,120,124,128,140,144,153,157,161
Unused Channel List                         : 36,40,56,132,136,149,165,169,173
```

Verify channel width

Verify channel width.

To verify the channel width and other channel information, use the commands:

```
Device# show ap dot11 5ghz summary

AP Name             Mac Address       Slot    Admin State  Oper State  Channel
    Width    Txpwr
-------------------------------------------------------------------------------
AP80e0.1d75.6954    80e0.1d7a.7620     1       Enabled      Up          (52)*
      160      1(*)
```

```
Device# show ap dot11 dual-band summary

AP Name         Subband    Radio Mac        Status     Channel    Power Level
 Slot ID         Mode
-------------------------------------------------------------------------------
```

```
kartl28021mi    All       002a.1058.38a0    Enabled   (52)*      (1)*
    1           REAP
```

```
Device# show ap name <ap-name> channel

802.11b/g Current Channel                          : 11
Slot ID                                            : 0
Allowed Channel List                               : 1,2,3,4,5,6,7,8,9,10,11
802.11a Current Channel ..................... 52  (160 MHz)
Slot ID                                            : 1
Allowed Channel List                               :
36,40,44,48,52,56,60,64,100,104,108,112,116,132,136,140,149,153,157,161,165
```

```
Device# show ap name <ap-name> config slot <slot-num>
.
.
.
Phy OFDM Parameters
Configuration                                 : Automatic
Current Channel                               : 52
Extension Channel                             : No Extension
Channel Width                                 : 160 MHz
Allowed Channel List                           :
36,40,44,48,52,56,60,64,100,104,108,112,116,132,136,140,149,153,157,161,165
TI Threshold                                   : 0
```

```
Device# show ap dot11 5ghz channel
.
.
.
DCA Sensitivity Level                         : MEDIUM : 15 dB
DCA 802.11n/ac Channel Width                  : 160 MHz
DCA Minimum Energy Limit                      : -95 dBm
.
.
.
```

```
Device# show ap rf-profile name <name> detail
.
.
.
Unused Channel List                : 165
DCA Bandwidth                      :  160 MHz
DCA Foreign AP Contribution        : Enabled
.
.
.
```

Verify client details

Verify client details.

To verify the client information, use the commands:

```
Device# show wireless client mac-address <mac-address> detail

Client MAC Address : a886.ddb2.05e9
Client IPv4 Address : 169.254.175.214
```

```
Client IPv6 Addresses : fe80::b510:a381:8099:4747
2001:DB8:300:57:4007:6abb:2c9a:61e2
Client Username: N/A
Voice Client Type   : Unknown
AP MAC Address : c025.5c55.e400
AP Name: APe4c7.22b2.948e
Device Type: N/A
Device Version: N/A
AP slot : 0
Client State : Associated
Policy Profile : default-policy-profile
Flex Profile : default-flex-profile
Wireless LAN Id : 1
Wireless LAN Name: SSS_OPEN
BSSID : c025.5c55.e406
Connected For : 23 seconds
Protocol : 802.11ax - 5 GHz
Channel : 8
Client IIF-ID : 0xa0000001
Association Id : 1
Authentication Algorithm : Open System
Client CCX version : No CCX support
Session Timeout : 86400 sec (Remaining time: 86378 sec)
.
.
.
```

```
Device# show wireless client summary

Number of Local Clients: 1

MAC Address    AP Name                          WLAN  State            Protocol
 Method     Role
-------------------------------------------------------------------------------------
a886.ddb2.05e9 APe4c7.22b2.948e                  1     Run               11ax(5)
 None       Local
```

```
Device# show wireless stats client detail


 Total Number of Clients : 1

 Protocol Statistics

 ---------------------------------------------------------------------------

 Protocol            Client Count
 802.11b              : 0
 802.11g              : 0
 802.11a              : 0
 802.11n-2.4GHz       : 0
 802.11n-5 GHz        : 0
 802.11ac             : 0
 802.11ax-5 GHz       : 0
 802.11ax-2.4 GHz     : 0
 802.11ax-6 GHz       : 1
```

Verify radio configuration

Verify radio configuration.

To verify the radio configuration information, use the commands:

```
Device# show ap dot11 5ghz network

802.11a Network                              : Enabled
.
.
.
802.11ax                                     : Enabled
  DynamicFrag                                : Enabled
  MultiBssid                                 : Disabled
802.11ax MCS Settings:
  MCS  7, Spatial Streams = 1                : Disabled
  MCS  9, Spatial Streams = 1                : Disabled
  MCS 11, Spatial Streams = 1                : Disabled
  MCS  7, Spatial Streams = 2                : Supported
  MCS  9, Spatial Streams = 2                : Supported
  MCS 11, Spatial Streams = 2                : Supported
  MCS  7, Spatial Streams = 3                : Supported
  MCS  9, Spatial Streams = 3                : Disabled
  MCS 11, Spatial Streams = 3                : Disabled
  MCS  7, Spatial Streams = 4                : Supported
  MCS  9, Spatial Streams = 4                : Supported
  MCS 11, Spatial Streams = 4                : Supported
  MCS  7, Spatial Streams = 5                : Supported
  MCS  9, Spatial Streams = 5                : Supported
  MCS 11, Spatial Streams = 5                : Supported
  MCS  7, Spatial Streams = 6                : Supported
  MCS  9, Spatial Streams = 6                : Supported
  MCS 11, Spatial Streams = 6                : Supported
  MCS  7, Spatial Streams = 7                : Supported
  MCS  9, Spatial Streams = 7                : Supported
  MCS 11, Spatial Streams = 7                : Supported
  MCS  7, Spatial Streams = 8                : Supported
  MCS  9, Spatial Streams = 8                : Supported
  MCS 11, Spatial Streams = 8                : Supported
Beacon Interval                              : 100
.
.
.
Maximum Number of Clients per AP Radio       : 200
```

```
Device# show ap dot11 24ghz network

802.11b Network                              : Enabled
.
.
.
802.11axSupport..................................... Enabled
      dynamicFrag.............................. Disabled
      multiBssid............................... Disabled
802.11ax                                     : Enabled
  DynamicFrag                                : Enabled
  MultiBssid                                 : Enabled
802.11ax MCS Settings:
  MCS  7, Spatial Streams = 1                : Supported
  MCS  9, Spatial Streams = 1                : Supported
  MCS 11, Spatial Streams = 1                : Supported
  MCS  7, Spatial Streams = 2                : Supported
  MCS  9, Spatial Streams = 2                : Supported
  MCS 11, Spatial Streams = 2                : Supported
  MCS  7, Spatial Streams = 3                : Supported
```

```
  MCS  9, Spatial Streams = 3             : Supported
  MCS 11, Spatial Streams = 3             : Supported
  MCS  7, Spatial Streams = 4             : Disabled
  MCS  9, Spatial Streams = 4             : Disabled
  MCS 11, Spatial Streams = 4             : Disabled
Beacon Interval                           : 100
.
.
.
Maximum Number of Clients per AP Radio    : 200
```

```
Device# show ap dot11 6ghz network
802.11 6Ghz Network                       : Enabled
802.11ax                                  : Enabled
.
.
.
802.11ax MCS Settings:
  MCS  7, Spatial Streams = 1             : Supported
  MCS  9, Spatial Streams = 1             : Supported
  MCS 11, Spatial Streams = 1             : Supported
  MCS  7, Spatial Streams = 2             : Supported
  MCS  9, Spatial Streams = 2             : Supported
  MCS 11, Spatial Streams = 2             : Supported
  MCS  7, Spatial Streams = 3             : Supported
  MCS  9, Spatial Streams = 3             : Supported
  MCS 11, Spatial Streams = 3             : Supported
  MCS  7, Spatial Streams = 4             : Supported
  MCS  9, Spatial Streams = 4             : Supported
  MCS 11, Spatial Streams = 4             : Supported
Beacon Interval                           : 95
.
.
.
Maximum Number of Clients per AP Radio    : 200
WiFi to Cellular RSSI Threshold           : -85 dbm
Client Network Preference                 : default
```

```
Device# show wlan id 1
WLAN Profile Name     : wlanon66
================================================
Identifier                                    : 1
Description                                   :
Network Name (SSID)                           : wlanon66
Status                                        : Enabled
Broadcast SSID                                : Enabled
Advertise-Apname                              : Enabled
Universal AP Admin                            : Disabled
Max Associated Clients per WLAN               : 0
Max Associated Clients per AP per WLAN        : 0
Max Associated Clients per AP Radio per WLAN  : 200
OKC                                           : Enabled
Number of Active Clients                      : 0
CHD per WLAN                                  : Enabled
WMM                                           : Allowed
WiFi Direct Policy                            : Disabled
.
.
.
Operational State of Radio Bands
    2.4ghz                                    : UP
```

```
    5ghz                                          : UP
    6ghz                                         : DOWN (Required config: Disable
 WPA2 and Enable WPA3 & dot11ax)
DTIM period for 802.11a radio                     :
DTIM period for 802.11b radio                     :
Local EAP Authentication                          : Disabled
Mac Filter Authorization list name                : Disabled
Mac Filter Override Authorization list name       : Disabled
Accounting list name                              :
802.1x authentication list name                   : Disabled
802.1x authorization list name                    : Disabled
Security
    802.11 Authentication                         : Open System
.
.
.
802.11ac MU-MIMO                                  : Enabled
802.11ax parameters
    802.11ax Operation Status                     : Enabled
    OFDMA Downlink                                : Enabled
    OFDMA Uplink                                  : Enabled
    MU-MIMO Downlink                              : Enabled
    MU-MIMO Uplink                                : Enabled
    BSS Target Wake Up Time                       : Enabled
    BSS Target Wake Up Time Broadcast Support     : Enabled
.
.
.
```

Note

For a 6 GHz radio, the 802.11ax parameters come from the multi BSSID profile associated with the 6 GHz RF profile of the access point. As a result, the dot11ax WLAN parameters are replaced by the parameters from the multi BSSID profile for the 6 GHz band. The 2.4 GHz and 5 GHz band WLANs do not change. These bands continue to use the WLAN parameters for 802.11ax

```
Device# show ap led-brightness-level summary

AP Name                          LED Brightness level
-------------------------------------------------------
AP00FC.BA01.CC00                 Not Supported
AP70DF.2FA2.72EE                 8
AP7069.5A74.6678                 2
APb838.6159.e184                 Not Supported
```

Cisco flexible antenna port

Explains the features and operating modes of Cisco flexible antenna ports used in wireless access points.

A Cisco flexible antenna port is an antenna configuration feature that

- enables APs to assign antenna ports as either single-band or dual-band
- optimizes wireless coverage and performance through flexible radio assignments, and

- supports both 2.4 GHz and 5 GHz wireless bands to accommodate various deployment scenarios.

Multiple antennas on the transmitters and receivers improve AP performance and reliability for your wireless deployment. Multiple antennas improve reception by selecting stronger signals or combining individual signals at the receiver. These mechanisms help APs deliver reliable wireless coverage. You can configure AP antenna ports as dual-band or single-band antennas to optimize radio coverage for your environment.

- In dual-band antenna mode, APs operate in both the 2.4 GHz and 5 GHz bands using all four antennas A, B, C, and D. An example of a dual-band antenna mode AP is the Cisco Industrial Wireless 3702 AP.
- In single-band antenna mode, antennas A and B operate in the 2.4 GHz band, while antennas C and D operate in the 5 GHz band. An example of a single-band antenna mode AP is the Cisco Catalyst Industrial Wireless 6300 AP.

Configure a Cisco flexible antenna port (GUI)

Configure a Cisco flexible antenna port using GUI.

Set the appropriate antenna mode on a selected AP.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click **AP Name**.
3. Click the **Advanced** tab.
4. Select the antenna mode from the **Antenna Mode** drop-down list.
5. Click **Apply & Update**.

Configure a Cisco flexible antenna port (CLI)

Configure a Cisco flexible antenna port using commands.

Set the band mode for a Cisco flexible antenna port to single or dual band using commands.

Procedure

Configure the antenna band mode as single or dual.

Example

```
Device# ap name ap-name antenna-brand-mode {dual | single}
```

Verify flexible antenna port configuration

Verify flexible antenna port configuration.

This is a sample output of the **show ap name** *ap_name* **config general** command that shows the bands selected on a specific AP:

```
Device# show ap name APXXXX.31XX.83XX config general

Cisco AP Name   : APXXXX.31XX.83XX
```

```
================================================
Cisco AP Identifier                             : b4de.312e.00c0
Country Code                                    : Multiple Countries : US,IN
Regulatory Domain Allowed by Country   : 802.11bg:-A   802.11a:-ABDN

AP Submode                                      : Not Configured
Antenna Band Mode                               : Dual
```

This is a sample output of the **show ap name** *ap_name* **config slot 0** command that shows the bands selected on a specific AP with dual-band mode enabled:

```
Device# show ap name APXXXX.31XX.83XX config slot 0 | sec 802.11n Antennas

802.11n Antennas
A                                               : ENABLED
B                                               : ENABLED
C                                               : ENABLED
D                                               : ENABLED

802.11n Antennas
MIMO                                            : x
Tx                                              : Unknown
Rx                                              : Unknown
```

This is a sample output of the **show ap name** *ap_name* **config slot 1** command that shows the bands selected on a specific AP with single-band mode enabled:

```
Device# show ap name APXXXX.31XX.83XX config slot 1 | sec 802.11n Antennas

802.11n Antennas
A                                               : DISABLED
B                                               : DISABLED
C                                               : ENABLED
D                                               : ENABLED

802.11n Antennas
MIMO                                            : x
Tx                                              : Unknown
Rx                                              : Unknown
```

Environmental sensors in APs

Explains how environmental sensors in Cisco Catalyst access points enable real-time monitoring of air quality, temperature, and humidity.

Environmental sensors in APs are wireless AP features that

- enable real-time collection of environmental data such as air quality, temperature, and humidity
- embed sensors directly within supported Cisco Catalyst AP models, and
- integrate with Cisco Spaces and wireless controller CLIs to allow configuration and monitoring of sensor data.

Use the Cisco Catalyst 9800 Series Wireless Controller CLIs to enable or disable the sensors and set their scan interval.

Cisco Catalyst 9136 Series APs include two sensors:

- Total volatile organic compounds (TVOC) air quality sensor
- Combined temperature and humidity sensor

Note

Starting with Cisco IOS XE Cupertino 17.8.1, this feature is supported on Cisco Catalyst 9136 Series APs.

Cisco IOS XE Cupertino 17.9.1 adds support for air quality, temperature, and humidity on Cisco Catalyst Wireless 9166I Series Access Points.

Feature history

Table 44: Feature history for environmental sensors on APs

Feature Name	Release Information	Feature Description
Environmental sensors in APs	Cisco IOS XE 17.8.1	This feature helps you collect real-time environmental data, such as air quality, temperature, and humidity from the environmental sensors embedded in Cisco Catalyst 9136 Series APs.
Environmental sensors in APs	Cisco IOS XE 17.9.1	This feature is supported on Cisco Catalyst Wireless 9166I Series APs.

Use cases

Use cases for environmental sensors in APs.

These are the use cases for the environmental sensors in APs:

- Maintain proper storage conditions and prevent pharmaceutical spoilage in healthcare environments
- Monitor and improve air quality for guests in hospitality settings, and
- Protect perishable products from spoilage in retail spaces.

Configure environmental sensors in an AP profile (CLI)

Configure environmental sensors in an AP Profile using commands.

Enable data collection from environmental sensors in an AP profile to monitor air quality and temperature.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile.

Example

```
Device(config)# ap profile ap-profile-name
```

3. Configure the AP environmental air quality sensor.

   Example

   ```
   Device(config-ap-profile)# sensor environment air-quality
   ```

   Enters AP sensor configuration mode.

4. Enable the AP air quality sensor configuration.

   Example

   ```
   Device(config-ap-sensor)# no shutdown
   ```

5. Configure the AP environmental temperature sensor.

   Example

   ```
   Device(config-ap-profile)# sensor environment temperature
   ```

   Enters AP sensor configuration mode.

6. Enable the AP temperature sensor configuration.

   Example

   ```
   Device(config-ap-sensor)# no shutdown
   ```

7. Configure the data sampling interval, in seconds.

   Example

   ```
   Device(config-ap-sensor)# sampling data sampling interval 200
   ```

   The valid range is between five and 3600. The default value is five. Use the **no** form of this command to set the data sampling interval to the default time of five.

8. Exit the sub mode.

   Example

   ```
   Device(config-ap-sensor)# exit
   ```

Configure environment sensors (CLI)

Configure environment eensors using commands.

If an AP is sending invalid data, such as being located near an air vent or a coffee machine, disable the sensor by entering the necessary commands in the controller's privileged EXEC mode.

> **Note**
>
> A sensor operates in the **Up** state only when both the AP profile configuration state and the AP administrative state are enabled. If either state is disabled, the sensor's operational status remains **Down**.

Procedure

1. Enable the privileged EXEC mode and enter the password if prompted.

   Example

   ```
   Device> enable
   ```

2. Enable the sensor admin state of the AP.

   Example

   ```
   Device# ap name CiscoAP sensor environment {air-quality | temperature} shutdown
   ```

3. Disable the sensor admin state of the AP.

   Example

   ```
   Device# ap name CiscoAP no sensor environment {air-quality | temperature}
   shutdown
   ```

Verify the AP sensor status

Verify the AP sensor status.

To verify the status of the AP sensors, run the command:

```
Device# show ap sensor status

AP Name                          MAC-address          Sensor-type
Config-State     Admin-State     Oper-Status      Sampling-Interval

-------------------------------------------------------------------------------------------
Cisco.1DBC                        xxxx.xxxx.xxx1       Air-quality
Disabled         Enabled          Down             5
Cisco.1DBC                        xxxx.xxxx.xxx2       Temperature
Disabled         Enabled          Down             5
Cisco.1E24                        xxxx.xxxx.xxx3       Air-quality
Disabled         Enabled          Down             5
Cisco.1E24                        xxxx.xxxx.xxx4       Temperature
Disabled         Enabled          Down             5
```

Accelerometer sensor support

Accelerometer sensors allow wireless network controllers to monitor and analyze the physical installation, movement, and orientation of access points. This capability enables optimal placement, enhanced security, and improved radio frequency (RF) performance.

An accelerometer sensor is a hardware component that

- measures AP orientation and tilt using X, Y, and Z coordinates
- detects movement, misplacement, or tampering of AP, and
- enables real-time telemetry integration with wireless controllers and management platforms.

Accelerometer sensor features

- Three-axis detection: The hardware includes a three-axis accelerometer that measures X, Y, and Z coordinates and tilt angles.
- Motion and orientation monitoring: The sensor detects if an AP has been moved, tilted, or had its orientation changed.
- Real-time reporting: The sensor provides real-time data to help troubleshoot installation issues or confirm that the AP is optimally placed for radio frequency (RF) performance.
- Security and tamper detection: Sensors can identify unauthorized movement or tampering with the device.
- Telemetry integration: AP can log tilt and movement events via telemetry to the controller, Cisco Catalyst Center, or Cisco Spaces.

Supported hardware

Accelerometer sensors are available on newer enterprise-grade Cisco Catalyst and Wi-Fi 7 APs, including:

- Cisco Wireless 9174E

> **Note**
>
> CW9174E supports accelerometer only when you use the AP with CW-ANT-T-D2-D8.

- Cisco Wireless 9179F
- Cisco Wireless 9178I Series Access Points
- Cisco Wireless 9176I Series Access Points
- Cisco Wireless 9176D1
- Cisco Catalyst 9166D1 Series Access Points

Restrictions for accelerometer sensor support

This topic outlines the mandatory restrictions for enabling and deploying the accelerometer sensor feature.

- The sensor must be calibrated during the manufacturing process; manual field calibration is not supported.

Troubleshoot accelerometer sensors

This topic outlines the best practices for troubleshooting accelerometer sensors, including steps to verify access point status, enable debugging, and inspect physical installation to ensure accurate sensor readings.

If the accelerometer readings appear inaccurate, use these steps:

- Ensure the AP is fully joined to the controller and running a compatible software version.
- If data reporting issues persist, enable specific debugging for the accelerometer feature to isolate communication problems between the AP and the controller.
- Ensure the AP is securely mounted, as loose mounting brackets may cause fluctuating tilt readings.

Configure the AP accelerometer sensor (CLI)

Use this procedure to shut down or enable the environmental accelerometer sensor on an access point using CLI commands.

Manage the operational state of the accelerometer sensor to monitor or stop monitoring AP movement and orientation.

The environmental accelerometer sensor detects the physical orientation and movement of the access point. You can manually shut down this sensor if it is not required for your deployment.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

2. Disable the environmental accelerometer sensor for a specific AP.

   Example

   ```
   Device# [no] ap name <ap-name> sensor environment accelerometer shutdown
   ```

   - Replace *<ap-name>* with the name of the AP.
   - Use the **no** form of the command to return the sensor to its operational state.

The accelerometer sensor is enabled or disabled for the specified AP as requested.

Configure accelerometer sensor at the AP join profile (CLI)

To enable the accelerometer sensor for all APs associated with a specific profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable or disable the accelerometer sensor for the profile.

Example

```
Device(config-ap-profile)# [no] ap name <ap-name> sensor environment
accelerometer
```

- Replace *<ap-name>* with the name of the AP.
- Use the **no** form of the command to return the sensor to its operational state.

The accelerometer sensor is now enabled for all APs assigned to this profile.

Verify AP sensor status and accelerometer data

This section provides information on how to verify the operational status of AP sensors and retrieve specific accelerometer data from the device and the internal database.

To ensure that AP sensors are correctly configured and operational by verifying their status, tilt angles, and database records.

Verify AP accelerometer database content

To verify AP accelerometer database content, enter this command from the console:

```
Device# show platform software process database wncd 0 chassis activ R0 details
 WNCD_DB "table tbl_ap_accelerometer" content
        Database: WNCD_DB - Table: table tbl_ap_accelerometer - Num Recs: 2
        Records

------------------------------------------------------------------------------
        422 TS:504303508759
        record@tbl_ap_accelerometer: {
        st_ap_accelerometer@st_ap_accelerometer: {
        ap_mac@tdl_mac_addr: {
        mac[0]@U8:148
        mac[1]@U8:13
        mac[2]@U8:75
        mac[3]@U8:156
        mac[4]@U8:39
        mac[5]@U8:224
        }
        last_update@calendar_time: {
        second@I64:1768515191
        microsec@U32:413575
        }
        x_coord@I16:-28
        y_coord@I16:-4
        z_coord@I16:-988
        tilt_angle@I16:1
        }
        }
```

Verify AP sensor status

To verify AP sensor status, enter this command from the console:

```
Device# show ap sensor status
        AP Name             MAC-address     Sensor-type     Config-State
Admin-State  Oper-Status  Sampling-Interval

_____________________________________________________________________________


        AP6849.92FF.A970    ecf4.0c92.ede0  Pressure        Enabled
Disabled     Down         3600
        AP6849.92FF.A970    ecf4.0c92.ede0  Accelerometer   Enabled
Enabled      Up           900
        AP8C88.8152.C6F0    940d.4b9c.27e0  Pressure        Enabled
Disabled     Down         3600
        AP8C88.8152.C6F0    940d.4b9c.27e0  Accelerometer   Enabled
Enabled      Up           900
```

Verify AP accelerometer details

To verify AP accelerometer details, enter this command from the console:

```
Device# show ap name default-ap-profile accelerometer
        Tilt angle        : 13 Degree(s)
        Last update       : 01/15/2026 22:11:47
```

To verify AP accelerometer details, enter this command from the controller:

```
Device# show ap name <ap-name> config general
        Cisco AP Name   : <ap-name>
      ================================================
      Cisco AP Identifier                     : <mac addr>
      Country Code                            : Multiple Countries : GB,IN,US

      Regulatory Domain Allowed by Country    : 802.11bg:-AE   802.11a:-ABDEN
  802.11 6GHz:-BR
      AP Country Code                         : US  - United States
      Tilt angle                              : 178 Degree(s)
```

CAPWAP LAG support

Explains how CAPWAP Link Aggregation (LAG) enables redundancy, simplifies configuration, and ensures continued connectivity between controllers and access points.

A CAPWAP LAG support feature is a controller-access point connectivity mechanism that

- enables link aggregation of controller ports.
- allows automatic failover of traffic if a port fails, and
- simplifies controller configuration by removing the requirement to configure primary and secondary ports for each interface.

Link aggregation ensures that traffic is redirected automatically to a functioning port if one controler port fails. With at least one controller port operating, network connectivity is maintained, APs stay connected and wireless clients remain operational.

CAPWAP (Control and Provisioning of Wireless Access Points) is a protocol that manages communication between wireless LAN controllers and access points.

Link Aggregation refers to using multiple network connections in parallel to increase throughput and provide redundancy.

Supported models

The CAPWAP LAG support feature applies to APs with multiple Ethernet ports for CAPWAP.

- 11AC APs with dual Ethernet ports require CAPWAP AP LAG support for their data channels.
- By default, the second port on Cisco Aironet 1850, 2800, and 3800 Series APs is used for link aggregation. If you disable LAG, you can use the LAG port as an RLAN port.

These APs use LAG port as an RLAN (Remote LAN) port:

- Cisco Aironet 1852E Access Point
- Cisco Aironet 1852I Access Point
- Cisco Aironet 2802E Access Point
- Cisco Aironet 2802I Access Point
- Cisco Aironet 3802E Access Point
- Cisco Aironet 3802I Access Point
- Cisco Aironet 3802P Access Point
- Cisco Catalyst 9136I Access Point

Note

The CAPWAP LAG feature do not support the Cisco Catalyst 9176I and Cisco Catalyst 9176D1 APs.

Restrictions for CAPWAP LAG support

CAPWAP LAG Support restrictions.

Ensure these requirements are met when using CAPWAP LAG support.

- APs must be specifically enabled for CAPWAP AP LAG support.
- CAPWAP data does not support IPv6.
- Do not enable Datagram Transport Layer Security (DTLS) when you enable LAG.
- CAPWAP LAG does not work with APs behind Network Address Translation (NAT) or Port Address Translation (PAT).

Enable CAPWAP LAG support on controller (GUI)

Enable CAPWAP LAG support on controller using GUI.

Enable LAG support for APs using the controller GUI.

Procedure

1. Choose **Configuration** > **Wireless** > **Wireless Global**.
2. Check the **AP LAG Mode** check box.

3. Click **Apply**.

Enable CAPWAP LAG support on controller (CLI)

Enable CAPWAP LAG support on controller using commands.

Enable LAG support for all APs on the controller using CAPWAP.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable CAPWAP LAG support on the controller.

   Example

   ```
   Device(config)# ap lag support
   ```

   Note

   After executing this command, this warning statement is displayed:

   Changing the lag support will cause all the APs to disconnect. Thus, all APs with LAG capability reboots and joins the enabled CAPWAP LAG.

3. Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

Enable and disable CAPWAP LAG globally on controller

Enable and disable CAPWAP LAG globally on controller.

If the CAPWAP LAG is enabled globally on the controller, these events occur:

- The AP joins the controller.
- The AP exchanges its CAPWAP support.
- LAG mode starts, if LAG is enabled on AP.

If the CAPWAP LAG is disabled globally on the controller, these events occur:

- The AP joins the controller.

- The AP exchanges its CAPWAP support.
- The AP receives the LAG configuration if LAG is already enabled on the AP.
- The AP reboots.
- The AP joins again with LAG disabled.

Enable CAPWAP LAG for an AP profile (GUI)

Enable CAPWAP LAG for an AP profile using GUI.

Enable LAG mode for a CAPWAP AP profile to improve reliability and throughput using GUI.

Procedure

1. Choose **Configuration** > **Tags (and) Profiles** > **AP Join**.
2. Click **Add**.
3. Under the **General** tab, enter the **Name** of the AP Profile and check the **LAG Mode** check box to set the CAPWAP LAG for the AP profile.
4. Click **Apply to Device**.

Enable CAPWAP LAG for an AP profile (CLI)

Enable CAPWAP LAG for an AP profile using commands.

Enable CAPWAP LAG on an AP profile to improve data throughput and provide redundancy for associated APs.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode. **ap profile** *ap-profile* .

   Example

   ```
   Device(config)# ap profile xyz-ap-profile
   ```

   Note

   When you delete a named profile, the APs associated with that profile do not revert to the default profile.

3. Enable CAPWAP LAG for an AP profile.

Example

```
Device(config-ap-profile)# lag
```

4. Exit configuration mode and return to privileged EXEC mode.

Example

```
Device(config-ap-profile)# end
```

Disable CAPWAP LAG for an AP profile (CLI)

Disable CAPWAP LAG for an AP profile using commands.

Disable CAPWAP LAG for a specified AP profile using commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure an AP profile and enter the AP profile configuration mode.

Example

```
Device(config)# ap profile xyz-ap-profile
```

> **Note**
>
> When you delete a named profile, the APs associated with that profile do not revert to the default profile.

3. Disable CAPWAP LAG for an AP profile.

Example

```
Device(config-ap-profile)# no lag
```

4. Exit configuration mode and return to privileged EXEC mode.

Example

```
Device(config-ap-profile)# end
```

Disable CAPWAP LAG support on controller

Disable CAPWAP LAG support on controller using commands.

Disable LAG support for the CAPWAP protocol using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Disable CAPWAP LAG support on the controller .

   Example

   ```
   Device(config)# no ap lag support
   ```

   Note

   All APs with LAG capability reboots and joins the disabled CAPWAP LAG.

3. Exit configuration mode and return to the privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

Verify CAPWAP LAG support configurations

Verify CAPWAP LAG support configurations.

To verify the global LAG status for all Cisco APs, use the command:

```
Device# show ap lag-mode

AP Lag-Mode Support Enabled
```

To verify the AP LAG configuration status, use the command:

```
Device# show ap name config general

Cisco AP Identifier : 0008.3291.6360
Country Code : US
Regulatory Domain Allowed by Country : 802.11bg:-A 802.11a:-AB
AP Country Code : US - United States
::
```

```
AP Lag Configuration Status : Enabled/Disabled
Has AP negotiated lag based on AP capability and per AP config
```

Configure bulk AP provisioning (GUI)

Configure bulk AP provisioning using GUI.

Use bulk AP provisioning to configure multiple parameters, such as admin status, floor location, geolocation, and high availability settings, for several APs at once.

Procedure

1. Choose **Configuration** > **Wireless** > **Bulk AP Provisioning**.

   View the current tasks and their status.

2. Click **Start a workflow to create an AP Provisioning task** to start a new bulk AP provisioning task.
3. In the **Select AP** page, configure the following:
   a) Change the name of the task.
   b) Select the APs you wish to provision.
   c) Click **Next**.
4. In the **Select Parameters** page, configure the following as required:
   a) Change the admin status using the drop-down list.
   b) Enter the location.
   c) Enter the above ground level height in meters. The range is from -100 to 1000.
   d) Enter the tolerance as uncertainty height in meters. The range is from zero to 100.
   e) Enter the cable length in meters. The range is from one to 100. The default is 10.

   > **Note**
   >
   > Keep the default value of 10 meters when using a Cisco provided external antenna. This option is available only on models that support external antennas.

   f) Enter the floor ID.
   g) Enter the name and management IP address of the primary, secondary, and tertiary controllers.
   h) Click **Next**.
5. In the **Summary** page, click **Apply** after confirming the changes.

   You can view the task status as it progresses from provisioning to completion. After selecting the task, you can view the number of configurations applied and not applied to each selected AP.

Activate tracking for deactivated infrastructure devices (GUI)

Use this procedure to initiate tracking for an inactive infrastructure device that has been deactivated. After the switch or AP becomes active, CUCM tracks its status and provides a list of endpoints that associate to the switch or AP.

Note

The controller synchronization process only adds new APs to the list and does not remove APs if they are not found.

To remove APs from the infrastructure list, mark them as inactive. After they are marked as inactive, navigate to the inactive page and delete the APs.

Because APs are stored independently in the database, controllers and APs are not directly associated. If you delete a controller, CUCM does not track AP or controller associations. APs learned from the controller stay in the system.

Before you begin

Configure Location Awareness. For more information, refer to the "Location Awareness" chapter in the *System Configuration Guide for Cisco Unified Communications Manager*.

Procedure

1. In Cisco Unified CM Administration, choose **Advanced Features** > **Device Location Tracking Services** > **Switches and Access Points**.
2. From **Related Links**, choose **Inactive Switches and Access Points** and click **Go**.
   The **Find and List Inactive Switches and Access Points** window displays the infrastructure devices that are not being tracked.
3. Select the switch or AP for which you want to initiate tracking.
4. Click **Reactivate Selected**.

CAPWAP message aggregation

Explains how CAPWAP message aggregates improve wireless network efficiency by consolidating multiple messages into a single transmission.

Control and provisioning of wireless APs message aggregation

This feature specifies details about CAPWAP message aggregation.

A CAPWAP message aggregation is a wireless communication optimization feature that

- combines multiple CAPWAP control messages of the same type into a single transmission
- temporarily stores messages in the controller when APs are busy processing packets, and
- limits the aggregated message size to the configured Maximum Transmission Unit (MTU) to maximize transmission efficiency.

Feature history
Table 45: Feature history for CAPWAP message aggregation

Feature Name	Release Information	Feature Description
CAPWAP Message Aggregation	Cisco IOS XE 17.14.1	This feature aggregates the CAPWAP control messages of the same type waiting in the queue to be transmitted to the AP.

Guidelines of CAPWAP message aggregation

Guidelines

- Applicable to all AP modes.
- The CAPWAP message aggregation feature is disabled by default.

Use cases of CAPWAP message aggregation

FlexConnect deployment

You can expect a round-trip delay when packets are sent over wide area network (WAN) in FlexConnect deployments.

- With the CAPWAP message aggregation, round-trip time decreases significantly.
- In addition, client join and roam operations complete more quickly.

Configure CAPWAP message aggregation (CLI)

The task file describes how to configure CAPWAP message aggregation in the controller console.

Enable CAPWAP message aggregation to optimize wireless network performance by reducing message overhead between APs and the controller.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile default-ap-profile
   ```

3. Enable CAPWAP message aggregation.

   Example

   ```
   Device(config-ap-profile)# capwap aggregation
   ```

This feature is disabled by default.

4. Exit the configuration mode and return to the privileged EXEC mode.

Example

```
Device(config-ap-profile)# end
```

Verify CAPWAP message aggregation

The reference file describes how to verify CAPWAP message aggregation in the controller console.

To view the total number of aggregated CAPWAP control packets for the controller, use the command:

```
Device# show wireless stats ap packet

Packet stats

Capwap Control Packets Received* : 11183016
Capwap Data Keep Alive Packets Received : 160399
Capwap Data DOT1X EAP Packets Received: 549
Capwap Data DOT1X Mgmt Packets Received: 6003
Capwap Data DOT1X Key Type Packets Received: 0
Capwap Data DOT1X Control Packets Received: 0
Capwap Data ARP Packets Received: 0
Capwap Data IP Packets Received: 0
Capwap Data IPV6 Packets Received: 0
Capwap Data RRM Packets Received: 0
Capwap Data DHCP Packets Received: 0
Capwap Data RFID Packets Received: 0
Capwap Data IAPP Packets Received: 2531939
Capwap Dgram Input Errors : 0
Capwap Discovery Packets Received : 22299
Capwap Discovery Dgram Input Errors : 0
Aggregated Capwap Control Packets Sent: 119337
**** Note: Capwap control packets exclude discovery/primary discovery packets
 ****
```

To verify the status of the CAPWAP message aggregation feature, use the command:

```
Device# show ap profile name default-ap-profile detailed

AP Profile Name                : default-ap-profile
Description                    : default custom profile
Country code                   : Not configured
Stats Timer                    : 180
Link Latency                   : ENABLED
Data Encryption                : DISABLED
LED State                      : ENABLED
NTP server                     : 0.0.0.0
NTP Authentication             : DISABLED
Jumbo MTU                      : ENABLED
24ghz Report Interval          : 90
5ghz Report Interval           : 90
bssid stats status             : ENABLED
bssid stats frqncy interval    : 120
bssid neighbor stats status    : ENABLED
```

```
bssid neighbor stats interval : 120
CAPWAP Control Aggregation    : ENABLED
```

33 Secure Data Wipe

Topics:

- Secure data wipe

Secure data wipe

Explains how the secure data wipe feature on Cisco access points erases sensitive files, stores wipeout status, and assists in verification and troubleshooting.

Secure data wipe is a feature that

- securely erases files from the file system of Cisco APs using the **clear ap config** command
- stores basic information about the wipeout and its status, and
- helps verify and troubleshoot erased files.

Types of files securely erased

These files are securely erased, without possibility of recovery:

- Configuration and backup configuration files
- Crash files
- Log files
- Boot variables
- Package logs

Feature history

Feature history for secure data wipe.

Table 46: Feature history for secure data wipe

Release	Feature	Feature Information
Cisco IOS XE Dublin 17.11.1	Secure data wipe	The Secure Data Wipe feature allows you to securely erase files from the file system of Cisco APs by using the **clear ap config** command.

Restrictions

Restrictions for secure wipe data.

Restriction for secure data wipe is as follows:

Data wipe for APs in Work Group Bridge mode can be done only through the mode button, which needs to be pressed for 20 to 60 seconds to remove storage content.

Supported AP models and software versions

Describes the supproted AP models and software versions.

Supported software versions

Cisco APs	Supported versions
Cisco Wave 1 APs	Cisco IOS XE 17.3, Cisco IOS XE 17.9.3 and its later versions, and Cisco IOS XE 17.12.
Cisco Wave 2 APs	Cisco IOS XE 17.11, Cisco IOS XE 17.12, and Cisco IOS XE 17.13.

Note

Cisco Wave 1 APs are not supported in 17.4, 17.5, 17.6, 17.7, 17.8, 17.10 and later releases.

Supported AP models

Cisco IOS APs	Cisco COS APs	Cisco Catalyst APs
3700 (I/E/P)	3800 (I/E/P)	9162I
		9164I
		9166I
		9166D1
		9163E
2700 (I/E)	2800 (I/E)	9124AX (I/D/E)
1700I	1815 (I/W)	9136I
702W	1840 (I)	9130AX (I/E)
1532I/E	1850 (I/E)	9120AX (I/E)
1570	1830 (I/W)	9117AXI
–	1810	9115AX (I/E)
–	1800I	9105AX (I/W)
–	1540	–
–	1560	–
–	4800	–

Verify data wipe

Verifies the data wipe.

To verify the output of the data wipe, run the **show flash wipeout-log** AP command:

```
Cisco-AP# show flash wipeout-log
DATA SANITATION LOGS
```

```
Filesystem Name    :    Flash
Filesystem size    :    519 M (bytes)
Total Files        :    95
Data Wipe Time     :    Fri Mar  8 09:50:49 UTC 2024
Data Wipe method   :    CLEAR
Files cleared      :    92
Bytes cleared      :    5484544 (bytes)
Total Free byte    :    458846208 (bytes)
Device PID         :    C9130AXI-E
Serial number      :    KWC233202MN
Data Wipe Status   :    SUCCESS
```

34 Wi-Fi 7 Operations

Topics:

- Wi-Fi 7 AP
- Per WLAN Wi-Fi 7 toggle

The details of the features supported by the Wi-Fi 7 APs are discussed in the following topics.

Wi-Fi 7 AP

Explains what Wi-Fi 7 APs are, their operating modes, and configurations for enterprise wireless networking.

Wi-Fi 7 APs are enterprise wireless networking devices that

- operate across tri-band frequencies (2.4 GHz, 5 GHz, and 6 GHz)
- support interoperability with 802.11ax/ac clients and hybrid deployments, and
- offer multiple radio and antenna modes for regulatory and deployment flexibility.

Cisco Wi-Fi 7 AP models

The Cisco Wireless 9176I Wi-Fi 7 AP, Cisco Wireless 9176D1 Wi-Fi 7 AP, Cisco Wireless 9178I Wi-Fi 7 AP, and Cisco Wireless 9172I Wi-Fi 7 AP, are an enterprise-class tri-band APs that operate across 2.4 GHz, 5 GHz, and 6 GHz frequencies. You can use the AP hardware with Cisco Catalyst Center, Cisco Catalyst stack, and Meraki cloud-based stack.

Operating modes and configuration

From the Cisco IOS-XE 17.15 release, split-PHY mode is supported in Cisco Wireless 9172I Series Wi-Fi 7 APs (CW9172I) radio. The radio on Cisco Wireless 9172I AP operates in the following modes:

- 5 GHz 4x4 (single-PHY)
- 5 GHz 2x2 + 6-GHz 2x2 (split-PHY)

The default mode is 5 GHz 2x2 or 6 GHz 2x2. If the AP operates in a regulatory domain where 6 GHz is not supported, it will operate in the 5 GHz 4x4 mode.

> **Note**
>
> If the AP radio power is off, then 802.11be capability of the radio is switched off. For the AP to be 802.11e capable, all radios have to be switched on.

Operating modes and configuration

From the Cisco IOS-XE 17.17 release, Cisco Wireless 9172H Series AP operates in two modes:

- 2.4 GHz(4x4) + 5 GHz(2x2) + 6 GHz(2x2)
- 2.4 GHz(4x4) + 5 GHz full band(4x4)

To access the 5 GHz full band mode(5 GHz radio in 4X4 mode):

- Configure antenna count as 4, under mapped radio profile.
- Disable the 6 GHz radio (slot 2) and reboot the AP.

If you do not reboot the AP, the 5 GHz radios will continue to operate in 2X2 mode even though the antenna count configuration is 4. The AP goes for a reboot when 6 GHz radio is enabled, as it requires a mode switch for the 6 GHz radio bring-up.

Multi-link operation (MLO)

Multi-link operation (MLO) details.

A multi-link operation (MLO) is a wireless networking feature that

- enables multi-link devices (MLDs) to maintain associations and transmit data across multiple frequency bands simultaneously including 2.4 GHz, 5 GHz, and 6 GHz bands
- integrates with the IEEE 802.11be (Wi-Fi 7) protocol to ensure dynamic network optimization, and
- adapts its functionality based on device power mode and protocol activation state.

When you enable the 802.11be (Wi-Fi 7) protocol on your device, MLO capabilities start automatically and network performance improves. If you disable the 802.11be protocol, MLO features will not be available.

Multi-link operation (MLO) modes

There are three widely adapted MLO modes.

- Multi-Link Single Radio (MLSR)
- Enhanced Multi-Link Single Radio (EMLSR)
- Multi-Link Multi Radio (MLMR)

Table 47: Feature history for multi-link operation (MLO)

Feature Name	Release Information	Feature Description
Multi-link operation (MLO)	Cisco IOS XE 17.18.1	Multi-Link Operation (MLO) is now supported even when radios are operating in low power mode. This enhancement ensures seamless multi-link operations under constrained power conditions, providing improved flexibility and efficiency for network operations. This feature is supported on the following access points only: • Cisco Wireless 9178 Series Access Points. • Cisco Wireless 9176 Series Access Points.

Feature Name	Release Information	Feature Description
Multi-link operation (MLO)	Cisco IOS XE 17.15.2	To alert users about possible disruptions, pop-up notifications or console informational messages will be displayed whenever there are changes to the radio parameters. These messages appear when the workflow is enabled or disabled, or when a radio reset occurs. The lack of support for link reconfiguration in Multi-Link Operation (MLO) may lead to client disruption during radio configuration changes. Disruption will occur for clients connected to an AP with 802.11be enabled radio, especially when 802.11be is enabled across all radio bands.

Multi-link statistics

Multi-link statistics details.

A multi-link statistics table is a data structure that

- tracks per-link and aggregated performance statistics for multi-link capable wireless clients
- uses the Multi-Link Device (MLD) client's MAC address as a key for identifying statistics, and
- maintains separate per-band statistics as clients connect to or roam between different APs and bands.

The table also provides per-band statistics for non-MLD stations. The table tracks statistics when a non-MLD station moves between bands. When a non-MLD client connects to an MLD AP, the table transfers the aggregated statistics to the per-band statistics section. When an MLD station connects to a non-MLD AP, the table preserves per-band statistics and continues accumulating aggregated statistics.

Note

In **Monitoring** > **Wireless** > **Clients** > **General** > **Clients** > **Statistics**, the display of client statistics has been updated. The statistics for Multi-Link Operation (MLO) are grouped by band instead of slot. When client statistics are recorded across multiple bands, the display shows a column for each band. Within each band, a client may be associated with one or more slots. For Non-MLO clients, the display shows aggregated data for the single band each client uses.

Configure 802.11be network parameters (CLI)

Configure 802.11be network parameters using CLI.

Set up 802.11be (Wi-Fi 7) network parameters on your device using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Disable the 802.11b/g network, if you use 2.4 GHz, 5 GHz or 6 GHz.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} shutdown
   ```

3. Configure the 802.11be network.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} dot11be
   ```

4. Enable the 802.11b/g network, if you use 2.4 GHz, 5GHz or 6GHz.

   Example

   ```
   Device(config)# no ap dot11 {24ghz | 5ghz | 6ghz} shutdown
   ```

Configure 802.11be network parameters (GUI)

Configure 802.11be network parameters using GUI.

Set up high throughput features for your wireless network by enabling and configuring 802.11be (Wi-Fi 7) options across supported frequency bands.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **High Throughput**.
2. Click the **6 GHz Band**, **5 GHz Band** or **2.4 GHz Band** tab.
   a) Check the check box to enable **802.11be**.
   b) In the **A-MPDU Tx** section, check the check boxes to set the desired priority levels.
   c) In the **A-MSDU Tx** section, check the check boxes to set the desired priority levels.
3. Click on the **5 GHz Band** tab and in the **A-MPTDU Tx** section, and complete these:
   a) Enter the value in the field to set the **Window Size**.
   b) Toggle the button to enable the **Aggregation Scheduler**.
   c) Select either **Automatic**, **Enabled**, or **Disabled** in the **Frame Burst** drop-down list.
   d) Enter the value in the **Realtime Traffic Timeout** field. The default value is 10.
4. Click the **2.4 GHz Band** tab.
5. In the **A-MPTDU Tx** section, complete these:
   a) Enter the value in the field to set the **Window Size**.

b) Toggle the button to enable the **Aggregation Scheduler**.
c) Enter the value in the **Realtime Traffic Timeout** field. The default value is 10.

6. Click **Apply**.

Configure A-MPDU window size in RF radio profile (GUI)

Configure A-MPDU window size in RF radio profile using GUI.

Change the A-MPDU window size to optimize wireless performance for an RF radio profile.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF/Radio**.

   The **RF** window is displayed with list of profiles.

2. Click **Add**.
   The **Add RF Radio Profile** window is displayed.
3. Enter an RF profile name in the **General** tab.
4. Click the **Advanced** tab.
5. In the **A-MPDU Window Size** field enter the value.

   The valid range is between one to 255. The default value is 255.

6. Click **Apply to Device**.

Configure A-MPDU window size in RF radio profile (CLI)

Describes the CLI configuration steps for compressed block-ack window.

Set the compressed block acknowledgment (block-ack) window size for 802.11 transmit aggregated MAC protocol data units (A-MPDU) in a specific RF profile.

Procedure

1. Enter the global configuration mode.

   Example

```
Device# configure terminal
```

2. Configure an RF profile for 2.4 GHz, 5 GHz, or the 6 GHz band.

   Example

```
Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rf-profile block-ack-rf-profile
```

3. Configure 802.11 transmit A-MPDU window size.

Example

```
Device(config-rf-profile)# dot11 a-mpdu tx block-ack window-size 255
```

Enable or disable client serving on AP (GUI)

Enable or disable client serving on AP using GUI.

Use this task to enable or disable client serving on an AP from the GUI.

Before you begin

Change the AP mode to **Client Serving**.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click the AP name from the 6 GHz, 5 GHz, or 2.4 GHz list on the **Access Points** page.
3. Select the **Assignment Method** as *Client Serving* in the **Role Assignment** section.

   Note

   You can edit this section only if the **Assignment Method** is set to **Custom**.

4. Configure these in the **RF Channel Assignment** section:
   - Select the Channel Width from the **RF Channel Width** drop-down list.
   - Select the type of assignment from the **Assignment Method** drop-down list.

   Note

   If you choose **Custom**, you must select a channel width and specify an RF channel number to the AP radio. A 320 MHz channel width is supported from the Cisco IOS XE 17.15.1 onwards.

5. Click **Update & Apply to Device**.

Configure 802.11be MCS rates (CLI)

Describes the CLI configuration steps for 802.11be MCS rates.

Configure the modulation and coding scheme (MCS) transmission rates for 802.11be (Wi-Fi 7) APs using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the 802.11be parameters.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} dot11be
   ```

3. Enable the 802.11be 2.4 GHz, 5 GHz, or 6 GHz band modulation and coding scheme (MCS) transmission rates.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} dot11be mcs tx index {11 | 13
    | 14 | 15 | 9} spatial-stream spatial-stream-value
   ```

Configure 802.11be radio parameter frequencies (GUI)

Configure 802.11be radio parameter frequencies using GUI.

Set data rates and enable 802.11be features on supported frequency bands.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **High Throughput**.
2. Choose the **6 GHz Band** or **5 GHz Band** or **2.4 GHz Band** tab.
   a) Expand the **11be** section.
   b) Check the **Enable 11be** check box, if required.
   c) Check either the **Select All** check box to configure all the data rates or select the desired options from the available data rates list.
3. Click **Apply**.

Configure multi BSSID 802.11be parameters (GUI)

Configure multi BSSID 802.11be parameters using GUI.

Configure advanced 802.11be parameters within a multi BSSID profile to manage Orthogonal frequency-division multiple access (OFDMA) and MU-MIMO settings.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Multi BSSID**.
2. Click **Add**.

The **Add Multi BSSID Profile** window is displayed.

3. Enable or disable the following parameters in the **802.11be** section:
   - **OFDMA Downlink**
   - **OFDMA Uplink**
   - **MU-MIMO Downlink**
   - **MU-MIMO Uplink**
   - **OFDMA Multi-RU**

   Note

   OFDMA Downlink and **OFDMA Uplink** are enabled by default.

4. Click **Apply to Device**.

Configure multi BSSID 802.11be parameters (CLI)

Describes the CLI steps to configure 802.11be parameters in the multi-BSSID profile.

Note

Multi-BSSID profile configuration overwrites the 802.11be profile configuration for 6-GHz band.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the multi BSSID profile and enter the multi BSSID profile configuration.

   Example

   ```
   Device# wireless profile multi-bssid default-multi-bssid-profile
   ```

3. Configure the 802.11be parameters.

Example

```
Device# dot11be {downlink-mumimo | downlink-ofdma | ofdma-multi-ru |
uplink-mumimo | uplink-ofdma}
```

Configure 802.11be parameters (GUI)

Configure 802.11be parameters using GUI.

Modify 802.11be profile parameters to optimize wireless performance for specific user profiles.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **802.11be**.

   The different user profiles are listed.

2. Click the required profile.

   The **Edit 802.11be Profile** window is displayed.

3. Click to **Enable** or **Disable** the **OFDMA Downlink**, **OFDMA Uplink**, **MU-MIMO Downlink**, **MU-MIMO Uplink**, and **OFDMA Multi-RU** in the **Edit 802.11be Profile** window.
4. Click **Update and Apply to Device**.

Monitor 5 GHz radio WLAN slot data (GUI)

Monitor 5 GHz radio WLAN slot data using GUI.

View and confirm the WLAN ID, BSSID MAC, and MLD MAC information for APs operating on the 5 GHz radio.

Procedure

1. Choose **Monitoring** > **Wireless** > **Radio Statistics**.
2. Click the **5 GHz Radios** tab, then select the required AP.
3. Verify the **WLAN ID**, **BSSID MAC**, and **MLD MAC** in the **WLAN Slot Data** section.

   The **5 GHz Band** window is displayed.

4. Click **OK**.

Monitor 802.11be radio parameters (GUI)

Monitor 802.11be radio parameters using GUI

View and verify preamble puncturing parameters for 802.11be radios in the wireless monitoring dashboard.

Procedure

1. Choose **Monitoring** > **Wireless** > **Radio Statistics**.
2. Click the **6 GHz Radios** tab and select the required AP.

3. Verify these to monitor preamble puncturing in the **11be Parameters** section:
   a) **Puncture Reason**
   b) **Punctured Channels**
4. Click **OK**.

Verify multi-link statistics

Verify multi-link statistics details.

Verify wireless client MAC address details

To verify the wireless client mac-address mobility history, use the command:

```
Device# #show wireless client mac-address e460.17bb.3a50 mobility history

Recent association history (most recent on top):

AP Name                                       BSSID           AP Slot    Assoc
 Time               Instance   Mobility Role   Run Latency (ms)     Dot11 Roam
 Type
-------------------------------------------------------------------------------
WiFi7_AP1_2036                                c614.a26e.c218  1
11/21/2024 18:15:20      0          Local           4001                N/A
```

To verify the wireless client mac-address, use the command:

```
Device# show wireless client mac-address e460.17bb.3a50 detail

Client MAC Address : e460.17bb.3a50
Client MAC Type : Universally Administered Address
Client DUID: NA
Client IPv4 Address : 198.51.100.10
Client IPv6 Addresses : fe80::8359:7407:bb2b:ee9e
                        2001:DB8::1
                        2001:DB8::2
                        2001:DB8::3
Client Username: N/A
AP MAC Address : c414.a26e.c200
AP Name: WiFi7_AP1_2036
AP slot : 1
Client State : Associated
Policy Profile : LOCAL_PP_1
Flex Profile : N/A
Multilink Client : Yes
Wireless LAN Id: 91
WLAN Profile Name: W7_WPA3_SAE
Wireless LAN Network Name (SSID): W7_WPA3_SAE
BSSID : c614.a26e.c218
Connected For : 259 seconds
Protocol : 802.11be
Channel : 36
Client IIF-ID : 0xa0000002
Association Id : 33
Authentication Algorithm : Simultaneous Authentication of Equals (SAE)
Idle state timeout : N/A
Session Timeout : 3000 sec (Remaining time: 2745 sec)
Session Warning Time : Timer not running
Input Policy Name  : None
```

```
Input Policy State : None
Input Policy Source : None
Output Policy Name  : None

Output Policy State : None
Output Policy Source : None
WMM Support : Enabled
U-APSD Support : Disabled
Fastlane Support : Disabled
Client Active State : Active
Power Save : OFF
Current Rate : m12 ss2
Supported Rates : 6.0,9.0,12.0,18.0,24.0,36.0,48.0,54.0
AAA QoS Rate Limit Parameters:
  QoS Average Data Rate Upstream             : 0 (kbps)
  QoS Realtime Average Data Rate Upstream    : 0 (kbps)
  QoS Burst Data Rate Upstream               : 0 (kbps)
  QoS Realtime Burst Data Rate Upstream      : 0 (kbps)
  QoS Average Data Rate Downstream           : 0 (kbps)
  QoS Realtime Average Data Rate Downstream  : 0 (kbps)
  QoS Burst Data Rate Downstream             : 0 (kbps)
  QoS Realtime Burst Data Rate Downstream    : 0 (kbps)
Mobility:
  Move Count                  : 0
  Mobility Role               : Local
  Mobility Roam Type          : None
  Mobility Complete Timestamp : 11/21/2024 18:15:23 UTC
Client Join Time:
  Join Time Of Client : 11/21/2024 18:15:20 UTC
Client State Servers : None
Client ACLs : None
Policy Manager State: Run
Last Policy Manager State : IP Learn Complete
Client Entry Create Time : 256 seconds
Policy Type : WPA3
Encryption Cipher : CCMP (AES)
Authentication Key Management : SAE
AAA override passphrase : No
SAE PWE Method : Hash to Element(H2E)
Transition Disable Bitmap : 0x00
Beacon Protection : No
User Defined (Private) Network : Disabled
User Defined (Private) Network Drop Unicast : Disabled
Encrypted Traffic Analytics : No
Protected Management Frame - 802.11w : Yes
EAP Type : Not Applicable
VLAN Override after Webauth : No
VLAN : 129
Multicast VLAN : 0
VRF Name : N/A
WiFi Direct Capabilities:
  WiFi Direct Capable           : No
Central NAT : DISABLED
Session Manager:
  Point of Attachment : capwap_90000019
  IIF ID              : 0x90000019
  Authorized          : TRUE
  Session timeout     : 3000
  Common Session ID: 1E800709000000384FEF901E
  Acct Session ID  : 0x00000000
  Auth Method Status List
        Method : SAE
```

```
  Local Policies:
        Service Template : wlan_svc_LOCAL_PP_1_local (priority 254)
                VLAN            : 129
                Absolute-Timer   : 3000
  Server Policies:
  Resultant Policies:
                VLAN Name        : VLAN0129
                VLAN             : 129
                Absolute-Timer   : 3000
DNS Snooped IPv4 Addresses : None
DNS Snooped IPv6 Addresses : None
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 0
Fast BSS Transition Details :
  Reassociation Timeout : 0
11v BSS Transition : Implemented
11v DMS Capable : No
QoS Map Capable : No
FlexConnect Data Switching : N/A
FlexConnect Dhcp Status : N/A
FlexConnect Authentication : N/A
Client Statistics:
  Number of Bytes Received from Client : 137038
  Number of Bytes Sent to Client : 39468
  Number of Packets Received from Client : 1064
  Number of Packets Sent to Client : 237
  Number of Data Retries : 0
  Number of RTS Retries : 0
  Number of Tx Total Dropped Packets : 0
  Number of Duplicate Received Packets : 0
  Number of Decrypt Failed Packets : 64
  Number of Mic Failured Packets : 0
  Number of Mic Missing Packets : 0
  Number of Policy Errors : 0
  Radio Signal Strength Indicator : -46 dBm
  Signal to Noise Ratio : 51 dB
  Last statistics update : 11/21/2024 18:18:18
  Number of links dropped due to downsizing : 0
  Client Statistics per band:
    Radio Band 5GHz
    Station Link MAC Address: e660.17bb.3a53
      AP slot : 1
      Number of Bytes Received from Client : 137038
      Number of Bytes Sent to Client : 39468
      Number of Packets Received from Client : 1064
      Number of Packets Sent to Client : 237
      Number of Data Retries : 0
      Number of RTS Retries : 0
      Number of Tx Total Dropped Packets : 0
      Number of Duplicate Received Packets : 0
      Number of Decrypt Failed Packets : 64
      Number of Mic Failured Packets : 0
      Number of Mic Missing Packets : 0
      Number of Policy Errors : 0
      Radio Signal Strength Indicator : -46 dBm
      Signal to Noise Ratio : 51 dB
      Last statistics update : 11/21/2024 18:18:18
```

```
    Radio Band 2.4GHz
    Station Link MAC Address: e660.17bb.3a54
      AP slot : 0
      Number of Bytes Received from Client : 0
      Number of Bytes Sent to Client : 0
      Number of Packets Received from Client : 0
      Number of Packets Sent to Client : 0
      Number of Data Retries : 0
      Number of RTS Retries : 0
      Number of Tx Total Dropped Packets : 0
      Number of Duplicate Received Packets : 0
      Number of Decrypt Failed Packets : 0
      Number of Mic Failured Packets : 0
      Number of Mic Missing Packets : 0
      Number of Policy Errors : 0
      Radio Signal Strength Indicator : Unavailable
      Signal to Noise Ratio : Unavailable
      Last idle time update : Never
Fabric status : Disabled
Radio Measurement Enabled Capabilities
  Capabilities: Neighbor Report, Passive Beacon Measurement, Active Beacon
Measurement, Table Beacon Measurement
Client Scan Report Time : Timer not running
Client Scan Reports
Assisted Roaming Neighbor List
Nearby AP Statistics:
EoGRE : Pending Classification
Max Client Protocol Capability: Wi-Fi 7 (802.11be)
WiFi to Cellular Steering : Not implemented
Cellular Capability : N/A
Advanced Scheduling Requests Details:
  Apple Specific Requests(ASR) Capabilities/Statistics:
    Regular ASR support: DISABLED
L3 Access : DISABLED
Client Gateway IPv4 Address : 198.51.100.1
Multi Link Information:
  Enhanced Multi Link    : EMLSR
  STR Capable            : No
  No. of associated links: 2
  Associated Links
    Band : 5 GHz
      MAC Address: e660.17bb.3a53
      BSSID: c414.a26e.c207
    Band : 2.4 GHz
      MAC Address: e660.17bb.3a54
      BSSID: c414.a26e.c208
Known Link Information:
  No. of Known links: 2
  Known Links
    Band : 5 GHz
      MAC Address: e660.17bb.3a53
    Band : 2.4 GHz
      MAC Address: e660.17bb.3a54
```

Verify wireless client summary

To verify the wireless client summary, use the command:

```
Device# show wireless client summary
Number of Clients: 1
```

```
MAC Address    AP Name                                          Type ID   State
             Protocol   Method     Role
-------------------------------------------------------------------------------
e460.17bb.3a50 WiFi7_AP1_2036                                   WLAN 91   Run
             11be(MLO)  SAE        Local
```

Verify wireless summary

To verify the wireless summary, use the command:

```
Device# show wireless summary

Priming controller         : DISABLED
Max APs supported          : 2000
Max clients supported      : 32000

Access Point Summary

                       Total    Up    Down
-----------------------------------------
802.11 2.4GHz              1     1       0
802.11 5GHz                3     1       2
802.11 6GHz                1     0       1
802.11 dual-band           1     0       1
802.11 dual-band(5/6GHz)   0     0       0
802.11 rx-dual-band        0     0       0

Client Serving(2.4GHz)     2     1       1
Client Serving(5GHz)       3     1       2
Client Serving(6GHz)       1     0       1
Monitor(Dual band)         0     0       0
Monitor(2.4GHz)            0     0       0
Monitor(5GHz)              0     0       0
Monitor(6GHz)              0     0       0
Sniffer(Dual band)         0     0       0
Sniffer(2.4GHz)            0     0       0
Sniffer(5GHz)              0     0       0
Sniffer(6GHz)              0     0       0

Client Summary

  Total Clients : 1
  Excluded      : 0
  Disabled      : 0
  Foreign       : 0
  Anchor        : 0
  Local         : 1
```

Verify 802.11be parameters

Verify 802.11be parameters details.

Verify 802.11be profile details

To display the 802.11be profile details, run the command:

```
Device# show dot11be profile detailed
Dot11be profile name : sample_11be_profile
-----------------------------------------------------------
Description : sample_11be_profile
```

```
802.11be parameters
OFDMA Downlink : Enabled
OFDMA Uplink : Enabled
OFDMA Multi-RU : Enabled
MU-MIMO Downlink : Enabled
MU-MIMO Uplink : Enabled
```

Verify 802.11 profile summary

To display the 802.11be profile summary, run the command:

```
Device# show dot11be profile summary
Number of dot11be profiles: 1
Dot11be profile name              Description
------------------------------------------------------------------------
default-dot11be-profile           Default 802.11be profile
```

Verify 320 MHz details (GUI)

Verify 320 MHz details using GUI.

> **Note**
>
> - 320 MHz is supported only on the 6-GHz band.
> - By default, the Dynamic Channel Assignment (DCA) maximum bandwidth is 160 MHz. Users need to enable 320 MHz in the 6-GHz RF Profile to utilize it.

Confirm that 320 MHz operation is enabled and functioning on the 6 GHz band by reviewing channel request and response data in the GUI.

Procedure

1. Choose **Monitoring** > **Wireless** > **AP Statistics**.
2. Select **General**.

   A list of APs under the **General** tab is displayed.
3. Select the required profile from the list.

   A **General** window is displayed.
4. Select **AFC** and then choose **Request**.
5. Verify the **320 MHz Channel Request Data** from the **Channel Request Data** table.
6. Select **AFC** and then choose **Response**.
7. Verify the **320 MHz Channel Response Data** from the **Channel Response Data** table.

Verify multi BSSID details

Verify multi BSSID details.

Verify multiple link device (MLD) MAC details

To display the 802.11be profile details, run the command:

```
Device# show ap name cisco-ap wlan dot11 5ghz
-------------------------------------------------------------
Slot id : 1
WLAN ID BSSID MLD
-------------------------------------------------
9 0aef.2500.010f 0aef.2501.0001
2 0aef.2500.010e 0aef.2501.0002
3 0aef.2500.010d 0aef.2501.0003
4 0aef.2500.010c 0aef.2501.0004
12 0aef.2500.010b 0aef.2501.0005
14 0aef.2500.010a 0aef.2501.0006
16 0aef.2500.0109 NA
```

Verify multi BSSID set ID and Tx VAP indicator details for 6 GHz radio

To display the 802.11be profile details, run the command:

```
Device# show ap name WSIM-AP-0001 wlan dot11 6ghz
-------------------------------------------------------------
Slot id : 3
WLAN ID BSSID MLD MBSSID Set ID
-------------------------------------------------------------------------------
9 0aef.2500.0108 0aef.2501.0001 1*
2 0aef.2500.0109 0aef.2501.0002 1
3 0aef.2500.010a 0aef.2501.0003 1
4 0aef.2500.010b 0aef.2501.0004 1
12 0aef.2500.010c 0aef.2501.0005 NA (11be WLAN not 6GHz compliant)
14 0aef.2500.010d 0aef.2501.0006 2*
16 0aef.2500.010e NA 3* (11ax 6GHz WLAN)
18 0aef.2500.010f NA NA (11ax non-6GHz WLAN)
```

Per WLAN Wi-Fi 7 toggle

Explains how Wi-Fi 7 options can be enabled or disabled per WLAN in network management platforms.

Feature history for per WLAN Wi-Fi 7 toggle

Per WLAN Wi-Fi 7 toggle feature history.

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 48: Feature history for access point upgrade management enhancements

Release	Feature Information
Cisco IOS XE 17.18.1	Per-WLAN 802.11be configuration is an enhancement in WLAN configurations that provides: • WLAN level control for 802.11be (Wi-Fi 7) configurations. • The ability to enable or disable 802.11 be (Wi-Fi 7) on a per WLAN basis. • Support for running both Wi-Fi 7 and non-Wi-Fi 7 WLANs simultaneously on the same radio frequency band.

Per-WLAN 802.11be configuration

This topic describes the per-WLAN 802.11be configuration.

Per-WLAN 802.11be configuration is an enhancement in WLAN configurations.

- It introduces WLAN-level control for 802.11be configurations.
- This capability supports both 802.11be and non-802.11be WLANs in the same radio frequency band.
- The configuration offers greater flexibility compared to previous versions.

Starting with Cisco IOS XE 17.18.1 release, you can enable or disable Wi-Fi 7 (802.11be) on a per WLAN basis. This change allows you to run both Wi-Fi 7 and non-Wi-Fi 7 WLANs simultaneously on the same radio band. It offers more flexibility compared to previous releases, where Wi-Fi 7 was configured as a global radio setting.

Upon upgrading to release 17.18.1, a default 802.11be profile is automatically created. This profile enables 802.11be on all bands (2.4 GHz, 5 GHz primary and secondary, and 6 GHz). You can configure custom 802.11be profiles starting from release 17.18.1.

> ⚠ **Caution**
>
> Modifying the default 802.11be profile affects the configuration across all bands where it is enabled.

Supported WLANs

This topic describes the supported WLANs.

WLANs with below security configuration combinations can operate in Wi-Fi 7 mode:

- WPA3-SAE-EXT-KEY/WPA3-SAE-EXT-KEY-FT + WPA3-SAE/WPA3-SAE-EXT + GCMP256 + AES
- WPA3-802.1x-SHA256/WPA3-SuiteB-192 + WPA2-802.1x + AES/GCMP256
- PMF Required/Optional

> **Note**
>
> WLANs that are not compatible with Wi-Fi 7 security restrictions are automatically downgraded to the 802.11ax mode. This process ensures that these WLANs remain operational even when Wi-Fi 7 is enabled globally.

Supported security configurations for 802.11be

This topic describes the supported security configurations for 802.11be.

Table 49: Security configurations

Security combination	AKM/Cipher	PMF (802.11w required/optional)	Beacon
WPA3-OWE + GCMP256	OWE + GCMP256 + AES	Yes	MLO + EHT(11be); No OWE element
WPA3 + WPA2	• 802.1x + 802.1x-sha256 • 802.1x + suite-b-192 • 802.1x-FT + 802.1x-sha256	Yes	MLO + EHT
WPA3	• 802.1x-sha256, 802.1x-FT • Suite B - 192 Bit Security • SAE-EXT/SAE-EXT-FT + GCMP256 + AES	Yes	MLO + EHT
WPA3+WPA2	SAE/SAE-FT+H2E + AES + SAE-EXT/SAE-EXT-FT + GCMP256	Yes	MLO + EHT
WPA3 OWE Transition Mode	• OWE WLAN: OWE + GCMP256 • OWE WLAN: OWE+AES+GCMP256	Yes	MLO + EHT
Wi-Fi 7 compliant security + non-compliant security	Wi-Fi 7 compliant security + non-compliant WPA3 security	Yes	MLO + EHT
WPA3 OWE	OWE + AES/GCMP256	Yes	MLO + EHT

> **Note**
>
> OWE transition WLAN(Wi-Fi Enhanced Open Transition Mode) configuration is not supported for Wi-Fi 7(802.11be) WLANs.

Beacon-Protection status

This topic describes Beacon-Protection status.

If 802.11be is enabled and WLAN security meets the 802.11be requirement, Beacon-Protection is auto-enabled.

> **Note**
>
> Beacon-Protection is a mandatory configuration for Wi-Fi 7(802.11be) WLANs.

MBSSID

MBSSID SET-ID

This topic describes the MBSSID SET-ID.

WLANs are arranged into compatible MBSSID (Multiple Basic Service Set Identifier) sets. The placement of each WLAN in an MBSSID set determines the VAP ID or BSSID MAC address. The 6-GHz band supports up to 16 WLANs. Therefore, four MBSSID sets with four WLANs in each group is automatically created. In each group, one WLAN is the Tx-VAP and the other three are non-Tx-VAP.

The grouping is done based on 802.11be against 802.11ax profile WLANs and whether it is a GCMP-128 against GCMP-256 cipher WLAN.

MBSSID grouping for 802.11be and non 802.11be deployments

This topic describes MBSSID grouping for 802.11be and non 802.11be deployments.

MBSSID grouping for non 802.11be deployments:

The configuration options when grouping WLANs into an MBSSID set are:

- 6 GHz policy (applicable or not applicable).
- Beacon protection state.
- Configure the cipher to use either 128-bit or 256-bit encryption when beacon protection is enabled.

MBSSID grouping for 802.11be deployments:

The configuration options when grouping WLANs into an MBSSID set are:

- 6 GHz policy (applicable or not applicable).
- The PHY mode for the 6 GHz slot in the corresponding 802.11be profile is either 802.11be or 802.11ax.
- Configure the cipher to use either 128-bit or 256-bit encryption.

> **Note**
>
> Beacon-protection is enabled for 802.11be deployments.

Disabled - no MBSSID slot

This topic describes the disabled - no MBSSID slot.

The WLAN is disabled in the 6 GHz band because there is no available MBSSID slot to accommodate it. This occurs due to MBSSID limitations or conflicting configurations. If the controller fails to map a WLAN into any of the four groups, it is marked as Disabled on the 6 GHz band by the system.

> **Note**
>
> This change does not affect the 2.4 GHz or 5 GHz bands.

Tx-VAP WLAN of MBSSID set for 6 GHz band

This topic describes Tx-VAP WLAN of MBSSID set for 6 GHz band.

The VAP within an MBSSID set transmits beacon frames for the group. In 802.11ax and 802.11be networks, multiple WLANs can be grouped into an MBSSID set to increase efficiency. The Tx-VAP sends beacon frames for the entire set. An asterisk (*) identifies the VAP that transmits beacon frames for the group.

The WLAN per-link configuration determines the VAP mode when 802.11be is enabled at the radio band level.

These conditions confirm the BSSID type as 802.11be or non 802.11be:

- If 802.11be is enabled at both the band and WLAN levels, the BSSID is 802.11be.
- If 802.11be is enabled at the band level but disabled at the WLAN level, the BSSID is non-802.11be.
- If 802.11be is disabled at the band level, all BSSIDs on that radio are non-802.11be, regardless of the WLAN per-link configuration.

Create a new 802.11be profile (GUI)

This topic describes how to create a new 802.11be profile using GUI.

Create a new 802.11be profile to to enable Wi-Fi 7 (802.11be) configuration using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **802.11be**.
2. Click **Add**.
3. In the **Add 802.11be Profile** page, add the **Profile Name** and **Description**.
4. In the **MLO group** section, use the toggle button to either enable or disable the **Primary 2.4GHz**, **Primary 5GHz**, **Secondary 5GHz** and **Primary 6GHz** bands.
5. Click **Apply to Device**.

Map the 802.11be profile to a WLAN (GUI)

This topic describes how to map the 802.11be profile to a WLAN using GUI.

Map the new 802.11be profile to a specific WLAN.

Procedure

1. Choose **Configuration** > **Tags and Profiles** > **WLANs**.
2. Select the WLAN and click the **Advanced** tab.
   The **Advanced** tab is displayed.
3. Scroll down to the **11be** section.
4. From the **802.11be Profile** drop-down list, select the required WLAN.
5. Click **Update & Apply to Device**.

Create and map 802.11be profile (CLI)

This topic describes how to create and map 802.11be profile using the CLI.

Create a 802.11be profile using the CLI.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure and name the wireless 802.11be profile.

   Example

   ```
   Device(config)# wireless profile dot11be 802.11be-prof-name
   ```

3. Configure MLO group settings for each radio link.

   Example

   ```
   Device(config-wireless-dot11be-profile)# mlo-group {24ghz | 5ghz | 5ghz-sec
   | 6ghz}
   ```

   The options are:

   - **24ghz**: Enable 2.4 GHz in MLO group.
   - **5ghz**: Enable 5 GHz in MLO group.
   - **5ghz-sec**: Enable secondary 5 GHz in MLO group.
   - **6ghz**: Enable 6 GHz in MLO group.

> **Note**
>
> Use the **no** form of this command to disable MLO group settings for each radio link, that is for 2.4 GHz, 5 GHz, secondary 5 GHz, and 6 GHz radio links, respectively.
>
> ```
> Device(config-wireless-dot11be-profile)# mlo-group {no-24ghz | no-5ghz  |
>  no-5ghz-sec | no-6ghz}
> ```

4. Exit to global configuration mode.

   Example

   ```
   Device(config-wireless-dot11be-profile)# exit
   ```

Map the 802.11be Profile to a WLAN

Map the 802.11be profile using the CLI.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN profile.

   Example

   ```
   Device(config)# wlan wlan-profile-name
   ```

   This command configures the WLAN profile and enters the WLAN configuration mode.

3. Disable the WLAN profile.

   Example

   ```
   Device(config-wlan)# shutdown
   ```

4. Configure 802.11be profile for this WLAN.

   Example

   ```
   Device(config-wlan)# [no] dot11be-profile dot11be-prof-name
   ```

   Use the **no** form of this command to disable this feature.

Verify 802.11be profile details

This topic describes how to verify 802.11be profile details.

To verify 802.11be profile details, run the **show wireless profile dot11be summary** command.

```
Device# show wireless profile dot11be summary
Number of dot11be profiles: 7

Dot11be profile name              Description
------------------------------------------------------------------------
6ghz@disable@11be                 disabling 11be on 6ghz band
24ghz_disable_11be                disabling 11be on 2.4GHz band
all^bands^11be^disable            disabling 11be on all bands
5ghz!slot2!disable!11be           disabling 11be on slot 2
5ghz*slot1*disable*11be
default-dot11be-profile           Default 802.11be profile
1234567890abcdefghijklmnopqrstuv
```

Verify 802.11be profile parameters with MLO mapping

This topic describes how to verify 802.11be profile parameters with MLO mapping.

To verify 802.11be profile parameters with MLO mapping, run the **show wireless profile dot11be detailed** *11be-prof-name* and **show wireless profile dot11be detailed***24ghz_disable_ 11be* commands.

```
Device# show wireless profile dot11be detailed 11be-prof-name
Device# show wireless profile dot11be detailed 24ghz_disable_11be


Dot11be profile name           : 24ghz_disable_11be
-------------------------------------------------
Description                    : disabling 11be on 2.4GHz band
802.11be parameters
    OFDMA Downlink             : Enabled
    OFDMA Uplink               : Enabled
    MU-MIMO Downlink           : Enabled
    MU-MIMO Uplink             : Enabled
    OFDMA Multi-RU             : Enabled
MLO Group
    24ghz link                 : Disabled
    5ghz link                  : Enabled
    5ghz secondary link        : Enabled
    6ghz link                  : Enabled
```

Verify 802.11be profile mapping to WLAN

This topic describes how to verify 802.11be profile mapping to WLAN.

To verify 802.11be profile mapping to WLAN, run the **show wireless profile dot11be detailed** *6ghz@disable@11be* command.

```
Device# show wireless profile dot11be detailed 6ghz@disable@11be
Dot11be profile name           : 6ghz@disable@11be
-------------------------------------------------
Description                    : disabling 11be on 6ghz band
802.11be parameters
    OFDMA Multi-RU             : Enabled
MLO Group
    24ghz link                 : Enabled
```

```
    5ghz link                  : Enabled
    5ghz secondary link        : Enabled
    6ghz link                  : Disabled
```

Verify VAP status for 802.11ax and 802.11be WLAN on 6 GHz band

This topic describes how to verify VAP status for 802.11ax and 802.11be WLAN on 6 GHz band.

To verify VAP status for 802.11ax and 802.11be WLAN on 6 GHz band, run the **show wireless tag policy detailed** *11ax-prof-name-policy-tag* **vap-id** command.

This table refers the MBBSID grouping for a 802.11ax AP (non 802.11be deployment).

```
Device #show wireless tag policy detailed 11ax-prof-name-policy-tag vap-id

WLAN-ID  WLAN profile name                  Policy profile name
VAP-ID  status    SET-ID  Beacon-Protection status  operational state
-------------------------------------------------------------------------------------------
1            open                          default-policy-profile            2
      Enabled   1*      Enabled (128bit cipher)   Enabled
4        wpa3sae                           policy-profile             1
  Enabled   1       Enabled (128bit cipher)   Enabled
5        wpa3owe                           policy-profile             3
  Enabled   1       Disabled                  Enabled
15       mbssid-1                          policy-profile             4
  Enabled   1       Disabled                  Enabled
12       mbssid-8                          default-policy-profile
6       Enabled   2*      Enabled (256bit cipher)   Enabled
10       mbssid-6                          policy-profile             5
  Enabled   2       Enabled (256bit cipher)   Enabled
7        mbssid-3                          default-policy-profile
7       Enabled   2       Disabled                  Enabled
9        mbssid-5                          policy-profile             8
  Enabled   2       Disabled                  Enabled
13       mbssid-9                          default-policy-profile
9       Enabled   3*      Disabled                  Enabled
14       mbssid-10                         policy-profile             12
  Enabled   3       Disabled                  Enabled
111      mbssid-a                          policy-profile             10
  Enabled   3       Disabled                  Enabled
112      mbssid-b                          default-policy-profile
11      Enabled   3       Disabled                  Enabled
17       PSKWLAN                           policy-profile             13
  Enabled   0       N/A                       N/A
8        mbssid-4                          default-policy-profile
14      Enabled   0       N/A                       N/A
6        mixed-mode                        policy-profile             15
  Enabled   0       N/A                       N/A

RLAN-ID  Remote-LAN profile name          Policy profile name
PORT-ID   VAP-ID
-------------------------------------------------------------------------------------------
16385    rlan_profile               rlan*policy                1        16
```

Note

***** refers to Tx-VAP WLAN of MBSSID set and 0 refers to invalid MBSSID set-id. WLAN is not broadcast in the 6-GHz band.

Verify Policy Tag for 802.11be WLAN VAP Status on 6 GHz

To verify Policy Tag detail view to check the VAP status for 802.11be WLAN status on 6 GHz band, run the **show wireless tag policy detailed** *11be-prof-name-policy-tag* **vap-id** command.

This table refers to the MBBSID grouping for a 802.11be AP (802.11be deployment). The WLANs downgraded to 802.11ax do not adhere to the 802.11be security standards. The WLAN with no MBSSID slot is not shown in the 6 GHz but only in the 2.4 GHz & 5 GHz.

```
Device #show wireless tag policy detailed 11be-prof-name-policy-tag vap-id

              |-------------------------------------------------------------|

                |                        6GHz VAP information
 |

              |-------------------------------------------------------------|

         WLAN     MBBSID                                          operational

WLAN-ID  WLAN profile name                Policy profile name
VAP-ID  status    SET-ID  PHY mode  Beacon-Protection status     state

-------------------------------------------------------------------------------------------------------
4        wpa3sae                          policy-profile              1
  Enabled   1*      802.11ax  Enabled (128bit cipher)   Enabled (11ax
downgraded)
5        wpa3owe                          policy-profile              2
  Enabled   1       802.11ax  Disabled                  Enabled (11ax
downgraded)
15       mbssid-1                         policy-profile              3
  Enabled   1       802.11ax  Disabled                  Enabled (11ax
downgraded)
9        mbssid-5                         policy-profile              4
  Enabled   1       802.11ax  Disabled                  Enabled (11ax
downgraded)
13       mbssid-9                         default-policy-profile
5       Enabled   2*      802.11ax  Disabled                  Enabled (11ax
downgraded)
14       mbssid-10                        policy-profile              6
  Enabled   2       802.11ax  Disabled                  Enabled (11ax
downgraded)
112      mbssid-b                         default-policy-profile
7       Enabled   0       802.11be  Enabled (256bit cipher)   Disabled (no
MBSSID slot)
17       PSKWLAN                          policy-profile              8
  Enabled   0       N/A       N/A                       N/A
1              open                       default-policy-profile
9       Enabled   3*      802.11be  Enabled (128bit cipher)   Enabled
8        mbssid-4                         default-policy-profile
```

```
10      Enabled   0       N/A       N/A                       N/A
6        mixed-mode                        policy-profile              11
  Enabled   0       N/A       N/A                       N/A
7        mbssid-3                          default-policy-profile
13      Enabled   4*      802.11be  Enabled (256bit cipher)   Enabled
10       mbssid-6                          policy-profile              14
  Enabled   4       802.11be  Enabled (256bit cipher)   Enabled
12       mbssid-8                          default-policy-profile
15      Enabled   4       802.11be  Enabled (256bit cipher)   Enabled
111      mbssid-a                          policy-profile              16
  Enabled   4       802.11be  Enabled (256bit cipher)   Enabled

RLAN-ID  Remote-LAN profile name          Policy profile name
PORT-ID   VAP-ID
-------------------------------------------------------------------------------
16385    rlan_profile                rlan*policy                1        12
```

Note

***** refers to Tx-VAP WLAN of MBSSID set and 0 refers to invalid MBSSID set-id. WLAN is not broadcast in the 6-GHz band. The first six WLANs are degraded to 802.11ax.

Verify WLAN downgrade from 802.11be to 802.11ax mode

This topic describes how to verify WLAN downgrade from 802.11be to 802.11ax mode.

To verify WLAN downgrade from 802.11be to 802.11ax mode, run the **show ap wlan summary** command.

This table describes the reason for downgrading the WLAN.

```
Device# show ap wlan summary

ecf4.0caf.23e1  wpa3owe                           wpa3owe
       Enabled        00:20:52          0            row-AP-11be
           8.31.237.183                              24ghz-EHT
Local            1485             23:31:01          *2/7 (11 dBm)    (1)*
                                 7 days 22 hours 55 minutes 49 seconds
          23 hours 31 minutes 4 seconds                          HE
                     Cipher Not Met
ecf4.0caf.23e5  mbssid-10                         mbssid-10
       Enabled        00:20:52          0            row-AP-11be
           8.31.237.183                              24ghz-EHT
Local            1485             23:31:01          *2/7 (11 dBm)    (1)*
                                 7 days 22 hours 55 minutes 49 seconds
          23 hours 31 minutes 4 seconds                          HE
                     AKM Not Met
ecf4.0caf.23e7  mdns-psk                          PSKWLAN
       Enabled        00:19:56          0            row-AP-11be
           8.31.237.183                              24ghz-EHT
Local            1485             23:31:01          *2/7 (11 dBm)    (1)*
                                 7 days 22 hours 55 minutes 49 seconds
          23 hours 31 minutes 4 seconds                          HE
                     PMF Required
ecf4.0caf.23e9  mbssid-4                          mbssid-4
       Enabled        00:20:52          0            row-AP-11be
           8.31.237.183                              24ghz-EHT
```

```
Local            1485            23:31:01          *2/7 (11 dBm)   (1)*
                                 7 days 22 hours 55 minutes 49 seconds
          23 hours 31 minutes 4 seconds                          HE
                    PMF Required
ecf4.0caf.23ea  mixed-mode-1                        mixed-mode
       Enabled        00:20:52          0             row-AP-11be
           8.31.237.183                                24ghz-EHT
Local            1485            23:31:01          *2/7 (11 dBm)   (1)*
                                 7 days 22 hours 55 minutes 50 seconds
          23 hours 31 minutes 5 seconds                          HE
                    AKM Not Met
ecf4.0caf.23ed  mbssid-6                            mbssid-6
       Enabled        00:20:52          0             row-AP-11be
           8.31.237.183                                24ghz-EHT
Local            1485            23:31:01          *2/7 (11 dBm)   (1)*
                                 7 days 22 hours 55 minutes 50 seconds
          23 hours 31 minutes 5 seconds                          EHT
                    NA
ecf4.0caf.23ee  mbssid-8                            mbssid-8
       Enabled        00:20:52          0             row-AP-11be
           8.31.237.183                                24ghz-EHT
Local            1485            23:31:01          *2/7 (11 dBm)   (1)*
                                 7 days 22 hours 55 minutes 50 seconds
          23 hours 31 minutes 5 seconds                          EHT
                    NA
ecf4.0caf.23ef  wpa3sae                             wpa3sae
       Enabled        00:20:52          0             row-AP-11be
           8.31.237.183                                 5ghz-EHT
Local            1485            23:31:01          *2/8 (16 dBm)   (64)*
                                 7 days 22 hours 55 minutes 50 seconds
          23 hours 31 minutes 5 seconds                          HE
                    AKM Not Met
ecf4.0caf.23ed  mbssid-1                            mbssid-1
       Enabled        00:20:52          0             row-AP-11be
           8.31.237.183                                 5ghz-EHT
Local            1485            23:31:01          *2/8 (16 dBm)   (64)*
                                 7 days 22 hours 55 minutes 50 seconds
          23 hours 31 minutes 5 seconds                          HE
                    Cipher Not Met
ecf4.0caf.23ec  mbssid-5                            mbssid-5
       Enabled        00:20:52          0             row-AP-11be
           8.31.237.183                                 5ghz-EHT
Local            1485            23:31:01          *2/8 (16 dBm)   (64)*
                                 7 days 22 hours 55 minutes 50 seconds
          23 hours 31 minutes 5 seconds                          HE
                    AKM Not Met
ecf4.0caf.23eb  mbssid-9                            mbssid-9
       Enabled        00:20:52          0             row-AP-11be
           8.31.237.183                                 5ghz-EHT
Local            1485            23:31:01          *2/8 (16 dBm)   (64)*
                                 7 days 22 hours 55 minutes 50 seconds
          23 hours 31 minutes 5 seconds                          HE
                    AKM Not Met
ecf4.0caf.23ea  mbssid-10                           mbssid-10
       Enabled        00:20:52          0             row-AP-11be
           8.31.237.183                                 5ghz-EHT
Local            1485            23:31:01          *2/8 (16 dBm)   (64)*
                                 7 days 22 hours 55 minutes 50 seconds
          23 hours 31 minutes 5 seconds                          HE
                    AKM Not Met
ecf4.0caf.23e9  mbssid-b                            mbssid-b
       Enabled        00:20:52          0             row-AP-11be
```

```
          8.31.237.183                                  5ghz-EHT
Local           1485            23:31:01          *2/8 (16 dBm)   (64)*
                                7 days 22 hours 55 minutes 50 seconds
         23 hours 31 minutes 5 seconds                          EHT
                    NA
ecf4.0caf.23e8  mdns-psk                          PSKWLAN
       Enabled        00:19:56          0            row-AP-11be
           8.31.237.183                                 5ghz-EHT
Local           1485            23:31:01          *2/8 (16 dBm)   (64)*
                                7 days 22 hours 55 minutes 50 seconds
         23 hours 31 minutes 5 seconds                          HE
                    PMF Required
ecf4.0caf.23e7       -open                          -open
   Enabled        00:20:53          0            row-AP-11be
       8.31.237.183                               5ghz-EHT              Local
            1485            23:31:02          *2/8 (16 dBm)   (64)*
                           7 days 22 hours 55 minutes 50 seconds
     23 hours 31 minutes 5 seconds                          EHT
                NA
ecf4.0caf.23e6  mbssid-4                          mbssid-4
       Enabled        00:20:53          0            row-AP-11be
           8.31.237.183                                 5ghz-EHT
Local           1485            23:31:02          *2/8 (16 dBm)   (64)*
                                7 days 22 hours 55 minutes 50 seconds
         23 hours 31 minutes 5 seconds                          HE
                    PMF Required
ecf4.0caf.23e5  mixed-mode-1                      mixed-mode
       Enabled        00:20:53          0            row-AP-11be
           8.31.237.183                                 5ghz-EHT
Local           1485            23:31:02          *2/8 (16 dBm)   (64)*
                                7 days 22 hours 55 minutes 50 seconds
         23 hours 31 minutes 5 seconds                          HE
                    AKM Not Met
ecf4.0caf.23e3  mbssid-3                          mbssid-3
       Enabled        00:20:53          0            row-AP-11be
           8.31.237.183                                 5ghz-EHT
Local           1485            23:31:02          *2/8 (16 dBm)   (64)*
                                7 days 22 hours 55 minutes 50 seconds
         23 hours 31 minutes 5 seconds                          EHT
                    NA
ecf4.0caf.23e2  mbssid-6                          mbssid-6
       Enabled        00:20:53          0            row-AP-11be
           8.31.237.183                                 5ghz-EHT
Local           1485            23:31:02          *2/8 (16 dBm)   (64)*
                                7 days 22 hours 55 minutes 50 seconds
         23 hours 31 minutes 5 seconds                          EHT
                    NA
ecf4.0caf.23e7  wpa3sae                           wpa3sae
       Enabled        00:20:53          0            row-AP-11be
           8.31.237.183                                 6ghz-EHT
Local           1485            23:31:02          *2/8 (16 dBm)
(69,65,73,77)*                          7 days 22 hours 55 minutes 50 seconds
                23 hours 31 minutes 5 seconds                           HE
                           AKM Not Met
ecf4.0caf.23e6  wpa3owe                           wpa3owe
       Enabled        00:20:53          0            row-AP-11be
           8.31.237.183                                 6ghz-EHT
Local           1485            23:31:02          *2/8 (16 dBm)
(69,65,73,77)*                          7 days 22 hours 55 minutes 50 seconds
                23 hours 31 minutes 5 seconds                           HE
                           Cipher Not Met
ecf4.0caf.23e5  mbssid-1                          mbssid-1
```

```
       Enabled        00:20:53          0             row-AP-11be
           8.31.237.183                                 6ghz-EHT
Local            1485             23:31:02          *2/8 (16 dBm)
(69,65,73,77)*                          7 days 22 hours 55 minutes 50 seconds
                 23 hours 31 minutes 5 seconds                            HE
                           Cipher Not Met
ecf4.0caf.23e4  mbssid-5                              mbssid-5
       Enabled        00:20:53          0             row-AP-11be
           8.31.237.183                                 6ghz-EHT
Local            1485             23:31:02          *2/8 (16 dBm)
(69,65,73,77)*                          7 days 22 hours 55 minutes 50 seconds
                 23 hours 31 minutes 5 seconds                            HE
                           AKM Not Met
ecf4.0caf.23e3  mbssid-9                              mbssid-9
       Enabled        00:20:53          0             row-AP-11be
           8.31.237.183                                 6ghz-EHT
Local            1485             23:31:02          *2/8 (16 dBm)
(69,65,73,77)*                          7 days 22 hours 55 minutes 50 seconds
                 23 hours 31 minutes 5 seconds                            HE
                           AKM Not Met
ecf4.0caf.23e2  mbssid-10                             mbssid-10
       Enabled        00:20:53          0             row-AP-11be
           8.31.237.183                                 6ghz-EHT
Local            1485             23:31:02          *2/8 (16 dBm)
(69,65,73,77)*                          7 days 22 hours 55 minutes 50 seconds
                 23 hours 31 minutes 5 seconds                            HE
                           AKM Not Met
ecf4.0caf.23ef      -open                               -open
     Enabled        00:20:53          0             row-AP-11be
         8.31.237.183                                6ghz-EHT               Local
            1485             23:31:02          *2/8 (16 dBm)   (69,65,73,77)*
                           7 days 22 hours 55 minutes 50 seconds
     23 hours 31 minutes 5 seconds                             EHT
                NA
ecf4.0caf.23eb  mbssid-3                              mbssid-3
       Enabled        00:20:53          0             row-AP-11be
           8.31.237.183                                 6ghz-EHT
Local            1485             23:31:02          *2/8 (16 dBm)
(69,65,73,77)*                          7 days 22 hours 55 minutes 50 seconds
                 23 hours 31 minutes 5 seconds                            EHT
                           NA
ecf4.0caf.23ea  mbssid-6                              mbssid-6
       Enabled        00:20:53          0             row-AP-11be
           8.31.237.183                                 6ghz-EHT
Local            1485             23:31:02          *2/8 (16 dBm)
(69,65,73,77)*                          7 days 22 hours 55 minutes 50 seconds
                 23 hours 31 minutes 5 seconds                            EHT
                           NA
ecf4.0caf.23e9  mbssid-8                              mbssid-8
       Enabled        00:20:53          0             row-AP-11be
           8.31.237.183                                 6ghz-EHT
Local            1485             23:31:02          *2/8 (16 dBm)
(69,65,73,77)*                          7 days 22 hours 55 minutes 50 seconds
                 23 hours 31 minutes 5 seconds                            EHT
                           NA
ecf4.0caf.23e8  mbssid-a                              mbssid-a
       Enabled        00:20:53          0             row-AP-11be
           8.31.237.183                                 6ghz-EHT
Local            1485             23:31:02          *2/8 (16 dBm)
(69,65,73,77)*                          7 days 22 hours 55 minutes 50 seconds
                 23 hours 31 minutes 5 seconds                            EHT
```

```
                                NA
```

Verify AP WLAN information on 6 GHz band with MBSSID slot

This topic describes how to verify AP WLAN information on 6 GHz band with MBSSID slot.

To verify AP WLAN information on 6 GHz band with MBSSID slot, run the **show ap name***row-AP-11be***wlan dot11 6ghz** command.

```
Device# show ap name row-AP-11be wlan dot11 6ghz

Slot id  : 2
  * Tx-VAP WLAN of MBSSID set
  WLAN ID    BSSID          MLD MAC                MBSSID SET ID
  ------------------------------------------------------------
  4          ecf4.0caf.23e7 Disabled                1*
  5          ecf4.0caf.23e6 Disabled                1
  15         ecf4.0caf.23e5 Disabled                1
  9          ecf4.0caf.23e4 Disabled                1
  13         ecf4.0caf.23e3 Disabled                2*
  14         ecf4.0caf.23e2 Disabled                2
  1          ecf4.0caf.23ef eef4.0caf.23f8          3*
  7          ecf4.0caf.23eb eef4.0caf.23fc          4*
  10         ecf4.0caf.23ea eef4.0caf.23fd          4
  12         ecf4.0caf.23e9 eef4.0caf.23fe          4
  111        ecf4.0caf.23e8 eef4.0caf.23ff          4
```

35 Troubleshooting Lightweight Access Points

Topics:

- Support resources for technical issue resolution

Support resources for technical issue resolution

Lists the available support resources, escalation guidance, and recommendations for resolving technical problems efficiently.

These resources help you resolve problems quickly and provide guidance if you need to escalate the issue.

We offer a variety of SME-authored support documents to help you resolve technical issues without opening a support ticket.

- You can access curated documents written by Cisco SMEs for common technical concerns.
- If a document resolves your issue, no support ticket is required.
- If you need additional help, visit the Cisco Community for solutions provided by other Cisco users. You can find information and advice from Cisco customers who have had similar issues and shared their solutions.
- If your issue persists, open a support ticket at Cisco Support.

> **Note**
>
> When you need to open a support ticket, these documents explain what information you should collect and include in your request. Specify the support document you consulted, and TAC can create an improvement request with the document owner.

Support articles

This section includes documents with use cases that explain why a tunnel break occurs between the Wireless Controller and the APs.

You can apply these documents to different software and hardware listed in the **Components Used** section of each article. However, this does not mean these documents are limited to the items listed in **Components Used**, even if the listed components are for earlier releases.

- Be aware that changes in software or hardware may affect commands to stop working, syntax to change, or GUIs and CLIs to look different between releases.
- The Troubleshoot Access Point Disassociation from Controller explains why a Control and Provisioning of Wireless Access Points (CAPWAP) or Lightweight Access Point Protocol (LWAPP) tunnel may break between Access Points (APs) and the Wireless Controller.

Provide feedback on support documents

You can help improve Cisco support documents by submitting feedback.

Your input helps. A key aspect to improving these support documents is customer feedback. Note that multiple teams maintain and update these documents within Cisco. If you find an issue specific to the document, such as unclear information, confusing sections, or missing information

- provide feedback using the **Feedback** button located in the right panel of the article. The document owner is notified and will either update the article or flag it for removal, and
- include information about the section, area, or issue you had with the document, and suggest what could be improved. Provide detailed information to help us understand your feedback.

Disclaimer and caution

Read this disclaimer and caution about default device configurations so you understand the potential impact of any command.

The information in this document is based on tests performed in a specific lab environment.

- All devices in the lab started with a cleared (default) configuration.
- If you are working on a live network, understand the potential impact of each command before you use it.

Part IV

Access Points Modes

Topics:

- Sniffer Mode
- Monitor Mode
- Management Mode Migration in Cisco Catalyst 916X Access Points
- Configure the controller for FlexConnect
- OEAP Link Test
- Cisco OEAP Split Tunneling
- Access Points Survey Mode
- AP Deployment Mode

36 Sniffer Mode

Topics:

- Sniffer

Sniffer

Introduces sniffer mode concepts for access points, covering XOR radio role sniffer support, prerequisites, restrictions, GUI and CLI configuration, enabling or disabling sniffer functionality, monitoring, XOR radio configuration tasks, and verification with practical examples for effective deployment and troubleshooting.

A sniffer is a network monitoring tool that

- captures and forwards packets on a specified channel to a remote packet analyzer
- allows monitoring and recording of network activity
- detects network problems, and
- receives encapsulated 802.11 traffic at the packet analyzer.

Key characteristics

- Network packet capture: The sniffer captures live packets and forwards them to a packet analyzer for inspection.
- Protocol support: It works with protocols like Airopeek for encapsulation and transfer via specified UDP ports.
- Management integration: Sniffers can be configured through AP modes and require resetting to revert to normal operations.

Recommendations

- Use **Clear** in AP mode to return the AP back to client-serving mode, such as local or FlexConnect depending on the remote site tag configuration.
- Do not use the AP command to change the CAPWAP mode.

XOR radio roles

XOR Radio Role Sniffer support details.

A XOR radio is a configuration that

- allows the XOR radio to function in multiple modes via a single radio interface
- eliminates the need to switch the entire AP into a separate mode, and
- is implemented at the radio level and referred to as "roles."

XOR radio roles facilitate the operation of wireless network radios. This is specifically applicable to models like the Cisco Catalyst 2800, 3800, 4800, and 9100 series AP models. The Sniffer role, supported from the current release onwards, is offered alongside the Client Serving and Monitor roles.

Feature history for sniffer mode
Table 50: Feature history

Release	Feature	Feature information
Cisco IOS XE 17.8.1	XOR Radio Role Sniffer Support on the Access Point	The XOR radio in APs like Cisco 2800, 3800, 4800, and the 9100 series AP models support sniffer role in single radio interface.

Supporting reference information

The radio role is supported in both Local and FlexConnect modes.

Essential hardware and software for sniffer setup

Sniffer prerequisites.

- A dedicated access point: An AP configured as a sniffer cannot simultaneously provide wireless access service on the network. To avoid disrupting coverage, use an access point that is not part of your existing wireless network.
- A remote monitoring device: A computer capable of running the analyzer software.
- Software, supporting files, plug-ins, or adapters: Your analyzer software may require specialized files to function effectively.

Restrictions on sniffer

Sniffer restrictions.

- These are the supported third-party network analyzer software applications:
  - Wildpackets Omnipeek or Airopeek
  - AirMagnet Enterprise Analyzer
  - Wireshark
- The latest version of Wireshark can decode the packets by going to the Analyze mode. Select **decode as**, and switch UDP5555 to decode as PEEKREMOTE.
- You cannot use Sniffer mode when the controller L3 interface is the Wireless Management Interface (WMI).
- When an AP or a radio operates in the sniffer mode, irrespective of its current channel width settings, the AP sniffs or captures only on the primary channel.
- Avoid enabling AP sniffer mode when the controller is connected to Cisco Application Centric Infrastructure (ACI) that uses default endpoint learning. For more information, refer to CSCwa45713.

> **Note**
>
> As both Cisco Catalyst 9166I and 9166D APs have XOR radios, a Board Device File (BDF) has to be loaded to initialize radio 2 for the radios of these APs to work as expected. While the BDF is being loaded and for the file to be loaded correctly, the firmware has to be made non-operational and radios have to be reset. This operation of radio reset due to firmware being non-operational for the purposes of loading the BDFs is deliberate and is an expected behavior. This operation can be observed in both the controller and Cisco Catalyst Center. We recommend that you ignore the core dump that is generated due to this deliberate operation.

Configure an access point as sniffer (GUI)

Configure an access point as sniffer using GUI.

This task guides you through configuring an access point to sniffer mode using the GUI, allowing the access point to capture wireless traffic in a specified location.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. On the **General** tab, update the name of the AP. The AP name can be ASCII characters from 33 to 126, without leading and trailing spaces.
3. Specify the physical location where the AP is present.
4. Choose the **Admin Status** as **Enabled** if the AP is to be in enabled state.
5. Choose the mode for the AP as *Sniffer*.
6. In the **Tags** section, specify the appropriate policy, site, and RF tags that you created on the **Configuration** > **Tags & Profiles** > **Tags** page.

   Note

   If the AP is in sniffer mode, you do not want to assign any tag.

7. Click **Update & Apply to Device**.
8. Choose the mode for the AP as **Clear** to return the AP back to the client-serving mode depending on the remote site tag configuration.

   Note

   Changing the AP mode to Sniffer will set all radios to manual mode. A warning prompts you to revert the radio submode to AUTO if required when changing modes.

The AP is configured in sniffer mode, ready for capturing wireless traffic at the specified location.

Configure an access point as sniffer (CLI)

Configure an access point as sniffer using commands.

Set an AP to sniffer mode so that it can monitor network traffic.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device>enable
   ```

2. Configure the AP to function as a sniffer.

Example

```
Device# ap name access1 mode sniffer
```

Where,

- *ap-name* is the name of the Cisco lightweight access point.
- Use the **no** form of this command to disable the access point as a sniffer.

The AP operates in sniffer mode, capturing and monitoring network traffic.

Enable or disable sniffing on the AP (GUI)

Use the GUI to enable or disable sniffing on the access point.

This task guides you through enabling or disabling sniffing mode on an AP using the GUI.

Before you begin

You must change the AP mode to sniffer mode.

Procedure

---

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. On the **Access Points** page, click the AP name from the 6 GHz, 5 GHz, or 2.4 GHz list.
3. In the **Role Assignment** section, select the **Assignment Method** as *Sniffer*.
4. In the **Sniffer Channel Assignment** section, check the **Sniffer Channel Assignment** checkbox to enable.

   Uncheck the checkbox to disable sniffing on the access point.
5. From the **Sniff Channel** drop-down list, select the channel.

   Note

   By default, the **Sniff Channel** is set to *36* for the **5 GHz** and *1* for the **2.4 GHz**.
6. Enter the IP address into the **Sniffer IP** field.

   To validate the IP address, click **Update & Apply to Device**. If the IP address is valid, the **Sniffer IP Status** displays *Valid*.
7. **Note**

   The section will be enabled for editing only if the **Assignment Method** is set to **Custom**.

   In the **RF Channel Assignment** section, configure these items:

   - From the **RF Channel Width** drop-down list, select the channel width.

- From the **Assignment Method** drop-down list, choose the type of assignment.

> **Note**
>
> If you choose Custom, you must select a channel width and specify an RF channel number to the access point radio. 320 MHz channel width is supported from Cisco IOS XE 17.15.1 onwards.

8. Click **Update & Apply to Device**.

The AP is configured to either operate in sniffing mode or have sniffing mode disabled based on your choice.

Enable or disable sniffing on the AP (CLI)

Enable or Disable Sniffing on the Access Point using commands.

This task enables you to manage the sniffing feature on an AP using CLI commands, specifically to enable or disable it as necessary.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

2. Enable sniffing on the AP.

   Example

   ```
   Device# <userinput>ap name access1 sniff dot11b 1 9.9.48.5</userinput>
   ```

   - *channel* is the valid channel to be sniffed. For 802.11a, the range is 36 to 165. For 802.11b, the range is 1 to 14. For dot11 6 GHz, the range is between 1 and 233.
   - *server-ip* is the IP address of the machine running network monitoring software.

3. Disable sniffing on the AP.

   Example

   ```
   Device#<userinput>ap name access1 no sniff dot116ghz</userinput>
   ```

The sniffing feature is enabled or disabled on the AP based on the commands executed. Ensure that you verify the current status of the configuration.

Configure XOR radio role sniffer support on the access point (CLI)

Configure XOR radio role sniffer support on access points using commands.

Enable the XOR radio on a AP to operate as a sniffer by manually configuring its role and settings through CLI.

Procedure

1. Enable privileged EXEC mode. Enter your password, if prompted.

   Example

   ```
   Device> enable
   ```

2. Shut down the XOR radio.

   Example

   ```
   Device# ap name AP687D.B45C.189C dot11 dual-band shutdown
   ```

3. Convert the XOR radio role to manual.

   Example

   ```
   Device# ap name ap-name dot11 dual-band role manual client-serving
   ```

4. Configure XOR radio to manually operate in a specific band.

   Example

   ```
   Device# ap name  AP687D.B45C.189C dot11 dual-band band 5ghz
   ```

5. Enable XOR radio role Sniffer support on AP from the controller.

   Example

   ```
   Device# ap name AP687D.B45C.189C dot11 dual-band radio role manual sniffer
   channel 100 ip 9.4.197.85
   ```

   Where,

   - *ap-name* is the name of the Cisco lightweight access point.
   - *channel-number* is the channel number.

6. Activate the XOR radio.

   Example

   ```
   Device# ap name AP687D.B45C.189C no dot11 dual-band shutdown
   ```

7. Return to privileged EXEC mode.

Example

```
Device# end
```

Note

When configuring the radio to work as a Sniffer in the 5 GHz band, make sure to change the band of the radio manually.

XOR radio on the AP is configured to operate as a sniffer, allowing you to monitor and analyze wireless traffic on a specified channel.

Verify sniffer configurations

Verify Sniffer configurations.

Use these commands to verify sniffer configurations on AP and gather specifics regarding the sniffing setup in multiple bands and slots.

Table 51: Commands for verifying sniffer configurations

Commands	Description
show ap name *ap-name* **config dot11 {24ghz | 5ghz | 6ghz | dual-band}**	Displays the sniffing details.
show ap name *ap-name* **config slot** *slot-ID*	Displays the sniffing configuration details. *slot-ID* ranges from 0 to 3. All access points have slot 0 and 1.

Verify XOR radio role sniffer configuration

Verify XOR radio role sniffer configuration.

To verify the XOR radio role sniffer configuration for a given AP, use this command:

```
Device# show ap name AP687D.B45C.189C config slot 0

Sniffing                                        : Enabled
Sniff Channel                                   : 6
Sniffer IP                                      : 198.51.100.10
Sniffer IP Status                               : Valid
ATF Mode                                        : Disable
ATE Optimization                                : N/A
AP Submode                                      : Not Configured
Remote AP Debug                                 : Disabled
Logging Trap Severity Level                     : information
Software Version                                : 17.9.0.18
Boot Version                                    : 1.1.2.4
Mini IOS Version                                : 0.0.0.0
Stats Reporting Period                          : 60
```

```
primary_discovery_timer                          : 120
LED State                                        : Enabled
LED Flash State                                  : Enabled
LED Flash Timer                                  : 0
PoE Pre-Standard Switch                          : Disabled
PoE Power Injector MAC Address                   : Disabled
Power Type/Mode                                  : PoE/Full Power
Number of Slots                                  : 4
AP Model                                         : C9136I-B
IOS Version                                      : 17.9.0.18
Reset Button                                     : Disabled
AP Serial Number                                 : FOC25322JJZ
AP Certificate Type                              : Manufacturer Installed
Certificate
AP Certificate Expiry-time                       : 08/09/2099 20:58:26
AP Certificate issuer common-name                : High Assurance SUDI CA
AP Certificate Policy                            : Default
AP CAPWAP-DTLS LSC Status
    Certificate status        : Not Available
AP 802.1x LSC Status
    Certificate status        : Not Available
AP User Name                                     : admin
AP 802.1X User Mode                              : Global
AP 802.1X User Name                              : Not Configured
Cisco AP System Logging Host                     : 203.0.113.10
AP Up Time                                      : 4 hours 20 minutes 55 seconds
AP CAPWAP Up Time                               : 4 hours 16 minutes 17 seconds
Join Date and Time                               : 01/19/2022 03:06:12

Attributes for Slot 0
  Radio Type                                     : 802.11ax - 2.4 GHz
  Radio Mode                                     : Sniffer
  Radio Role                                     : Sniffer
  Maximum client allowed                         : 400
  Radio Role Op                                  : Manual
  Radio SubType                                  : Main
  Administrative State                           : Enabled
  Operation State                                : Up
```

Examples for sniffer configurations and monitoring

Sniffer configurations and monitoring examples.

This example shows how to configure an AP as sniffer:

```
Device# ap name access1 mode sniffer
```

This example shows how to enable sniffing on the AP:

```
Device# ap name sniffer dot11 5ghz sniff 44 1.1.1.1
```

This example shows how to disable sniffing on the AP:

```
Device# ap name access1 no sniff dot11b
```

This example shows how to display the sniffing configuration details:

```
Device# show ap name access1 config dot11 24ghz
Device# show ap name access1 config slot 0
```

37 Monitor Mode

Topics:

- Monitor mode

Monitor mode

Introduces the fundamental concepts of Monitor Mode in wireless deployments and provides guidance for enabling Monitor Mode using both GUI and CLI interfaces.

A monitor mode is a wireless operational mode that

- optimizes the monitoring of 802.11b/g/x network channels
- enhances location calculation for Radio Frequency Identification (RFID) tags, and
- allows limited channel scanning.

Key features of monitor mode

- Channel optimization: Optimize the monitoring by limiting the scanning to 2.4-GHz channels, such as 1, 6, and 11.
- RFID tag tracking: Enable precise tag-tracking by using specific operational frequencies.

AP mode transition

You can move an AP to a particular mode (sensor mode to local mode or FlexConnect mode) using the site tag with the corresponding mode. If the AP is not tagged to any mode, it uses the default site tag mode.

> **Tip**
>
> To optimize operational efficiency, ensure that the AP is tagged correctly.

returning AP to client-serving mode

You must use clear in AP mode to return the AP back to client-serving mode, for example the local mode or FlexConnect mode depending on the remote site tag configuration.

Enable monitor mode (GUI)

Enable monitor mode using GUI.

Switch the AP to monitor mode using the GUI.

Use these steps to enable monitor mode for the AP:

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. In the **Access Points** page, expand the **All Access Points** section and click the name of the AP to edit.
3. In the **Edit AP** page, click the **General** tab and from the **AP Mode** drop-down list, choose **Monitor**.
4. Click **Update & Apply to Device**.
5. Choose the mode for the AP as **clear** to return the AP back to the client-serving mode depending on the remote site tag configuration.

The AP is now in monitor mode and can observe wireless traffic without serving clients.

Enable monitor mode (CLI)

Enable monitor mode using commands.

Enable and configure monitor mode on APs, ensuring they scan specific channels for network monitoring.

Follow these steps to enable monitor mode:

Procedure

1. Enable monitor mode for the AP.

   Example

   ```
   Device# ap name 3602a mode monitor
   ```

2. Configure the AP to scan only the Dynamic Channel Assignment (DCA) channels supported by its country of operation.

   Example

   ```
   Device# ap name 3602a monitor tracking-opt
   ```

3. Choose up to four specific 802.11b channels to be scanned by the AP.

   Example

   ```
   Device# ap name 3602a monitor dot11b 1 2 3 4
   ```

   In the United States, you can assign any value from 1 to 11 (inclusive) to the channel variable. Other countries support additional channels. You must assign at least one channel.

   Note

4. Configure the 802.11 6-Ghz radio role manual monitor.

   Example

   ```
   Device# ap name cisco-ap dot11 6ghz slot 3 radio role manual monitor
   ```

   slot 3 radio role manual monitor

5. View configuration and statistics of 802.11a or 802.11b or 6-GHz channel assignment.

   Example

   ```
   Device# show ap dot11 5ghz channel
   ```

6. View configuration and statistics summary of 6 GHz band APs.

Example

```
Device# show ap dot11 6ghz summary
```

The APs are set to monitor mode, scanning the specified channels and enabling effective network monitoring and channel assessment.

38 Management Mode Migration in Cisco Catalyst 916X Access Points

Topics:

- Management mode migration for Cisco Catalyst Wireless 916X Series AP

Management mode migration for Cisco Catalyst Wireless 916X Series AP

Outlines management modes for Cisco Catalyst Wireless 916X APs, describes dual-band radio support, and details procedures and considerations for migrating access points from DNA Management mode to Meraki Management mode using CLI or controller GUI.

A management mode is a configuration setting in networking devices that:

- determines how a device connects to the network
- controls the operating mode of access points, and
- allows flexibility through configurable options such as cloud-based or on-premises management.

Cisco Catalyst Wireless 916X APs support both 6 GHz and 5 GHz bands through dual-band slot 3 radios.

Feature History for management mode migration in Cisco catalyst wireless 916X access points
Table 52: Feature history

Release	Feature	Feature information
Cisco IOS XE Cupertino 17.9.1	Management Mode Migration in Cisco Catalyst Wireless 916X Series Access Points	This feature allows you to convert the AP mode between DNA Management mode and Meraki Management mode, depending on your requirements. **Note** The document explains the conversion from DNA Management mode to Meraki Management mode and not vice versa.

Management modes

- DNA Management mode: Allows the access point to utilize Cisco's Digital Network Architecture for advanced network capabilities and management.
- Meraki Management mode: Enables integration and management through Cisco's Meraki Cloud.

> **Note**
>
> The management mode migration configuration is specifically oriented for transitioning from DNA Management mode to Meraki Management mode and not the reverse. Migration can be configured through CLI in privileged EXEC mode at the AP level and from the controller GUI.

Regulatory domain

Regulatory domain details.

For regulatory domain support, Cisco Catalyst 916x Series APs (CW916x) support Rest of the World (RoW) and various fixed domains.

The Cisco Catalyst 916x Series APs support these domains:

- -B
- -E
- -A
- -Z
- -Q
- -I
- -R

These domains define the specific regions or countries where the Cisco Catalyst 916x can operate in compliance with local regulations.

AP join flow functionality

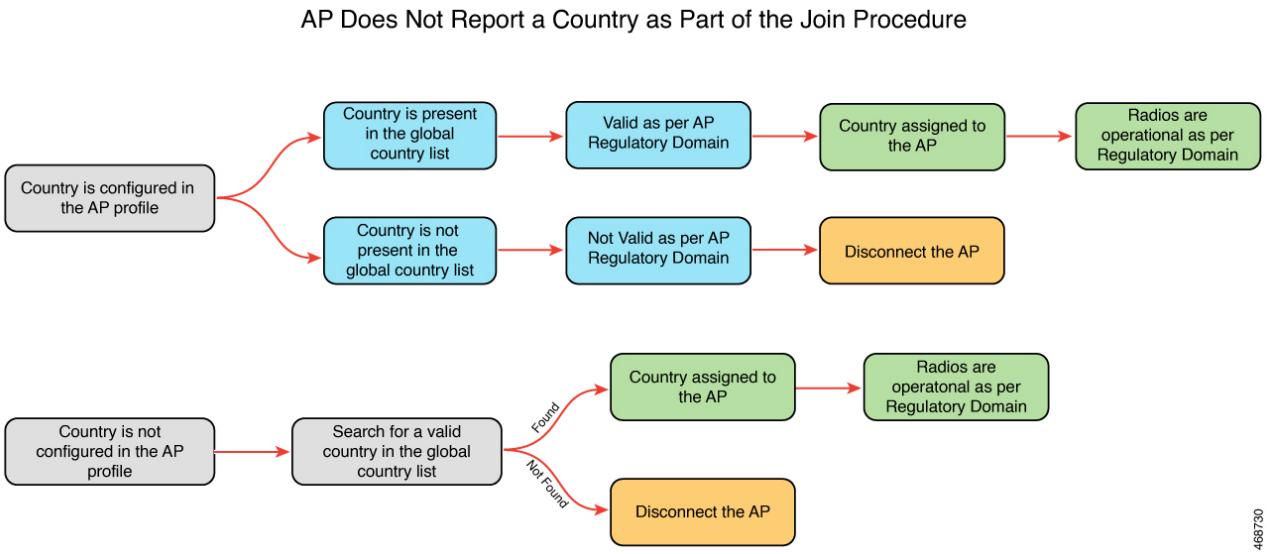
During the AP join flow, the AP passes the regulatory domain details and configured country to the controller. The controller assigns or validates the right country of operation. After validation based on the decision tree, the controller informs the AP of the country with which it should be configured.

Recommendation to configure AP regulatory domain

Use cases for regulatory domain.

AP configured with non-RoW regulatory domain

Case 1: AP does not report a country as part of the join procedure.

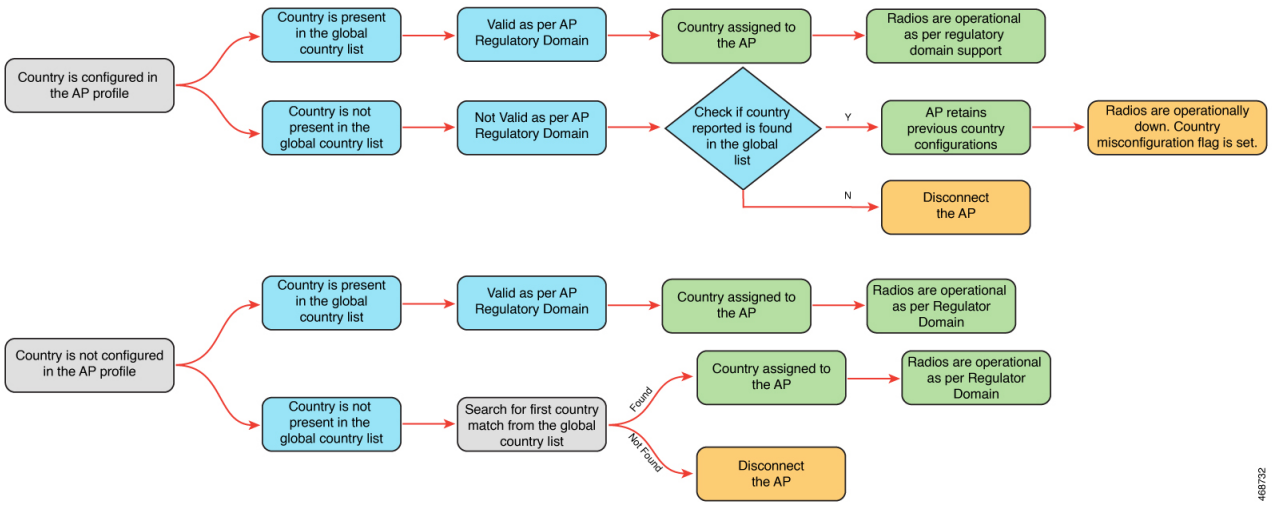


In the non-RoW regulatory domain, when an AP does not report a country as part of the join procedure, the process occurs:

- AP profile has a country configured.
  - If the country configured in the AP profile is present in the global country list, and is valid as per the AP regulatory domain, the country that is configured in the AP profile is assigned to the AP. Radios become operational as per the country or regulatory domain support.
  - If the country configured in the AP profile is not present in the global country list, and is not valid as per the AP regulatory domain, the AP is disconnected.
- AP profile does not have a country configured. Find a valid country from the global country list (the first match), as per the AP regulatory domain.
  - If the country is found, the country is assigned to the AP and the radios become operational as per the country or regulatory domain support.
  - If the country is not found, the AP is disconnected.

Case 2: AP reports a country as part of the join procedure.

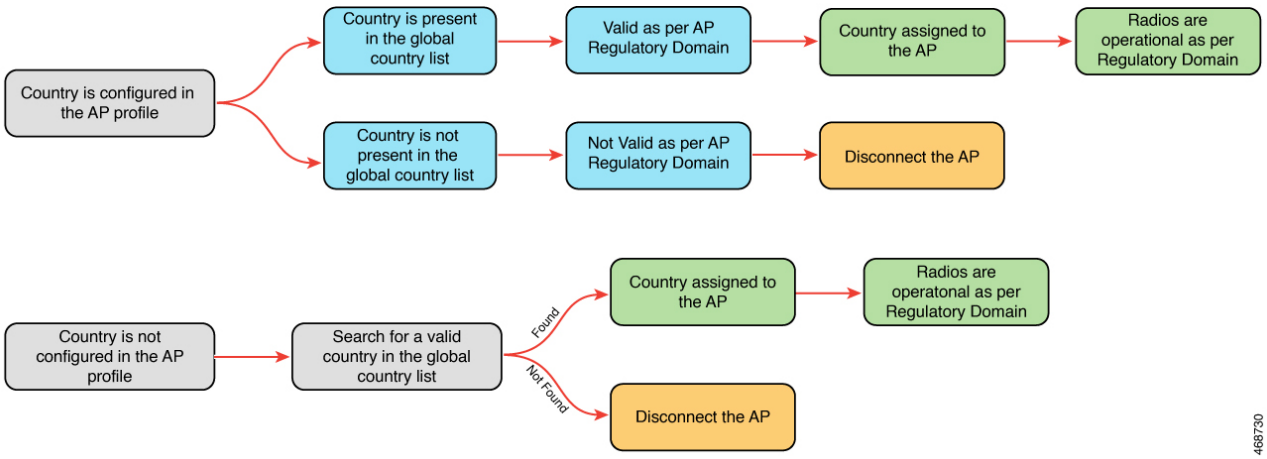
AP Reports Country as Part of the Join Procedure



In the non-RoW regulatory domain, when an AP reports a country as part of the join procedure, the following takes place:

- The AP profile has a country configured.
  - If the country configured in the AP profile is present in the global country list, and it is valid as per the AP regulatory domain, the country that is configured in the AP profile is assigned to the AP. Radios become operational as per the country or regulatory domain support.
  - If the country configured in the AP profile is not present in the global country list, and is not valid as per the AP regulatory domain, check the global country list to confirm the country's presence. If it is present in the global list, the AP retains the previous country configuration and the radios are not operational with the country misconfiguration flag set. If the country is not located in the global list, the AP is disconnected.
- The AP profile does not have a country configured.
  - If the country reported by the AP is found in the global country list, and is valid as per the AP regulatory domain, the country is assigned to the AP and the radios become operational as per the country or regulatory domain support.
  - If the country is not present in the list, search for the first country match from the global list. If the country is found, the country is assigned to the AP and the radios become operational. If the country is not found, the AP is disconnected.

AP Does Not Report a Country as Part of the Join Procedure



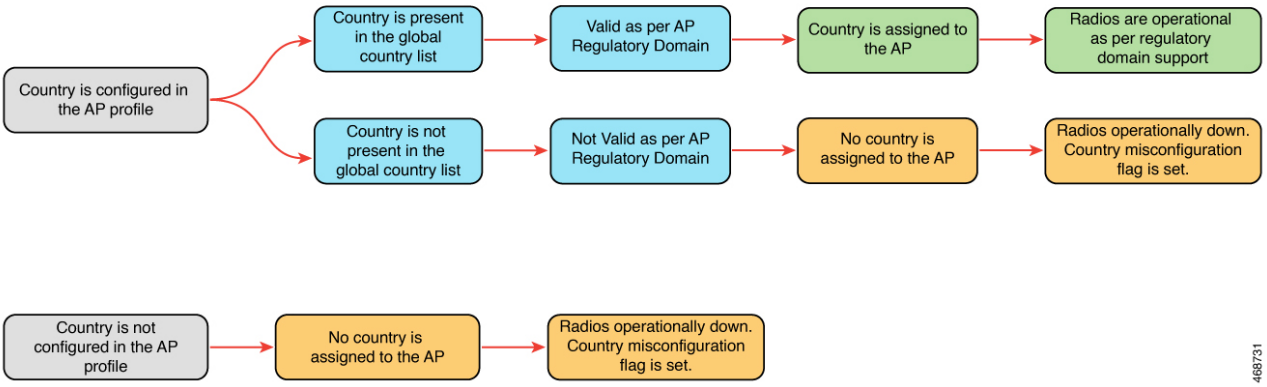
In the non-RoW regulatory domain, when an AP does not report a country as part of the join procedure, the following takes place:

- AP profile has a country configured.
  - If the country configured in the AP profile is present in the global country list, and is valid as per the AP regulatory domain, the country that is configured in the AP profile is assigned to the AP. Radios become operational as per the country or regulatory domain support.
  - If the country configured in the AP profile is not present in the global country list, and is not valid as per the AP regulatory domain, the AP is disconnected.
- AP profile does not have a country configured. Find a valid country from the global country list (the first match), as per the AP regulatory domain.
  - If the country is found, the country is assigned to the AP and the radios become operational as per the country or regulatory domain support.
  - If the country is not found, the AP is disconnected.

AP configured with RoW regulatory domain

Case 1: The AP does not report a country as part of the join procedure.

AP Does Not Report a Country as Part of the Join Procedure

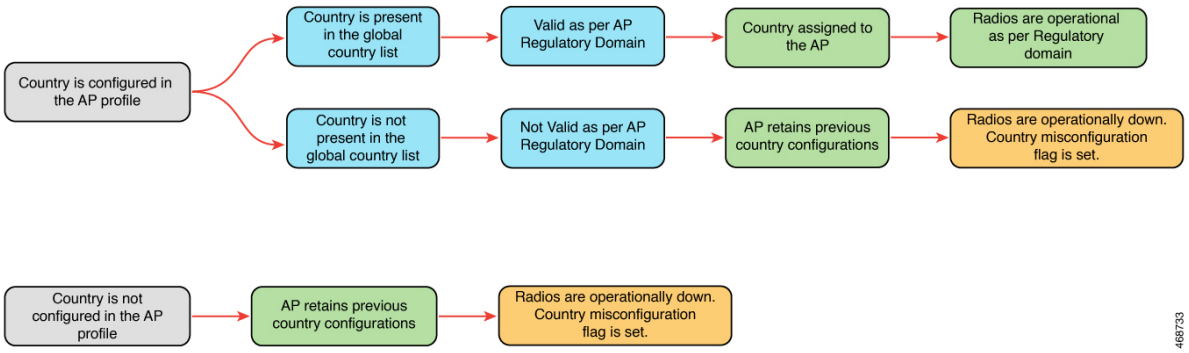


In the RoW regulatory domain, when an AP does not report a country as part of the join procedure, this process occurs:

- The AP profile has a country configured.
  - If the country configured in the AP profile is present in the global country list, and is valid as per the AP regulatory domain, the country configured in the AP profile is assigned to the AP. Radios become operational as per the country or regulatory domain support.
  - If the country configured in the AP profile is not present in the global country list, and is not valid as per the AP regulatory domain, country is not assigned to the AP and radios are not operational, and the country misconfiguration flag is set.
- If the AP profile does not have a country configured, the country is not assigned to the AP and radios are not operational, and the country misconfiguration flag is set.

Case 2: The AP reports a country as part of the join procedure.

AP Reports a Country as Part of the Join Procedure



In the RoW regulatory domain, when an AP reports a country as part of the join procedure, this happens:

- The AP profile has a country configured.
  - If the country configured in the AP profile is present in the global country list, and it is valid as per the AP regulatory domain, the country that is configured in the AP profile is assigned to the AP. Radios become operational as per the country or regulatory domain support.
  - If the country configured in the AP profile is not present in the global country list, and is not valid as per the AP regulatory domain, the AP retains the previous country configuration and the radios are not operational with the country misconfiguration flag set.
- The AP retains the previous country configuration and the radios are not operational with the country misconfiguration flag set.

Configure management mode migration (GUI)

Configure Management Mode Migration using GUI.

Use the GUI to transition APs from DNA management mode to Meraki management mode. This improves network management efficiency.

Before you begin

You must configure the country code on the AP profile. To configure the country code, navigate to **Configuration** > **Tags & Profiles** > **AP Join** page. Click an AP profile to edit. In the **General** tab, select the country code from the drop-down list.

Use these steps to migrate the management mode using the GUI:

Procedure

1. Choose **Configuration** > **Wireless** > **Migrate to Meraki Management Mode**.
2. Select the APs you need by checking the boxes next to them from the displayed list.
   The **Migrate to Meraki Management Mode** button is enabled.
3. Click **Migrate to Meraki Management Mode** button to perform a validation check on the selected APs. If the validation check is successful, the **Next** button is enabled.
4. Click **Next** to start the process.
5. On the **Confirm Management Mode Migration** window, perform these actions:
   a. Select the **Agree and continue** check box.
   b. Click **Yes** to confirm.

   The **Management Mode Migration Successful** section displays the APs that were migrated to the Meraki management mode. The **Management Mode Migration Failed** section displays the APs that were retained in DNA management mode.
6. Click **Restart Workflow** to restart the workflow for APs that did not migrate from DNA management mode to Meraki management mode.

APs migrate to Meraki management mode. APs that fail to migrate remain in DNA management mode, allowing for further troubleshooting or additional attempts.

Export APs migrated to Meraki management mode (GUI)

Export the list of Meraki management mode-migrated APs to ensure information is available for further use, or integration into other tools.

You can export the details about the Meraki management mode-migrated APs either from the **Change to Meraki Persona** tab after the workflow is completed or from the **Previously changed APs** tab.

Use these steps to export Meraki management mode-migrated APs:

Procedure

1. Choose **Configuration** > **Wireless** > **Migrate to Meraki Management Mode**.
2. Click **Export** to export the list of APs.
3. Select whether you want to export only the current page or all pages. Click **Yes** to continue.
4. On the **Export** window, select the export method. The available options are:
   - **Serial Number**
   - **JSON**
   - **Export to Meraki Dashboard**

> **Note**
>
> We recommend the Export to Meraki Dashboard option as you can directly export the migrated APs information into the Meraki Dashboard.

5. Click **Copy** to copy the migrated APs. Click **Download** and save the file location.

The list of migrated APs was successfuly exported, ensuring that the data is readily available for review, storage, or integration.

Configure the access point management mode (CLI)

Configure the AP Management Mode using commands.

Change the management mode of an AP to Meraki using CLI.

Before you begin

Ensure that the AP is compatible with Meraki to run any of the EXEC commands. To verify, use the **show ap management-mode meraki capability summary** command.

> **Note**
>
> If the country code is misconfigured, the change of management mode will not be allowed for any of the EXEC commands, except the **force** command.
>
> If the regulatory domain is misconfigured for any slot, the change of management mode is not allowed for any of the EXEC commands, except the **force** command.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter the password, if prompted.

2. Change the AP management mode to Meraki.

   Example

   ```
   Device# ap name <i>Cisco-AP-name</i> management-mode meraki
   Device# ap name Cisco-AP-name management-mode meraki force
   Device# ap name Cisco-AP-name management-mode meraki noprompt
   Device# ap name Cisco-AP-name management-mode meraki force noprompt
   ```

   Here, **force** skips the validations at the controller and attempts Meraki management mode change at the AP.

noprompt skips the user prompt for attempting AP management mode change.

3. Optional: Clear the Meraki AP-related data.

Example

```
Device# clear ap meraki stats
```

The AP is configured to the Meraki management mode.

Verify the management mode migration details

Verify the management mode migration details

To view the summary of the Meraki-capable AP information, run this command:

```
Device# show ap management-mode meraki capability summary
AP Name                    AP Model             Radio MAC        MAC Address
     AP Serial Number       Meraki Serial Number
-------------------------------------------------------------------------------
APXXXD.BXXX.1XXX           CW9162I             6XXd.bXXe.eXX0   6XXd.bXXe.eXX0
    FOCXXXXXB90             FOCXXXXXB90
```

To view the failure summary of the AP along with the migration attempt timestamp, run this command:

```
Device# show ap management-mode meraki failure summary
AP Name               AP Model          Radio MAC        MAC Address
Conversion Attempt        AP Serial Number       Meraki Serial Number   Reason
 Code
-------------------------------------------------------------------------------
APXXXD.BXXC.1         CW9162I           6XXd.bXXe.eXX0   6XXd.bXXe.eXX0
03/03/2022 17:17:42 IST   FOCXXXXXB90            FOCXXXXXB90
Regulatory domain not set
```

To view the successful Meraki management mode migration attempts of all the APs, run this command:

```
Device# show ap management-mode meraki change summary
AP Name                   AP Model          Radio MAC        MAC Address
Conversion Timestamp       AP Serial Number        Meraki Serial Number
-------------------------------------------------------------------------------
APXXXX.3XXX.EXXX          CW9166I-B         1XXX.2XXX.1100   ccXX.3XXX.eXX0
05/02/2022 07:48:56 CST    KWC2XXXXX5G             Q5XX-4XXX-K7XX
```

39 Configure the controller for FlexConnect

Topics:

- FlexConnect Authentication
- Configure FlexConnect
- NAT-PAT functionalities in FlexConnect
- Split tunneling for FlexConnect
- VLAN-based central switching for FlexConnect
- OfficeExtend AP for FlexConnect
- Proxy address resolution protocol
- Overlapping client IP address in flex deployment
- FlexConnect high scale mode
- Flex resilient with Flex and bridge modes
- SuiteB ciphers in FlexConnect mode

Configure the controller for FlexConnect.

You can configure the controller for FlexConnect in either centrally switched WLAN or locally switched WLAN environments.

Table 53: WLAN scenarios

The controller configuration for FlexConnect consists of creating centrally switched and locally switched WLANs. This table shows three WLAN scenarios.

WLAN	Security	Authentication	Switching	Interface mapping (GUEST VLAN)
Employee	WPA1+WPA2	Central	Central	Management (centrally switched GUEST VLAN)
Employee-local	WPA1+WPA2 (PSK)	Local	Local	101 (locally switched GUEST VLAN)
Guest-central	Web authentication	Central	Central	Management (centrally switched GUEST VLAN)
Employee-local-auth	WPA1+WPA2	Local	Local	101 (locally switched VLAN)

FlexConnect Authentication

Introduces FlexConnect architecture, authentication workflows, controller discovery, switching modes, local authentication, RADIUS server configuration, operational restrictions, and provides procedures for configuring policy tags, site tags, ACLs, and applying access policies through both GUI and CLI.

A FlexConnect authentication mode is a WLAN operating state that

- defines how a FlexConnect AP handles client authentication and data switching
- changes its behavior based on connection status to the controller, and
- enables resilient client connectivity during both connected and standalone operation.

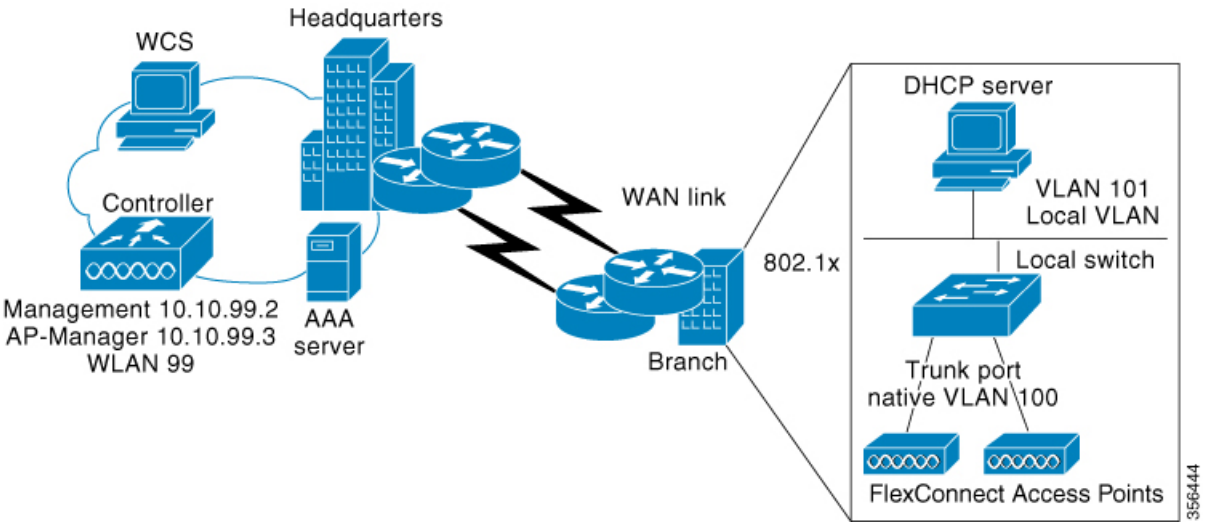
Locally switched: The AP forwards client data directly to the local LAN or VLAN at the site instead of tunneling it through the controller.

Centrally switched: The AP forwards the client's data traffic to the controller, depending on the WLAN configuration.

FlexConnect is a wireless solution for branch office and remote office deployments. It enables customers to configure and control access points (AP) in a branch or remote office from the corporate office through a wide area network (WAN) link without deploying a controller in each office. The FlexConnect access points can also switch client data traffic locally and perform client authentication locally when their connection to the controller is lost. When they are connected to the controller, they can also send traffic back to the controller. FlexConnect access points support multiple SSIDs. In the connected mode, the FlexConnect access point can also perform local authentication.

An access point does not have to reboot when moving from local mode to FlexConnect mode and vice-versa.

FlexConnect Deployment



The embedded wireless controller software uses a more robust fault-tolerance methodology with FlexConnect APs. Whenever a FlexConnect AP disassociates from controller, it moves to the standalone mode. Centrally switched clients are disassociated. However, the FlexConnect access point continues to serve locally switched clients. When a FlexConnect AP loses and rejoins its primary or identically configured secondary controller, locally switched client sessions are maintained, providing seamless connectivity.

After the client connection is established, the controller does not restore the original attributes of the client. The client username, current rate and supported rates, and listen interval values are reset to the default or new configured values only after the session timer expires.

The controller can send multicast packets in the form of unicast or multicast packets to an access point. In FlexConnect mode, an access point can receive only multicast packets.

In Cisco Catalyst 9800 Series Wireless Controller, you can define a flex connect site. A FlexConnect site can have a flex connect profile associate with it. You can have a maximum of 100 access points for each flex connect site.

FlexConnect access points support a one-to-one network address translation (NAT) configuration. They also support PAT for all features except true multicast. Multicast is supported across NAT boundaries when configured using the Unicast option. FlexConnect access points also support a many-to-one NAT or PAT boundary, except when you want true multicast to operate for all centrally switched WLANs.

Workgroup Bridge and Universal Workgroup Bridge modes are supported on FlexConnect APs for locally switched clients.

FlexConnect supports IPv6 clients by bridging traffic to the local VLAN, similar to IPv4 operation.

- An office with intermittent WAN connectivity keeps wireless clients connected using local authentication, with data switched locally until the central controller is available again.
- A FlexConnect AP at a remote branch uses backup RADIUS for 802.1X authentication during WAN outages.

Analogy: retail chain

Imagine a retail chain with a central headquarters (controller) and a branch store (the FlexConnect AP).

Normal day: Every time a customer wants to make a purchase, the cashier phones headquarters for approval and processing. This is like **central authentication and switching**.

Local authentication (policy choice): Even on a normal day, the branch can be configured to keep a small credit-card terminal in the store. If management decides to use it, the cashier can approve transactions locally without phoning headquarters. The headquarters link is still up, but the store chooses to handle the verification itself. That terminal is **“local authentication"** .

Stand-alone mode (connectivity condition): One day the phone lines to headquarters go down. The branch is forced to rely on its credit-card terminal, whether it originally planned to or not, if it wants to keep making sales. The store switches on its emergency lights, keeps serving customers, and records the day’s sales to upload later. That forced independence is **“standalone mode”**.

Key Takeaway

Standalone mode is the situation (phone lines down).

Local authentication is the tool (the in-store terminal) that lets the store keep serving customers—even when the phone lines are fine and especially when they are not.

How FlexConnect authentication works

Summary

FlexConnect authentication enables wireless APs to maintain client connectivity and authentication in various network scenarios, either by connecting to a central controller or operating autonomously. This process is essential for branch offices or remote sites with unreliable WAN links.

The key components involved in the process are:

- FlexConnect AP: Discovers controllers, downloads configuration, and performs client authentication and data switching either locally or through the controller.
- The controller: Centralizes configuration management and client authentication when available.
- Client devices: Attempt to authenticate and connect to the network through FlexConnect APs.
- RADIUS server: Provides authentication services, either centrally through the controller or locally through a backup server in standalone modes.

Workflow

The process involves the following stages:

1. **Controller Discovery and Join**
   - When a FlexConnect AP boots up, it searches for and joins a wireless LAN controller, downloading the latest configuration and software image.

When...	Then...	And...
The controller is reachable, the AP enters connected mode.	The controller performs central authentication.	Based on WLAN configuration • client data is switched either through the controller (central switching), or • locally at the AP (local switching).
The controller is not reachable, the AP enters standalone mode.	The AP performs local authentication using stored configuration and, if needed, a backup RADIUS server.	• Client data is switched locally. • Special states (such as “authentication down, local switching”) manage client behavior when authentication cannot occur. When operating locally, guest authentication and local RADIUS on the controller are not supported.

2. **Failover and Recovery**
   - If controller connectivity is lost, the AP attempts to reach the gateway via ARP and retries controller discovery.
   - If discovery fails, it attempts DHCP renewal and, if still unsuccessful after multiple attempts, falls back to static IP and reboots to recover.
3. **Return to Connected Mode**

   Upon reconnecting to the controller, the AP disassociates clients, applies new configuration, and resumes normal connectivity, with central authentication and state management.

Result

FlexConnect authentication ensures clients can maintain connectivity and authentication even during network disruptions, supports flexible deployment models, and reduces branch office WAN dependency.

Controller discovery methods

When a FlexConnect AP boots up, it searches for a controller. If it finds one, it joins the controller, downloads the latest software and configuration, and initializes the radio. The configuration is saved in nonvolatile memory to support standalone mode if the controller becomes unreachable.

A FlexConnect AP can discover the controller’s IP address through multiple methods:

- **DHCP-based discovery:** If the access point gets its IP from a DHCP server, it uses CAPWAP or LWAPP discovery. OTAP is not supported.
- **Static IP discovery**: If configured with a static IP, the access point can use all discovery methods except DHCP option 43. DNS resolution is recommended if Layer 3 broadcast fails. With DNS, any AP with a static IP address that knows of a DNS server can find at least one controller.
- **Priming**: For remote networks without CAPWAP or LWAPP, priming allows manual configuration through the CLI to specify the controller.

FlexConnect authentication and switching modes

> **Note**
>
> The LEDs on the AP change as the device enters different FlexConnect modes. See the hardware installation guide for your AP for information on LED patterns.

When a client associates with a FlexConnect AP, the AP sends all authentication messages to the controller and, based on the WLAN configuration, either switches the client's data packets locally (locally switched) or sends them to the controller (centrally switched).

For client authentication (open, shared, EAP, web authentication, and NAC) and data packets, the WLAN operates in one of the following states, determined by its configuration and the controller connectivity status:

- Central authentication, central switching: The controller handles both client authentication and data switching. All client data is tunneled to the controller. This state is valid only in the connected mode.
- Central authentication, local switching–In this state, the controller handles client authentication, and the FlexConnect AP switches data packets locally. After the client authenticates successfully, the controller sends a configuration command with a new payload to instruct the FlexConnect AP to start switching data packets locally. This message is sent per client. This state is applicable only in connected mode.
- Local authentication, local switching: The AP both authenticates clients and switches data locally. This state works in both connected and standalone mode.
- Authentication down, switch down: The WLAN disassociates clients and stops sending beacons and probes. Valid for both connected and standalone modes.
- Authentication down, local switching: New client authentication is rejected, but existing client sessions are kept alive. Valid in standalone mode.

In the connected mode, the controller receives only minimal information about locally authenticated clients. Some information not available to the controller are:

- Policy type
- Access VLAN
- VLAN name
- Supported rates
- Encryption cipher

When a FlexConnect AP is unable to reach the controller, WLANs configured with open, shared, WPA-PSK, or WPA2-PSK authentication continue to authenticate clients locally if an external RADIUS server is available. If the controller becomes reachable again, all clients are disassociated and a new configuration is applied before connectivity resumes.

In web-authentication mode, client DNS replies must pass through the controller during authentication. After successful web authentication, all traffic switches locally.

Standalone Mode

When a FlexConnect AP cannot reach the controller, it automatically enters standalone mode and begins authenticating clients on its own

Behavior in standalone mode

- WLANs configured for open, shared, WPA-PSK, or WPA2-PSK authentication enter the **local authentication, local switching** state and continue new client authentications.

- WLANs configured for 802.1X, WPA-802.1X, WPA2-802.1X, or Cisco Centralized Key Management require an external or local RADIUS server to remain operational.
- WLANs configured for central switching move to **authentication down, switching down**; WLANs configured for local switching move to **authentication down, local switching**
- The AP forwards data frames locally while it authenticates clients.

When a FlexConnect AP enters a standalone mode, the AP checks whether it is able to reach the default gateway through ARP. If so, it will continue to try and reach the controller .

If the AP fails to establish the ARP:

- The AP attempts to discover for five times and if it still cannot find the controller , it tries to renew the DHCP on the ethernet interface to get a new DHCP IP.
- The AP will retry for five times, and if that fails, the AP will renew the IP address of the interface again, this will happen for three attempts.
- If the three attempts fail, the AP will fall back to the static IP and will reboot (only if the AP is configured with a static IP).
- Reboot is done to remove the possibility of any unknown error the AP configuration.

Local authentication in a FlexConnect AP

Local authentication in a FlexConnect AP is an authentication method in which

- the FlexConnect AP independently authenticates clients without forwarding authentication requests to a central controller
- client data packets are switched locally by the AP, reducing round-trip latency and dependence on WAN bandwidth
- authentication capabilities are built into the AP for handling protocols like 802.1X, WPA-PSK, WPA2-PSK, and others reduces the latency requirements of the branch office, and
- is useful where you cannot maintain a remote office setup of a minimum bandwidth of 128 kbps with the round-trip latency no greater than 100 ms and the maximum transmission unit (MTU) no smaller than 576 bytes.

Additional information

- Do not enable guest authentication on WLANs with FlexConnect local authentication; guest authentication is unsupported in this configuration.
- Do not use local RADIUS authentication on the controller for FlexConnect local authentication-enabled WLANs; it is not supported.
- Once the client has been authenticated, roaming is only supported after the controller and the other FlexConnect access points in the group are updated with the client information.

Local and backup RADIUS server configuration

-
- When connected to a central controller, FlexConnect APs use the controller's primary RADIUS servers in the specified order unless overridden for a particular WLAN. The order is specified on the **RADIUS Authentication Servers** window or using the the **config radius auth add** command
- In standalone mode, each FlexConnect AP must have its own backup RADIUS server to perform 802.1X EAP authentication for clients. The controller itself does not use a backup RADIUS server in this mode.

- You can configure a backup RADIUS server for individual FlexConnect APs in standalone mode by using the controller CLI or for groups of FlexConnect APs in standalone mode by using either the GUI or CLI. An AP-specific backup RADIUS configuration overrides any group configuration.

> **Note**
>
> A controller does not use a backup RADIUS server. The controller uses the backup RADIUS server in local authentication mode.

WLAN authentication and switching states

When a primary RADIUS server becomes unavailable, WLANs will enter either:

- **Authentication down, switching down**

  state if the WLAN was configured for central switching
- **Authentication down, local switching** state if the WLAN was configured for local switching

Web authentication and DNS handling

When web-authentication is used on FlexConnect APs at a remote site, the clients get the IP address from the remote local subnet.

To resolve the initial URL request, the DNS is accessible through the subnet's default gateway. In order for the controller to intercept and redirect the DNS query return packets, these packets must reach the controller at the data center through a CAPWAP connection. During the web-authentication process, the FlexConnect APs allows only DNS and DHCP messages; the APs forward the DNS reply messages to the controller before web-authentication for the client is complete. After web-authentication for the client is complete, all the traffic is switched locally.

Restrictions

- OTAP is not supported.
- After the AP reboots with the new controller software, convert it to FlexConnect mode.
- 802.1X authentication on the AUX port is unsupported for Cisco Aironet 2700 series FlexConnect APs.
- FlexConnect passive client mode disables IP Learn timeout by default in local switching, central authentication deployments.
- When a FlexConnect AP enters standalone mode, only WLANs configured for open, shared, WPA-PSK, or WPA2-PSK authentication support local authentication for new clients; 802.1X types require an external or local RADIUS server.
- When FlexConnect APs are connected to the controller (rather than in standalone mode), the controller uses its primary RADIUS servers and accesses them in the order specified on the RADIUS Authentication Servers page or in the **config radius auth add** CLI command (unless the server order is overridden for a particular WLAN). However, to support 802.1X EAP authentication, FlexConnect APs in standalone mode need to have their own backup RADIUS server to authenticate clients.

> **Note**
>
> A controller does not use a backup RADIUS server. The controller uses the backup RADIUS server in local authentication mode.

When the FlexConnect AP cannot reach its default gateway through ARP and cannot discover the controller, it attempts multiple DHCP renewals and reboots if configured with a static IP to recover connectivity; controller discovery failures trigger fallback behavior.

If the AP fails to establish the ARP, this happens:

- The AP attempts to discover for five times and if it still cannot find the controller, it tries to renew the DHCP on the ethernet interface to get a new DHCP IP.
- The AP will retry for five times, and if that fails, the AP will renew the IP address of the interface again, this will happen for three attempts.
- If the three attempts fail, the AP will fall back to the static IP and will reboot (only if the AP is configured with a static IP).
- The AP reboots to eliminate potential configuration errors.

Guidelines and restrictions for FlexConnect

FlexConnect guidelines and restrictions.

Configuration Changes

- When you apply a configuration change to a locally switched WLAN, the access point resets the radio, causing associated client devices to disassociate, including those not associated with the modified WLAN. Modify the configuration only during a maintenance window. This is applicable when a centrally switched WLAN is changed to a locally switched WLAN.
- This guideline is specific to Wave 1 APs, and not for Wave 2 APs or 11AX APs.

VLAN and Switched WLANs

- FlexConnect mode can support only 16 VLANs per AP.
- NAC out-of-band integration is supported only on WLANs configured for FlexConnect central switching, not for local switching.
- FlexConnect access points with locally switched WLANs cannot perform IP source guard and prevent ARP spoofing.

Network and Client Requirements

- You can deploy a FlexConnect access point with either a static IP address or a DHCP address. Ensure a DHCP server is available locally and able to provide the IP address for the access point at bootup.
- FlexConnect supports up to 4 fragmented packets or a 576-byte MTU WAN link.
- Round-trip latency must not exceed 300 milliseconds (ms) between the access point and the controller, and CAPWAP control packets must be prioritized over all other traffic.

Roaming and Associations

When a client roams from one AP to another and the roaming is successful, this happens:

- The client does not send any traffic to the new AP.

- The client's state is IP LEARN pending.
- The client is deauthenticated after 180 seconds, if there is no traffic for the entire duration. In case the DHCP Required flag is set, the deauthentication occurs after 60 seconds.

Authentication and Support

- FlexConnect APs do not forward the DHCP packets after Change of Authorization (CoA) and change of VLANs using 802.1X encryption. You must disconnect the client from the WLAN and reconnect the client to enable the client to get an IP address in the second VLAN.
- The configuration on the controller must be the same between the time the access point went into standalone mode and the time the access point came back to connected mode.
- Local authentication fallback is not supported when a user is not available in the external RADIUS server.

Configuration Practices

- In the FlexConnect mode, use a named site tag instead of default-site-tag. If you use default-site-tag, the client Pairwise Master Key (PMK) is not sent to APs. This results in client roam and reassociation issues.

Rate limiting per SSID is not supported in FlexConnect local switching mode

- Rate limiting per SSID is not supported in FlexConnect local switching mode.
- Rate limiting is enforced per flow (5-tuple) and not per SSID or per client.
- For rate limiting to work as expected, FlexConnect central switching mode should be used.
- In FlexConnect local switching mode, QoS policies are applied at the AP level.

Configure a site tag (CLI)

Configure a site tag using commands.

Configure a site tag using CLI to centrally manage configurations for APs within a network. By completing this task, you streamline the management of configuration profiles and associated devices on the network.

Use these steps to configure a site tag using CLI:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure site tag and enter site tag configuration mode.

   Example

   ```
   Device(config)# wireless tag site default-site-tag
   ```

3. Move the AP to FlexConnect mode.

Example

```
Device(config-site-tag)# no local-site
```

> **Note**
>
> **no local-site** must be configured before configuring flex-profile. Otherwise, flex-profile will not be applied to the site tag.

4. Map a flex profile to a site tag.

   Example

   ```
   Device(config-site-tag)# flex-profile rr-xyz-flex-profile
   ```

5. Assign an AP profile to the wireless site.

   Example

   ```
   Device(config-site-tag)# ap-profile xyz-ap-profile
   ```

6. Add a description for the site tag.

   Example

   ```
   Device(config-site-tag)# description "default site tag"
   ```

7. Save the configuration, exit the configuration mode, and return to privileged EXEC mode.

   Example

   ```
   Device(config-site-tag)# end
   ```

8. Display the summary of site tags.

   Example

   ```
   Device# show wireless tag site summary
   ```

You configured the new site tag for the network. Now visible in the system, the site tag allows you to efficiently manage AP profiles and flex profiles associated with specific network sites.

Configure a policy tag (CLI)

Configure a policy tag using commands.

Create and apply a policy tag to group wireless local area network (WLAN) and policy profiles for your network configuration.

Use this task when you need to define or update policy tags for your wireless network devices using the CLI.

Before you begin

- Prepare unique names for policy tags using ASCII characters (32 to 126, no leading or trailing spaces).
- Identify the WLAN and policy profiles you plan to map.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure policy tag and enter policy tag configuration mode.

   Example

   ```
   Device(config-policy-tag)# wireless tag policy default-policy-tag
   ```

   Note

   When performing local web authentication, the clients connected to a controller get disconnected intermittently before session timeout.

3. Add a description to a policy tag.

   Example

   ```
   Device(config-policy-tag)# description "default-policy-tag"
   ```

4. Map a remote-LAN profile to a policy profile.

   Example

   ```
   Device(config-policy-tag)# remote-lan remote-lan-name policy
   profile-policy-name port-id port-id-number
   ```

5. Map a policy profile to a WLAN profile.

   Example

   ```
   Device(config-policy-tag)# wlan wlan-name policy profile-policy-name
   ```

> **Note**
>
> Ensure that the WLAN profile is not used by any other profiles. If the AP uses the default profile, ensure that the **no central switching** command is configured on other profiles.

6. Exit policy tag configuration mode, and return to privileged EXEC mode.

Example

```
Device(config-policy-tag)# end
```

7. Optional: Display the configured policy tags.

Example

```
Device# show wireless tag policy summary
```

> **Note**
>
> To view detailed information about a policy tag, use the **show wireless tag policy detailed** *policy-tag-name* command.

Your device has the new policy tag applied. The mapped WLAN and policy profiles are now active based on your configuration.

What to do next

Verify that connected devices use the updated policy tag and the expected network policies are applied.

Attach policy and site tags to an access point (GUI)

Attach a policy tag and a site tag to an AP using GUI.

This task allows you to efficiently organize and manage access points by assigning specific policy and site tags through GUI.

Use these steps to assign policy and site tags to an AP using GUI:

Procedure

---

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click the **Access Point** name.
3. Go to the **Tags** section.
4. Choose the **Policy Tag** from the **Policy** drop-down list.
5. Choose the **Site Tag** from the **Site** drop-down list.

6. Click **Update and Apply to Device**.

The AP has the designated policy and site tags, ensuring it operates under defined network conditions and policy settings.

Attach policy tag and site tag to an AP (CLI)

Attach Policy Tag and Site Tag to an AP using commands.

Assign a policy tag and site tag to an AP using the CLI.

Use this procedure to associate specific network policies and locations with an AP in your Cisco wireless deployment.

Before you begin

Make sure you have the wired MAC address of the AP.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a Cisco AP and enters AP profile configuration mode.

   Example

   ```
   Device(config)# ap F866.F267.7DFB
   ```

   Note

   The *mac-address* should be a wired mac address.

3. Map a policy tag to the AP.

   Example

   ```
   Device(config-ap-tag)# policy-tag rr-xyz-policy-tag
   ```

4. Map a site tag to the AP.

   Example

   ```
   Device(config-ap-tag)# site-tag rr-xyz-site
   ```

5. Associate the RF tag.

Example

```
Device(config-ap-tag)# rf-tag rf-tag1
```

6. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

Example

```
Device(config-ap-tag)# end
```

7. Optional: Display AP details and the tags associated to it.

Example

```
Device# show ap tag summary
```

8. Display the AP name with tag information.

Example

```
Device# show ap name "ap-name" tag info
```

9. Optional: Display the AP name with tag details.

Example

```
Device# show ap name ap-name tag detail
```

The AP is now associated with the specified policy, site, and optionally, RF tags you selected

Link an ACL policy to the defined ACL (GUI)

Link an ACL policy to the defined ACL using GUI.

The task of linking an ACL policy to a defined ACL using the GUI enables you to assign security rules to specific network traffic based on ACL configurations.

Use these steps to link an ACL policy to a defined ACL using the GUI

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Flex**.
2. Click **Add**.
3. In the **General** tab, enter the **Name** of the Flex Profile. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.
4. In the **Policy ACL** tab, click **Add**.
5. Select the ACL from the **ACL Name** drop-down list and click **Save**.
6. Click **Apply to Device**.

The ACL policy is linked to the defined ACL and applied to the specified device.

Apply access control lists on FlexConnect

Apply ACLs on FlexConnect using commands.

Apply Access Control Lists (ACLs) on a FlexConnect wireless profile to filter packet movement through a network. Use these steps to apply ACLs on FlexConnect.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a wireless flex profile and enter wireless flex profile configuration mode.

   Example

   ```
   Device(config)# wireless profile flex Flex-profile-1
   ```

3. Configure an ACL policy.

   Example

   ```
   Device(config-wireless-flex-profile)# acl-policy ACL1
   ```

   Access control lists (ACLs) perform packet filtering to control the movement of packets through a network.

4. Return to wireless flex profile configuration mode.

   Example

   ```
   Device(config-wireless-flex-profile-acl)# exit
   ```

5. Configure native vlan-id information.

   Example

   ```
   Device(config-wireless-flex-profile)# native-vlan-id 25
   ```

6. Configure a VLAN.

   Example

   ```
   Device(config-wireless-flex-profile)# vlan-name VLAN0169
   ```

7. Configure an ACL for the interface.

   Example

   ```
   Device(config-wireless-flex-profile-vlan)# acl ACL1
   ```

8. Configure VLAN information.

Example

```
Device(config-wireless-flex-profile-vlan)# vlan-id 169
```

The ACLs are applied to the FlexConnect wireless profile, ensuring controlled packet filtering through configured VLAN settings.

Configure FlexConnect

Details FlexConnect configuration processes, including remote site switch setup, controller configuration, local and central switching (via GUI and CLI), AP configuration for FlexConnect and local authentication, client WLAN connectivity, Ethernet fallback, and authentication scenarios with deployment examples.

Configure the switch at a remote site

Configure a switch at a remote site.

Configure a switch to support a FlexConnect access point at a remote site by ensuring proper VLAN and IP address settings.

Use the steps to configure the switch at a remote site

Procedure

1. Attach the AP by connecting the FlexConnect access point to either a trunk or an access port on the switch.

> **Note**
>
> The sample configuration in this procedure shows the FlexConnect access point connected to a trunk port on the switch.

2. This example configuration guides you on configuring a switch to support a FlexConnect AP.

   In this sample configuration, the FlexConnect access point is connected to the trunk interface FastEthernet 1/0/2 with native VLAN 100. The access point needs IP connectivity on the native VLAN. The remote site has local servers or resources on VLAN 101. A DHCP pool is created in the local switch for both the VLANs in the switch. The first DHCP pool (NATIVE) is used by the FlexConnect access point, and the second DHCP pool (LOCAL-SWITCH) is used by the clients when they associate to a WLAN that is locally switched.

```
.
.
.
ip dhcp pool NATIVE
   network 209.165.200.224 255.255.255.224
   default-router 209.165.200.225
   dns-server 192.168.100.167
!
ip dhcp pool LOCAL-SWITCH
```

```
   network 209.165.201.224 255.255.255.224
   default-router 209.165.201.225
   dns-server 192.168.100.167
!
interface Gig1/0/1
 description Uplink port
 no switchport
 ip address 209.165.202.225 255.255.255.224
!
interface Gig1/0/2
 description the Access Point port
 switchport trunk encapsulation dot1q
 switchport trunk native vlan 100
 switchport trunk allowed vlan 101
 switchport mode trunk
!
interface Vlan100
 ip address 209.165.200.225 255.255.255.224
!
interface Vlan101
 ip address 209.165.201.225 255.255.255.224
end
!
.
.
.
```

The switch is configured to support the FlexConnect access point, enabling network connectivity for the access point and local servers or resources in the VLANs specified.

Configure the controller for FlexConnect

Configure the controller for FlexConnect.

You can configure the controller for FlexConnect in either centrally switched WLAN or locally switched WLAN environments.

Table 54: WLAN scenarios

The controller configuration for FlexConnect consists of creating centrally switched and locally switched WLANs. This table shows three WLAN scenarios.

WLAN	Security	Authentication	Switching	Interface mapping (GUEST VLAN)
Employee	WPA1+WPA2	Central	Central	Management (centrally switched GUEST VLAN)
Employee-local	WPA1+WPA2 (PSK)	Local	Local	101 (locally switched GUEST VLAN)
Guest-central	Web authentication	Central	Central	Management (centrally switched GUEST VLAN)
Employee-local-auth	WPA1+WPA2	Local	Local	101 (locally switched VLAN)

Configure local switching in FlexConnect mode (GUI)

Configure local switching in FlexConnect mode using GUI.

Enable local switching for a device operating in FlexConnect mode using the GUI.

Use these steps to configure local switching.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. On the **Policy Profile** page, click the name of a policy profile to edit it or click **Add** to create a new one.
3. In the **Add/Edit Policy Profile** window that is displayed, uncheck the **Central Switching** check box.
4. Click **Update & Apply to Device**.

The device is now configured to use local switching in FlexConnect mode.

Configure local switching in FlexConnect mode (CLI)

Configure local switching in FlexConnect mode using commands.

Configure a WLAN for local switching when operating in FlexConnect mode, enabling WLANs to be locally switched at the AP.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile
   policy rr-xyz-policy-1
   ```

3. Configure the WLAN for local switching.

   Example

   ```
   Device(config-wireless-policy)# no central switching
   ```

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The WLAN operates with local switching at the AP, allowing the WLAN data traffic to be processed locally rather than routed through a central controller.

Configure central switching in FlexConnect mode (GUI)

Configure central switching in FlexConnect mode using GUI.

Enable or disable central switch mode in FlexConnect to manage traffic more effectively based on your network setup.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. On the **Policy Profile** page, select a policy.
3. In the **Edit Policy Profile** window, in General Tab, use the slider to enable or disable **Central Switching**.
4. Click **Update & Apply to Device**.

Central switch mode has been configured as specified, and the network policy is updated according to your current setup needs.

Configure central switching in FlexConnect mode (CLI)

Configure central switching in FlexConnect mode using commands.

Establish central switching in FlexConnect mode on your device using the CLI.

Use these steps to configure central switching in FlexConnect mode.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device
   config)# wireless profile policy rr-xyz-policy-1
   ```

3. Configure the WLAN for central switching.

   Example

   ```
   Device config-wireless-policy)# central switching
   ```

4. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

Central switching configuration is applied, and data from devices is centrally processed in FlexConnect mode.

Configure an access point for FlexConnect

Configure an AP for FlexConnect.

For more information, see *Configuring a Site Tag (CLI)* topic in New Configuration Model chapter.

Configure an access point for local authentication on a WLAN (GUI)

Configure an AP for local authentication on a WLAN using GUI.

Configure an AP so that it uses local authentication for wireless LANs, enhancing the security and autonomy of network access control processes.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. In the **Policy Profile** page, select a policy profile name. The **Edit Policy Profile** window is displayed.
3. In the General tab, deselect **Central Authentication** check box.
4. Click **Update & Apply to Device**.

The AP is set up to authenticate users locally without relying on central authentication systems, providing secure and efficient network access verification.

Configure an access point for local authentication on a WLAN (CLI)

Configure an AP for local authentication on a WLAN using commands.

Configure an AP to use local authentication on a WLAN to enhance security control at the network edge.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile
   policy rr-xyz-policy-1
   ```

3. Configure the WLAN for local authentication.

Example

```
Device(config-wireless-policy)# no central authentication
```

4. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

The AP is configured to authenticate WLAN users locally, bypassing central authentication mechanisms.

Connect client devices to WLANs

Connect client devices to WLANs.

A client device connection to a WLAN is a profile creation process that

- allows client devices to connect to wireless networks
- requires specific authentication methods, and
- assigns IP addresses upon successful authentication.

Additional Reference Information

Refer to the instructions for your client device to create profiles to connect to the WLANs you created. These instructions are specified in the Configuring the vEWLC for FlexConnect document.

Example Scenarios:

1. Employee WLAN: Create a client profile that uses WPA or WPA2 with PEAP-MSCHAPV2 authentication. After the client is authenticated, the management VLAN of the embedded controller assigns an IP address to the client.
2. Local-Employee WLAN: Create a client profile that uses WPA or WPA2 authentication. After the client is authenticated, the client is allotted an IP address by VLAN 101 on the local switch.
3. Guest-Central WLAN: Create a client profile that uses open authentication. After the client is authenticated, the client is allocated an IP address by VLAN 101 on the network local to the access point. After the client connects, a local user can enter any HTTP address in the web browser. The user is automatically directed to the controller to complete the web authentication process. When the web login window appears, the user should enter the username and password.

> **Note**
>
> Ensure that the authentication settings are configured correctly for each client profile.

FlexConnect ethernet fallback

FlexConnect Ethernet Fallback details.

A FlexConnect Ethernet Fallback is a configuration feature that

- allows the AP to shut down its radio when the Ethernet link is non-operational

- enables the AP to set its radio back to operational state when the Ethernet link is restored, and
- operates independently of the AP being in connected or standalone mode.

To prevent radios from flapping when there is Ethernet interface instability, a configurable delay timer is provided.

Configure FlexConnect ethernet fallback (CLI)

Configure FlexConnect ethernet fallback using commands.

Use CLI to configure the FlexConnect Ethernet fallback on specific APs to ensure network reliability in case of port failover.

Before you begin

This feature is not applicable to APs with multiple ports.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a wireless flex profile and enter wireless flex profile configuration mode.

   Example

   ```
   Device(config)# wireless profile flex test
   ```

3. Enable radio interface shutdown.

   Example

   ```
   Device(config-wireless-flex-profile)# fallback-radio-shut
   ```

4. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-flex-profile)# end
   ```

5. Optional: Display detailed information about the selected profile.

   Example

   ```
   Device# show wireless profile flex detailed test
   ```

The FlexConnect Ethernet fallback is configured, ensuring that the radio interface shuts down during Ethernet failure, maintaining network continuity.

Configure FlexConnect AP local authentication (GUI)

Configure Flex AP local authentication using the GUI interface.

Configure the local authentication settings on a FlexConnect AP using the GUI to enable authentication and client handling directly on the AP.

Use these steps to configure FlexConnect AP local authentication:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Flex**.
2. In the **Flex** page, click the name of the **Flex Profile** or click **Add** to create a new one.
3. In the **Add/Edit Flex Profile** window, click the **Local Authentication** tab.

   When you enable local authentication and association on the Access Point with Flex mode, these outcomes occur:

   - AP handles the authentication.
   - AP handles the rejection of client joins (in Mobility).

   > **Note**
   >
   > You will not receive updated statistics from the controller when the AP rejects client associations.

4. Choose the server group from the **RADIUS Server Group** drop-down list.
5. Use the **Local Accounting Radius Server Group** drop-down to select the RADIUS server group.
6. Check the **Local Client Roaming** check box to enable client roaming.
7. Choose the profile from the **EAP Fast Profile** drop-down list.
8. Choose to enable or disable these:

   - **LEAP:** Lightweight Extensible Authentication Protocol (LEAP) is an 802.1X authentication type for wireless LANs and supports strong mutual authentication between the client and a RADIUS server using a logon password as the shared secret. It provides dynamic per-user, per-session encryption keys.
   - **PEAP:** Protected Extensible Authentication Protocol (PEAP) is a protocol that encapsulates the Extensible Authentication Protocol (EAP) within an encrypted and authenticated Transport Layer Security (TLS) tunnel.
   - **TLS:** Transport Layer Security (TLS) is a cryptographic protocol that provides communications security over a computer network.
   - **RADIUS:** Remote Authentication Dial-In User Service (RADIUS) is a networking protocol that provides centralized Authentication, Authorization, and Accounting (AAA or Triple A) management for users who connect and use a network service.

9. In the **Users** section, click **Add**.
10. Enter username and password details and click **Save**.
11. Click **Save & Apply to Device**.

The AP is configured to handle local authentication requests, enabling improved client management.

Configure FlexConnect access point local authentication

Flex AP local authentication using commands.

You enable FlexConnect APs local authentication, allowing user authentication directly at the AP level using RADIUS profiles and methods.

> **Note**
>
> The Cisco Catalyst 9800 Series Wireless Controller + FlexConnect local authentication + AP acting as RADIUS are not supported on Cisco COS and IOS APs.

Procedure

1. Create a AAA authentication model.

   Example

   ```
   Device(config)# aaa new-model
   ```

2. Send session ID information from the RADIUS group for a given call.

   Example

   ```
   Device(config)# aaa session-id common
   ```

3. Enable system authorization control for the RADIUS group.

   Example

   ```
   Device(config)# dot1x system-auth-control
   ```

4. Create an EAP profile.

   Example

   ```
   Device(config)# eap profile aplocal-test
   ```

5. Configure the FAST method on the profile.

   Example

   ```
   Device(config-eap-profile)# method fast
   ```

6. Return to configuration mode.

   Example

   ```
   Device(config-radius-server)# exit
   ```

7. Configure the flex policy.

Example

```
Device(config)# wireless profile flex default-flex-profile
```

8. Configure EAP-FAST profile details.

Example

```
Device(config-wireless-flex-profile)# local-auth ap eap-fast aplocal-test
```

9. Configure LEAP method.

Example

```
Device(config-wireless-flex-profile)# local-auth ap leap
```

10. Configure the PEAP method.

Example

```
Device(config-wireless-flex-profile)# local-auth ap peap
```

11. Configure DHCP broadcast for locally switched clients.

Example

```
Device(config-wireless-flex-profile)# dhcp broadcast
```

12. Configure username and password.

Example

```
Device(config-wireless-flex-profile)# local-auth ap username username password
```

13. Configure another username and password.

Example

```
Device(config-wireless-flex-profile)# local-auth ap username username password
```

14. Return to configuration mode.

Example

```
Device(config-wireless-flex-profile)# exit
```

15. Configure profile policy.

Example

```
Device(config)# wireless profile policy default-policy-profile
```

16. Disable the policy profile.

Example

```
Device(config-wireless-policy)# shutdown
```

17. Disable central authentication.

Example

```
Device(config)# no central authentication
```

18. Configure VLAN name or VLAN ID.

Example

```
Device(config)# vlan-id 54
```

19. Enable the configuration.

Example

```
Device(config)# no shutdown
```

You can now authenticate users locally with FlexConnect APs by using specified EAP methods and profiles, operating under defined policy configurations.

Configure FlexConnect access point local authentication with external RADIUS server

FlexConnect access Point local authentication with external radius server using commands.

Set up local authentication on a FlexConnect access point using an external RADIUS server.

In this mode, an AP handles client authentication and switches client data packets locally. This state is valid in standalone mode and connected mode.

Use these steps to create and configure FlexConnect AP local authentication with a RADIUS server:

Procedure

---

1. Create a AAA authentication model.

Example

```
Device(config)# aaa new-model
```

2. Send session ID information from the RADIUS group for a given call.

Example

```
Device(config)# aaa session-id common
```

3. Enable the system authorization control for the RADIUS group.

Example

```
Device(config)# dot1x system-auth-control
```

4. Specify the RADIUS server name.

Example

```
Device(config)# radius server Test-SERVER1
```

> **Note**
>
> To authenticate clients with freeradius over RADSEC, you should generate an RSA key longer than 1024 bit. Use the **crypto key generate rsa general-keys exportable label** *name* command to achieve this.
>
> Do not configure **key-wrap** option under the radius server and radius server group, as it may lead to clients getting stuck in authentication state.

5. Specify the primary RADIUS server parameters.

Example

```
Device(config-radius-server)# address ipv4 ip-address address auth-port
port-number acct-port port-number
```

```
Device(config-radius-server)# address ipv6 ip-address address auth-port
port-number acct-port port-number
```

6. Specify the authentication and encryption key used between the device and the RADIUS daemon running on the RADIUS server.

Example

```
Device(config-radius-server)# key test123
```

> **Note**
>
> The maximum number of characters allowed for the shared secret is 63.

7. Specify the RADIUS server name.

Example

```
Device(config)# radius server Test-SERVER2
```

8. Specify the secondary RADIUS server parameters.

Example

```
Device(config-radius-server)# address ipv4 124.3.52.62 auth-port 1112 acct-port
 1113
```

```
Device(config-radius-server)# address ipv6 2001:DB8:0:21::15 auth-port 1812
acct-port 1813
```

9. Specify the authentication and encryption key used between the device and the RADIUS daemon running on the RADIUS server.

   Example

   ```
   Device(config-radius-server)# key test113
   ```

10. Return to configuration mode.

    Example

    ```
    Device(config-radius-server)# exit
    ```

11. Create a RADIUS server group identification.

    Example

    ```
    Device(config)# aaa group server radius aaa_group_name
    ```

    Note

    server-group refers to the server group name. The valid range is from 1 to 32 alphanumeric characters.

12. Specify the RADIUS server name.

    Example

    ```
    Device(config)# radius server Test-SERVER1
    ```

13. Specify the RADIUS server name.

    Example

    ```
    Device(config-radius-server)# radius server Test-SERVER2
    ```

14. Exit from RADIUS server configuration mode.

    Example

    ```
    Device(config-radius-server)# exit
    ```

15. Create a new flex policy.

Example

```
Device(config)# wireless profile flex default-flex-profile
```

16. Configure the authentication server group name.

Example

```
Device(config-wireless-flex-profile)# local-auth radius-server-group
aaa_group_name
```

17. Return to configuration mode.

Example

```
Device(config-wireless-flex-profile)# exit
```

18. Configure a WLAN policy profile.

Example

```
Device(config)# wireless profile policy default-policy-profile
```

19. Disable a policy profile.

Example

```
Device(config-wireless-policy)# shutdown
```

20. Disable central authentication.

Example

```
Device(config-wireless-policy)# no central authentication
```

21. Configure a VLAN name or VLAN Id.

Example

```
Device(config-wireless-policy)# vlan-id 54
```

22. Enable the configuration.

Example

```
Device(config-wireless-policy)# no shutdown
```

The FlexConnect AP is now set up for local authentication using the specified RADIUS server parameters.

Configuration example: FlexConnect with central and local authentication

FlexConnect with central and local authentication configuration example.

To see a configuration example on how to configure a controller for FlexConnect central and local authentication, refer to FlexConnect Configuration with Central and Local Authentication on Catalyst 9800 Wireless Controllers.

NAT-PAT functionalities in FlexConnect

Explains NAT-PAT implementation for FlexConnect, outlining procedures for WLAN and remote LAN setup, wireless profile and policy configuration, site tag deployment, and mapping policy and site tags to access points using both GUI and CLI interfaces.

NAT-PAT for FlexConnect is a networking function that:

- enables the use of a central DHCP server for assigning IP addresses to clients across remote sites
- involves an AP translating client traffic by replacing the private IP address with its own public IP address, and
- supports efficient management of IP address allocation.

If implementing NAT and PAT for flexibly managed networks, enable local switching and configure central DHCP. When ensuring DHCP service, use the **ipv4 dhcp required** command.

Create a WLAN

Create a WLAN using commands.

Configure and enable a WLAN using command line inputs, ensuring it is active and ready for use.

Use these steps to create a WLAN.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-demo 1 ssid-demo
   ```

   - *wlan-name*–Enter the profile name. The range is from 1 to 32 alphanumeric characters.
   - *wlan-id*–Enter the WLAN ID. The range is from 1 to 512.
   - *SSID-name*–Enter the Service Set Identifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID.

> **Note**
>
> If you have already configured a WLAN, use the *wlan wlan-name* command.

3. Shut down the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

4. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

The WLAN is successfully configured and activated, allowing devices to connect using the specified SSID.

Configure a wireless profile policy and NAT-PAT (GUI)

Configure a wireless profile policy and NAT-PAT using GUI.

Define and apply a wireless profile policy and NAT-PAT settings using GUI.

Procedure

1. Navigate to **Configuration** > **Tags & Profiles** > **Policy**.
2. Click **Add** to create a new policy.
3. In the **General** tab, enter the **Name** of the policy.
4. Disable the **Central Switching** toggle button.
5. Enable the **Central DHCP** toggle button.
6. Enable the **Flex NAT/PAT** toggle button.
7. In the **Advanced** tab, under the **DHCP** Settings, check the **IPv4 DHCP Required** check box.
8. Apply the configuration by selecting **Apply to Device**.

The configuration of the wireless profile policy and NAT-PAT settings is complete.

Configure a wireless profile policy and NAT-PAT (CLI)

Configure a wireless profile policy and NAT-PAT using commands.

Configure a wireless profile policy and enable NAT-PAT settings for a device using CLI.

Use the steps here to configure a wireless profile policy and NAT-PAT.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the policy profile for NAT.

   Example

   ```
   Device(config)# wireless profile policy nat-enabled-policy
   ```

3. Configure the WLAN for local switching.

   Example

   ```
   Device(config-wireless-policy)# no central switching
   ```

4. Configure the DHCP parameters for WLAN.

   Example

   ```
   Device(config-wireless-policy)# ipv4 dhcp required
   ```

5. Configure the central DHCP for locally switched clients.

   Example

   ```
   Device(config-wireless-policy)# central dhcp
   ```

6. Enable NAT-PAT.

   Example

   ```
   Device(config-wireless-policy)# flex nat-pat
   ```

7. Enable policy profile.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

8. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

Wireless profile policy is configured and NAT-PAT is activated, which facilitates network traffic management and enables efficient packet handling on your device.

Map a WLAN to a policy profile (CLI)

Map a WLAN to a Policy Profile using commands.

Enable seamless network management by mapping a WLAN to a designated policy profile through CLI.

Use these steps to map a WLAN to a policy profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a policy tag and enter policy tag configuration mode.

   Example

   ```
   Device(config)# wireless tag policy demo-tag
   ```

3. Map a policy profile to a WLAN profile.

   Example

   ```
   Device(config-policy-tag)# wlan wlan-demo policy nat-enabled-policy
   ```

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-policy-tag)# end
   ```

WLAN is mapped to the specified policy profile, ensuring the application of the required network policies.

Configure a site tag

Configure a site tag using commands.

Configure a site tag to enhance management and control of your wireless network.

Use these steps to configure a site tag:

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a site tag and enter site tag configuration mode.

Example

```
Device(config)# wireless tag site flex-site
```

3. Move the AP to FlexConnect mode.

Example

```
Device(config-site-tag)# no local-site
```

4. Return to privileged EXEC mode.

Example

```
Device(config-site-tag)# end
```

The FlexConnect mode is configured onto the assigned AP, enhancing network flexibility and management.

Attach policy and site tags to an access point (GUI)

Attach a policy tag and a site tag to an AP using GUI.

This task allows you to efficiently organize and manage access points by assigning specific policy and site tags through GUI.

Use these steps to assign policy and site tags to an AP using GUI:

Procedure

---

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click the **Access Point** name.
3. Go to the **Tags** section.
4. Choose the **Policy Tag** from the **Policy** drop-down list.
5. Choose the **Site Tag** from the **Site** drop-down list.
6. Click **Update and Apply to Device**.

The AP has the designated policy and site tags, ensuring it operates under defined network conditions and policy settings.

Attach a policy tag and a site tag to an access point (CLI)

Attach a policy tag and a site tag to an AP using commands.

Apply network policy and site tags to an AP using commands.

Use these steps to attach a policy tag and a site tag to an AP:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure APs and enter ap-tag configuration mode.

   Example

   ```
   Device(config)# ap F866.F267.7DFB
   ```

3. Map the policy tag to the AP.

   Example

   ```
   Device(config-ap-tag)# policy-tag demo-tag
   ```

4. Map the site tag to the AP.

   Example

   ```
   Device(config-ap-tag)# site-tag flex-site
   ```

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-tag)# end
   ```

The AP has the specified policy and site tags applied, ready for network reconfiguration.

Split tunneling for FlexConnect

Describes split tunneling capabilities for FlexConnect, detailing restrictions, access control list definition, wireless profile and policy configuration, WLAN creation, mapping WLANs to policy profiles, site tag assignment, and procedures for deploying split tunneling on WLANs and remote LANs.

Split tunneling is a network feature that:

- minimizes unnecessary bandwidth consumption on WAN links
- allows traffic classification based on packet contents for local switching, and
- ensures efficient routing of data by distinguishing between local and central switching requirements.

If a client connects over a WAN link associated with a centrally switched WLAN, traffic intended for a device present in the local site is typically sent over CAPWAP to the controller, then back to the local site over CAPWAP or via some off-band connectivity. This consumes WAN link bandwidth unnecessarily. The split tunneling feature mitigates this by classifying client traffic based on packet contents. Matching packets are locally switched, while the rest are centrally switched.

Configuration details

To configure local split tunneling on an AP, ensure that you have enabled DHCP Required on the policy profile using the **ipv4 dhcp required** command. This ensures the client associating with the split WLAN performs DHCP.

Restriction: split tunneling for FlexConnect

Restriction for split tunneling for FlexConnect

- Ensure Apple iOS clients receive option 6 (DNS) in the DHCP offer for split tunneling to function correctly.
- VLAN-based central switching for FlexConnect in auto-anchor deployment is not supported.
- You cannot use split tunneling with RLAN clients. When the **split-tunnel** option is enabled on RLAN, traffic denied by the split tunnel ACL is not translated based on the IP address, instead the traffic is sent back to the controller through CAPWAP.
- Do not configure URL filters with wildcard URLs such as * and ".".

Configure Split Tunneling for a WLAN or Remote LAN

-

Define an access control list for split tunneling (GUI)

Define an ACL for Split Tunneling using GUI.

Define an ACL for split tunneling.

Use these steps to define an ACL for split tunneling in the GUI.

Procedure

1. Choose **Configuration** > **Security** > **ACL**.
2. Click **Add**.
3. In the **Add ACL Setup** dialog box, enter the **ACL Name**.
4. Choose the ACL type from the **ACL Type** drop-down list.
5. Under the **Rules** settings, enter the **Sequence** number and choose the **Action** as either **permit** or **deny**.
6. Choose the required source type from the **Source Type** drop-down list.

If...	Then...
Source type is **Host**	Enter the **Host Name/IP**
Source type is **Network**	Specify the **Source IP** address and **Source Wildcard** mask

7. Check the **Log** check box if you want the logs.
8. Click **Add**.
9. Add the rest of the rules and click **Apply to Device**.

The ACL is defined and applied to the specified device for the purpose of split tunneling. You can view the rules in the device's ACL configuration.

Define an ACL for split tunneling (CLI)

Define an ACL for split tunneling using commands.

Define an ACL for split tunneling to manage traffic effectively between local and remote networks, improving network performance and security.

Use these steps to create an ACL for split tunneling.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Define an extended IPv4 access list using a name, and enter access-list configuration mode.

   Example

   ```
   Device(config)# ip access-list extended split_mac_acl
   ```

3. Allow the traffic to switch centrally.

   Example

   ```
   Device(config-ext-nacl)# deny ip any host 9.9.2.21
   ```

4. Allow the traffic to switch locally.

   Example

   ```
   Device(config-ext-nacl)# permit ip any any
   ```

5. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-ext-nacl)# end
   ```

The ACL selectively allows local or central switching of network traffic, enhancing performance and security management.

Link an ACL policy to the defined ACL (CLI)

Link an ACL policy to the defined ACL using commands.

This task provides the steps necessary to associate an ACL policy with a defined ACL, enhancing the ability to manage and control network traffic according to specified security parameters.

Use these steps to link an ACL policy to the defined ACL.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the Flex profile and enter flex profile configuration mode.

   Example

   ```
   Device(config)# wireless profile flex flex-profile
   ```

3. Configure an ACL policy for the defined ACL.

   Example

   ```
   Device(config-wireless-flex-profile)# acl-policy acl-policy-name
   ```

4. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-flex-profile)# end
   ```

The ACL policy is linked, allowing the defined ACL to enforce specified access control rules on network traffic.

Create a WLAN

Create a WLAN using commands.

Configure and enable a WLAN using command line inputs, ensuring it is active and ready for use.

Use these steps to create a WLAN.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the WLAN name and ID

   Example

   ```
   Device(config)# wlan wlan-demo 1 ssid-demo
   ```

   - *wlan-name*–Enter the profile name. The range is from 1 to 32 alphanumeric characters.
   - *wlan-id*–Enter the WLAN ID. The range is from 1 to 512.

- *SSID-name*–Enter the Service Set Identifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID.

3. Shut down the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

4. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

The WLAN is successfully configured and activated, allowing devices to connect using the specified SSID.

Configure a wireless profile policy and a split MAC ACL name (GUI)

Configure a Wireless Profile Policy and a Split MAC ACL Name using GUI.

Configure a wireless profile policy and apply a split MAC ACL name to optimize resource allocation and traffic management.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click **Add**.
3. In the **General** tab, enter the **Name** of the policy.
4. Enable the **Central Switching** toggle button.
5. Enable the **Central DHCP** toggle button.
6. In the **Advanced** tab, under the **DHCP** settings, check the **IPv4 DHCP Required** check box and enter the **DHCP Server IP Address**.
7. Under the **WLAN Flex Policy** settings, choose the split MAC ACL from the **Split MAC ACL** drop-down list.
8. Click **Apply to Device**.

The wireless profile policy and split MAC ACL name are configured and applied to the device, ensuring coordinated network resource management.

Configure a wireless profile policy and a split MAC ACL name

Configure a wireless profile policy and a split MAC ACL name using commands.

You will establish a secure wireless network environment by configuring a wireless profile policy and a split MAC ACL name.

Use these steps to configure a wireless profile policy and a split MAC ACL name:

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a WLAN policy profile and enter wireless policy configuration mode.

Example

```
Device(config)# wireless profile policy split-tunnel-enabled-policy
```

3. Configure a split MAC ACL name.

Example

```
Device(config-wireless-policy)# flex split-mac-acl split_mac_acl
```

> **Note**
>
> You must use the same ACL name for linking the flex and the policy profile.

4. Configure WLAN for central switching.

Example

```
Device(config-wireless-policy)# central switching
```

5. Enable central DHCP for centrally switched clients.

Example

```
Device(config-wireless-policy)# central dhcp
```

6. Configure the DHCP parameters for a WLAN.

Example

```
Device(config-wireless-policy)# ipv4 dhcp required
```

7. Optional: Configure the override IP address of the DHCP server.

Example

```
Device(config-wireless-policy)# ipv4 dhcp server 9.1.0.100
```

8. Enable a policy profile.

Example

```
Device(config-wireless-policy)# no shutdown
```

The wireless profile policy is active with a configured split MAC ACL name, ensuring traffic is managed according to the defined policy guidelines.

Map a WLAN to a policy profile (GUI)

Map a WLAN to a policy profile using GUI.

Map a WLAN to its associated policy profile to ensure network policy configurations are enforced.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. Click **Add**.
3. Enter the **Name** of the Tag Policy.
4. Under **WLAN-POLICY Maps** tab, click **Add**.
5. Choose the WLAN Profile from the **WLAN Profile** drop-down list.
6. Choose the Policy Profile from the **Policy Profile** drop-down list.
7. Click the **Tick** Icon.
8. Click **Apply to Device**.

The WLAN is mapped to the desired policy profile by the system, and configuration is enforced on the device.

Map WLAN to a policy profile

Map a WLAN to a policy profile using commands.

Map a WLAN to a policy profile to enhance network management by applying specific policies to WLAN configurations.

Use these steps to map WLAN to a policy profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a policy tag and enter policy tag configuration mode.

   Example

   ```
   Device(config)# wireless tag policy split-tunnel-enabled-tag
   ```

3. Map a policy profile to a WLAN profile.

Example

```
Device(config-policy-tag)# wlan wlan-demo policy split-tunnel-enabled-policy
```

4. Return to privileged EXEC mode.

Example

```
Device(config-policy-tag)# end
```

The WLAN is associated with the policy profile, ensuring compliance with network policies for connected devices.

Configure a site tag

Configure a site tag using commands.

Configure a site tag for split tunneling, optimizing network performance through selective traffic routing.

Use the steps to configure a site tag:

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a site tag and enter site tag configuration mode.

Example

```
Device(config)# wireless tag site flex-site
```

3. Ensure the local site is not configured on the site tag.

Example

```
Device(config-site-tag)# no local-site
```

4. Configure a flex profile.

Example

```
Device(config-site-tag)# flex-profile flex-profile
```

5. Return to privileged EXEC mode.

Example

```
Device(config-site-tag)# end
```

The site tag for split tunneling is successfully configured, providing optimized network traffic routing.

Attach policy and site tags to an AP

Attach policy and site tags to your AP using commands.

Use this task to configure policy and site tags on your AP using CLI.

Use these steps to attach a policy tag and site tag to your AP.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a Cisco AP and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap 188b.9dbe.6eac
   ```

3. Map a policy tag to the AP.

   Example

   ```
   Device(config-ap-tag)# policy-tag split-tunnel-enabled-tag
   ```

4. Map a site tag to the AP.

   Example

   ```
   Device(config-ap-tag)# site-tag flex-site
   ```

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-tag)# end
   ```

The policy and site tags configure the AP with designated network management settings. Verify the mapping by checking the APs configuration status in the system.

VLAN-based central switching for FlexConnect

Details VLAN-based central switching in FlexConnect environments, providing instructions for GUI and CLI configuration to support centralized VLAN switching for wireless deployments.

VLAN-based central switching for FlexConnect is a network configuration method that

- enables traffic redirection to the controller when a VLAN is not defined locally
- supports local switching if the VLAN is present in the AP's database, and
- requires VLANs to be defined on the controller for proper functionality.

Expanded explanation

- In FlexConnect local switching, if the VLAN definition is not available in an AP, the corresponding client does not pass traffic. This scenario is applicable when the AAA server returns the VLAN as part of client authentication.
- When a WLAN is locally switched in FlexConnect and a VLAN is configured on the AP side, the traffic is switched locally. When a VLAN is not defined in an AP, the VLAN drops the packet.

Special considerations

- The controller forwards the traffic to its corresponding VLAN.
- Ensure that VLAN is defined on the controller for VLAN-based central switching.
- VLAN-based central switching is not supported by mac filter.
- For local switching, ensure that VLAN is defined on both the policy profile and FlexConnect profile.
- VLAN-based central switching with central web authentication enabled in Flex profile is not supported.

Restriction on multiple policy profiles and VLAN behavior

Restriction

If you add multiple policy profiles to a policy tag that uses a VLAN-based, central-switching SSID, do not configure the central switch VLAN in any profile. Avoid using the VLAN ID, and do not define a VLAN name in the Flex profile. Otherwise, the system switches all traffic for that VLAN locally

The reason for this behavior: If the VLAN is present in the access point (AP) database, the system overrides central switching and switches client traffic locally

Configure VLAN-based central switching (GUI)

Configure VLAN-based central switching using GUI.

Enable VLAN-based central switching on a policy profile using the GUI to manage network traffic effectively.

Use these steps to configure VLAN-based central switching.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click the name of the policy profile.

3. In the **Edit Policy Profile** window, perform these tasks:
   - Set **Central Switching** to **Disabled** state.
   - Set **Central DHCP** to **Disabled** state.
   - Set **Central Authentication** to **Enabled** state.
4. Click the **Advanced** tab.
5. Under **AAA Policy**, check the **Allow AAA Override** check box to enable AAA override.
6. Under **WLAN Flex Policy**, check the **VLAN Central Switching** check box to enable VLAN-based central switching on the policy profile.
7. Click **Update & Apply to Device**.

VLAN-based central switching is configured in the policy profile, enabling centralized network traffic management.

Configure VLAN-based central switching (CLI)

Configure VLAN-based central switching using commands.

Configure VLAN-based central switching in a wireless network environment using CLI to enable efficient data forwarding and management.

Use these steps to configure VLAN-based central switching.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a wireless policy profile.

   Example

   ```
   Device(config)# wireless profile policy default-policy-profile
   ```

3. Configure a WLAN for local switching.

   Example

   ```
   Device(config-wireless-policy)# no central switching
   ```

4. Configure local DHCP mode, with DHCP being performed in an AP.

   Example

   ```
   Device(config-wireless-policy)# no central dhcp
   ```

5. Configure a WLAN for central authentication.

Example

```
Device(config-wireless-policy)# central authentication
```

6. Configure AAA policy override.

   Example

   ```
   Device(config-wireless-policy)# aaa-override
   ```

7. Configure VLAN-based central switching.

   Example

   ```
   Device(config-wireless-policy)# flex vlan-central-switching
   ```

8. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

9. Optional: Display detailed information of the policy profile.

   Example

   ```
   Device# show wireless profile policy detailed default-policy-profile
   ```

VLAN-based central-switching is established, optimizing network traffic flow and centralizing control.

OfficeExtend AP for FlexConnect

Outlines features and best practices for OEAP in FlexConnect, including configuration and disabling of OEAPs, support for personal SSIDs, related configuration tasks, and examples of viewing OfficeExtend settings and personal SSID management.

A Cisco OfficeExtend Access Point (OEAP) is a type of wireless access point that

- extends the corporate WLAN over the Internet to remote locations
- ensures secure communication between the controller and access point through DTLS encryption, and
- provides users with a seamless experience comparable to being at a corporate office.

Datagram Transport Layer Security (DTLS) encryption is utilized between the access point and the controller to maintain the highest level of communication security.

Configure OfficeExtend AP

Configure OEAP using commands.

Enable and configure OEAP mode on FlexConnect APs.

Use these steps to configure OEAP.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a wireless flex profile and enter wireless flex profile configuration mode.

   Example

   ```
   Device(config)# wireless profile flex test
   ```

3. Enable the OEAP mode for a FlexConnect AP.

   Example

   ```
   Device(config-wireless-flex-profile)# office-extend
   ```

4. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-flex-profile)# end
   ```

   Note

   After creating a flex profile, ensure that OEAP is in flex connect mode and mapped to its corresponding site tag.

   OfficeExtend is disabled by default. To clear the access point's configuration and return it to the factory-defaults, use the **clear ap config** *cisco-ap* command.

The OEAP is configured and enabled in FlexConnect mode, ready for deployment in a remote office setup.

Disable OfficeExtend AP

Disable OEAP using commands.

Disabling the OEAP mode on a specific FlexConnect AP to optimize wireless network management and security.

Use these steps to disable an OEAP.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a wireless flex profile and enter wireless flex profile configuration mode.

   Example

   ```
   Device(config)# wireless profile flex test
   ```

3. Disable OfficeExtend AP mode for a FlexConnect AP.

   Example

   ```
   Device(config-wireless-flex-profile)# no office-extend
   ```

4. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-flex-profile)# end
   ```

The configured FlexConnect AP is no longer operating in OEAP mode.

Best practices for OfficeExtend AP for FlexConnect

- Preconfigure your controller IP for a zero-touch deployment with OEAP. Configure the local SSID from the AP, allowing other home users to connect using the same AP.
- In releases prior to Cisco IOS XE 17.3.2, when an AP is converted to OEAP, the local DHCP server on the AP is enabled by default. If the DHCP server on the home router has a similar configuration, a network conflict occurs, preventing the AP from rejoining the controller. Change the default DHCP server to resolve this.
- For OEAP, when configuration changes are made from the OEAP GUI to Radio Status, Radio Interface Status, 802.11 n-mode, 802.11 ac-mode, Bandwidth, or Channel Selection (2.4 GHz or 5 GHz), restart CAPWAP to synchronize configurations between the AP and the controller. During this interval, the AP GUI may not respond; it will resume functionality once the AP rejoins the controller. Wait about 1–2 minutes for the AP to rejoin the controller before making further changes from the OEAP GUI.
- In OEAP, if the OEAP local DHCP server is enabled and the user configures DNS IP from OEAP GUI, the wireless and wired clients connected to Cisco OEAP will receive that IP as DNS server IP in DHCP ACK.

OEAP personal SSID support

OEAP Personal SSID Support details.

A personal SSID is a feature of the Cisco OEAP that

- enables local home clients to connect using personal network identifiers
- allows leveraging existing OEAP infrastructure for local connectivity, and
- supports standard security protocols for safe operation.

Additional information

- OEAP supports the enabling or disabling of personal SSID.
- Datagram Transport Layer Security (DTLS) encryption can be enabled or disabled between an access point and the controller.
- Rogue detection can be configured using the controls available on the AP profile page in the GUI.
- The local network access and DTLS encryption are enabled by default.

> **Note**
>
> These configurations are applicable for OEAP or for APs in the OEAP mode.

Configure OEAP personal SSID (GUI)

Configure OEAP personal SSID using GUI.

Setup and configure OEAP personal SSID using GUI for enhanced local network access and security features on AP devices.

Use these steps to configure OEAP personal SSID.

Procedure

1. Choose **Configuration** > **AP Tags & Profiles** > **AP Join**.

   The **AP Join Profile** section displays all the AP Join profiles.

2. To edit the configuration details of an AP Join profile, select APs in the OEAP mode.

   The **Edit AP Join Profile** window is displayed.

3. In the **General** tab, under the **OfficeExtend AP Configuration** section, configure the following:

   Configure the options according to your requirements for local network access, data encryption, and rogue detection.

   Example

   ```
   Configuration settings include enabling Local Access, Link Encryption, and
   Rogue Detection.
   ```

   a) Check the **Local Access** check box to enable the local network. By default, **Local Access** is enabled. After the AP joins the controller using AP join profile where local access is enabled, the AP will not broadcast the default personal SSID. Since the local access is enabled, you can login to the AP GUI and configure the personal SSID.
   b) Check the **Link Encryption** check box to enable data DTLS. By default, **Link Encryption** is enabled.
   c) Check the **Rogue Detection** check box to enable rogue detection. Rogue detection is disabled by default for OfficeExtend APs because these APs, deployed in a home environment, are likely to detect a large number of rogue devices.

The AP is configured with specific OEAP personal SSID settings for local access, encryption, and detection capabilities, ensuring secure and tailored network operations.

Configure OfficeExtend AP personal SSID (CLI)

Configure OEAP personal SSID using commands.

Configure a personal SSID on an OEAP using CLI to enable local access and encryption features.

Use these steps to configure OEAP personal SSID using CLI:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile
   ```

3. Enable the local access to AP.

   Example

   ```
   Device(config-ap-profile)# oeap local-access
   ```

   Local access consists of local AP GUI, LAN ports, and personal SSID. The **no** form of this command disables the feature. If the local access is disabled, you will not be able to access the AP GUI, the local LAN port will be disabled, and personal SSID will not be broadcasted.

4. Enable DTLS encryption for OEAP APs or APs moving to the OEAP mode.

   Example

   ```
   Device(config-ap-profile)# oeap link-encryption
   ```

   The **no** form of this command disables the feature. This feature is enabled by default.

5. Enable OEAP DTLS encryption in the AP profile configuration mode.

   Example

   ```
   Device(config-ap-profile)# no oeap rogue-detection
   ```

   This feature is disabled by default.

The OEAP personal SSID is configured with local access and DTLS encryption enabled, allowing secure connection and management of the AP through the local interface.

View OEAP personal SSID configuration

View OEAP personal SSID configuration.

To view the OEAP personal SSID configuration, run this command:

```
Device# show ap profile name default-ap-profile detailed
.
.
.
OEAP Mode Config
Link Encryption : ENABLED
Rogue Detection : DISABLED
Local Access : ENABLED
```

Clearing personal SSID from an OfficeExtend access point

Clear Personal SSID from an OEAP.

To clear the personal SSID from an access point, run this command:

ap name *Cisco_AP*clear-personal-ssid

Example: viewing OfficeExtend configuration

Viewing OfficeExtend Configuration example.

This example displays an OfficeExtend configuration:

```
Device# show ap config general

Cisco AP Name   : ap_name
=================================================

Cisco AP Identifier                              : 70db.986d.a860
Country Code                                     : Multiple Countries : US,IN
Regulatory Domain Allowed by Country             : 802.11bg:-A   802.11a:-ABDN
AP Country Code                                  : US  - United States
AP Regulatory Domain
  Slot 0                                         : -A
  Slot 1                                         : -D
MAC Address                                      : 002c.c899.7b84
IP Address Configuration                         : DHCP
IP Address                                       : 192.0.2.0
IP Netmask                                       : 255.255.255.0
Gateway IP Address                               : 198.51.100.0
CAPWAP Path MTU                                  : 1485
Telnet State                                     : Disabled
SSH State                                        : Disabled
Jumbo MTU Status                                 : Disabled
Cisco AP Location                                : default location
Site Tag Name                                    : flex-site
RF Tag Name                                      : default-rf-tag
Policy Tag Name                                  : split-tunnel-enabled-tag
AP join Profile                                  : default-ap-profile
Primary Cisco Controller Name                    : uname-controller
Primary Cisco Controller IP Address              : 203.0.113.1
Secondary Cisco Controller Name                  : uname-controller1
Secondary Cisco Controller IP Address            : 0.0.0.0
Tertiary Cisco Controller Name                   : uname-ewlc2
Tertiary Cisco Controller IP Address             : 0.0.0.0
Administrative State                             : Enabled
Operation State                                  : Registered
AP Mode                                          : FlexConnect
```

```
AP Submode                                          : Not Configured
Office Extend Mode                                  : Enabled
Remote AP Debug                                     : Disabled
Logging Trap Severity Level                         : information
Software Version                                    : 16.8.1.1
Boot Version                                        : 1.1.2.4
Mini IOS Version                                    : 0.0.0.0
Stats Reporting Period                              : 0
LED State                                           : Enabled
PoE Pre-Standard Switch                             : Disabled
PoE Power Injector MAC Address                      : Disabled
Power Type/Mode                                    : PoE/Full Power (normal mode)
```

Proxy address resolution protocol

Explains Proxy-ARP functionality in FlexConnect, providing instructions for enabling Proxy ARP on FlexConnect APs through GUI and CLI workflows.

A proxy address resolution protocol (Proxy ARP) is a method that

- enables learning about MAC addresses through a proxy device
- allows APs to act on behalf of clients by responding to ARP requests, and
- reduces airtime usage by handling ARP requests via controllers rather than clients.

Additional information

The AP functions as an ARP proxy to respond to ARP requests on behalf of clients, minimizing unnecessary air traffic by preventing requests from reaching clients directly when Proxy ARP is enabled. APs that don't own the destination client drop ARP requests unless ARP caching is disabled, in which case APs bridge requests, potentially increasing wireless broadcasts.

Enable proxy ARP for FlexConnect access points (GUI)

Enable Proxy ARP for FlexConnect APs using GUI.

Enable Proxy ARP for FlexConnect APs through the GUI.

Use these steps to enable proxy ARP for FlexConnect APs.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Flex**.
2. Click **Add**.
3. In the **General** tab, enter the **Name** of the Flex Profile and check the **ARP Caching** check box. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.
4. Click **Apply to Device**.

The AP handles ARP requests efficiently, improving network performance through enabled proxy ARP.

Enable proxy ARP for FlexConnect access points (CLI)

Enable Proxy ARP for FlexConnect APs using commands.

Configure proxy ARP for FlexConnect APs using the CLI.

Use these steps to configure proxy ARP for FlexConnect APs.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter wireless flex profile configuration mode.

   Example

   ```
   Device(config)# wireless profile flex flex-test
   ```

3. Enable ARP caching.

   Example

   ```
   Device(config-wireless-flex-profile)# arp-caching
   ```

   Use the **no arp-caching** command to disable ARP caching.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-flex-profile)# end
   ```

5. Display ARP configuration information.

   Example

   ```
   Device# show running-config | section wireless profile flex
   ```

6. Optional: Display detailed information of the flex profile.

   Example

   ```
   Device# show wireless profile flex detailed flex-test
   ```

7. Optional: Display ARP summary.

   Example

   ```
   Device# show arp summary
   ```

Proxy ARP is enabled for FlexConnect APs, allowing for more effective handling of ARP requests in a network setup.

Overlapping client IP address in flex deployment

Describes overlapping client IP scenarios in FlexConnect deployments, offering step-by-step procedures for enabling and verifying overlapping client IP addresses using both GUI and CLI, with verification references.

An overlapping client IP address in Flex deployment is a network configuration strategy that

- uses the same subnet configurations across multiple sites and branches
- includes local DHCP servers configured with identical subnets at each location, and
- identifies and manages multiple client sessions using the same IP address to prevent issues such as IP theft.

In Flex deployments, this strategy allows for cookie-cutter configurations across sites, simplifying network management by ensuring consistent settings.

Controllers in this setup detect duplicate IP usage among clients, categorizing such occurrences as IP theft. These clients are subsequently placed on a blocked list to maintain network security and integrity.

Enable overlapping client IP address in flex deployment (GUI)

Enable Overlapping Client IP Address in Flex Deployment using GUI.

Enable the setting for overlapping client IP addresses to optimize IP management in a Flex deployment.

Use these steps to enable overlapping client IP address in Flex deployment.

Procedure

1. Choose **Configuration > Tags & Profiles > Flex** and click **Add**.
2. On the **Add Flex Profile** window and **General** tab.
3. Check the **IP Overlap** checkbox to enable overlapping client IP address in Flex deployment.
4. Click **Apply to Device**.

Overlapping client IP addresses are now enabled in the Flex deployment. This allows efficient IP address management where different clients might share similar IP addresses.

Enable overlapping client IP address in flex deployment

Enable Overlapping Client IP Address in Flex Deployment using commands.

Enable overlapping client IP address assignment in a Flex deployment to enhance network segmentation and address allocation.

Use these steps to enable overlapping client IP addresses in a Flex deployment:

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a Flex profile and enter Flex profile configuration mode.

   Example

   ```
   Device(config)# wireless profile flex flex1
   ```

3. Enable overlapping client IP address in flex deployment.

   Example

   ```
   Device(config-wireless-flex-profile)# [no] ip overlap
   ```

   By default, the configuration is disabled.

Overlapping client IP addresses in a Flex deployment are now enabled, allowing for flexible IP management and improved network segmentation.

Verify overlapping client IP address in flex deployment (GUI)

Verify overlapping client IP address in flex deployment using GUI.

Verify the client statistics in the IP monitoring interface to ensure accurate data representation and maintain operational oversight.

Use these steps to verify the client statistics in IP monitoring.

Procedure

---

1. Choose **Monitoring** > **Wireless** > **Clients**
2. Click the client in the table to view properties and statistics for each client.
3. On the **Client** window and **General** tab, click **Client Statistics** tab to view the following details:
   - Number of bytes received from client
   - Number of bytes sent to client
   - Number of packets received from client
   - Number of packets sent to client
   - Number of policy errors
   - Radio signal strength indicator
   - Signal-to-noise ratio
   - IP - zone ID mapping
4. Click **OK**.

The verified client statistics are displayed confirming network performance adherence to set requirements.

Verify overlapping client IP address in flex deployment

Verify Overlapping Client IP Address in Flex Deployment using commands.

To verify if the overlapping client IP address in Flex deployment feature is enabled or not, use this command:

```
Device# show wireless profile flex detailed flex1
Fallback Radio shut            : DISABLED
ARP caching                    : ENABLED
Efficient Image Upgrade        : ENABLED
OfficeExtend AP                : DISABLED
Join min latency               : DISABLED
IP overlap status              : DISABLED
```

To view additional details about the overlapping client IP address in Flex deployment feature, use this command:

```
Device# show wireless device-tracking database ip

IP                                          ZONE-ID       STATE      DISCOVERY
   MAC
-------------------------------------------------------------------------------------------

  9.91.59.154                                 0x00000002  Reachable   IPv4
Packet 6038.e0dc.3182
  1000:1:2:3:90d8:dd1a:11ab:23c0              0x00000002  Reachable   IPv6
Packet 58ef.680d.c6c3
  1000:1:2:3:f9b5:3074:d0da:f93b              0x00000002  Reachable   IPv6
Packet 58ef.680d.c6c3
  2001:9:3:59:90d8:dd1a:11ab:23c0            0x00000002  Reachable   IPv6 NDP
    58ef.680d.c6c3
  2001:9:3:59:f9b5:3074:d0da:f93b            0x00000002  Reachable   IPv6 NDP
    58ef.680d.c6c3
  fe80::f9b5:3074:d0da:f93b                  0x80000001  Reachable   IPv6 NDP
    58ef.680d.c6c3
```

To view APs in various site tags, use this command:

```
Device# show ap tag summary
Number of APs: 5

AP Name AP Mac Site Tag Name Policy Tag Name RF Tag Name Misconfigured Tag
Source
-----------------------------------------------------------------------------------------------------
AP3802 70b3.17f6.37aa flex_ip_overlap-site-tag-auto-3
flex_ip_overlap_policy_tag_1 default-rf-tag No Static
AP-9117AX 0cd0.f894.0f8c default-site-tag default-policy-tag default-rf-tag
No Default
AP1852JJ9 38ed.18ca.2b48 flex_ip_overlap-site-tag-auto-2
flex_ip_overlap_policy_tag_2 default-rf-tag No Static
AP1852I 38ed.18cc.61c0 flex_ip_overlap-site-tag-auto-1
flex_ip_overlap_policy_tag_1 default-rf-tag No Static
AP1542JJ9 700f.6a84.1b30 flex_ip_overlap-site-tag-auto-2
flex_ip_overlap_policy_tag_2 default-rf-tag No Static
```

To view APs in FlexConnect mode, use this command:

```
Device# show ap status
AP Name     Status      Mode        Country
-------------------------------------------------------------------------
AP3802      Disabled   FlexConnect   IN
AP1852I     Enabled    FlexConnect   US
AP-9117AX   Enabled    FlexConnect   IN
AP1542JJ9   Disabled   FlexConnect   US
AP1852JJ9   Enabled    FlexConnect   US
```

Troubleshoot overlapping client IP address in flex deployment

To verify the WNCD instance for each of the APs, use this command:

```
Device# show wireless loadbalance ap affinity wncd 0
AP Mac          Discovery Timestamp    Join Timestamp          Tag
-------------------------------------------------------------------------------
0cd0.f894.0f8c   10/27/20 22:11:05    10/27/20 22:11:14    default-site-tag
38ed.18ca.2b48   10/27/20 22:06:09    10/27/20 22:06:19
flex_ip_overlap-site-tag-auto-2
700f.6a84.1b30   10/27/20 22:25:03    10/27/20 22:25:13
flex_ip_overlap-site-tag-auto-2
```

FlexConnect high scale mode

Describes FlexConnect site scaling capabilities, providing procedures for enabling PMK propagation via CLI to support scaling to higher numbers of APs per site tag.

A FlexConnect site capacity is a network feature that:

- scales up to accommodate 300 APs
- supports 3000 802.1x clients, and
- uses PMK caching to bypass EAP exchange during client roaming.
- When a client connects to an AP using an 802.1x authentication setup, the AP under the FlexConnect architecture stores the PMK from the EAP exchange.
- With PMK caching, subsequent connections from the same client bypass the EAP exchange, reducing authentication time.

Supporting information

- The PMK propagation feature is disabled by default.
- Until Cisco IOS XE 17.8.1, the wireless controller pushed the PMK cache to each FlexConnect AP at the same site for faster roaming.
- From Cisco IOS XE 17.8.1 onwards, the PMK cache is shared among FlexConnect APs at the same site for faster roaming.

Enable PMK propagation (CLI)

Enable PMK Propagation using commands.

Enable PMK propagation to facilitate seamless wireless client roaming by sharing PMK information among AP within a single site.

Use these steps to enable PMK propagation:

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Create a FlexConnect profile.

Example

```
Device(config)# wireless profile flex test-flex-profile
```

3. Propagate PMK information to the other APs in the site.

Example

```
Device(config-wireless-flex-profile)# pmk propagate
```

> **Note**
>
> The PMK propagation feature is disabled by default.

PMK propagation is enabled, allowing APs to share PMK information within the site, which enhances wireless client mobility by reducing the latency typically associated with re-authentication.

Example

```
Device# configure terminal
Device(config)# wireless profile flex test-flex-profile
Device(config-wireless-flex-profile)# pmk propagate
```

Flex resilient with Flex and bridge modes

Explains Flex Resilient deployment with Flex and bridge mode APs, detailing configuration steps for Flex and mesh profiles, site tag and mesh associations, switch interface setup, AP attachment, and verification of resilient configurations.

A flex resilient system in flex and bridge modes is a network feature that

- operates within a configuration involving both flex and bridge mode APs
- allows the primary functionality of APs as mesh links between a root access point (RAP) and mesh access point (MAP), and
- ensures that the feature's operations are limited to flex and bridge mode APs.

Additional reference information

- Flex+Bridge Mode: A configuration mode combining the flexibility of standalone (local) management for APs with stable mesh network connectivity through root and mesh APs.
- Mesh Link Operation: Exists between RAP and MAP, allowing communication and bridging.

- Client Connectivity:
  - No new or disconnected clients can associate with a Mesh AP in flex+bridge mode.
  - Ongoing connections are maintained until a loss of the parent link for a child MAP.
- In a locally switching WLAN, the client traffic of a Flex+Bridge MAP is sent to the RAP's switchport.
- A child MAP that loses its parent connection cannot connect to a new parent without reacquiring a connection to the CAPWAP controller.
- The feature does not operate on non-Flex+Bridge mode APs.
- A new or disconnected wireless client cannot associate with a Mesh AP when in this mode.

Configure a flex profile (GUI)

Configure a flex profile using GUI.

Set up a flex profile to enhance network resilience and enable local authentication.

Use these steps to configure a flex profile using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Flex**.
2. Click a **Flex Profile Name**. The **Edit Flex Profile** dialog box appears.
3. Under the **General** tab, choose the **Flex Resilient** check box to enable the Flex Resilient feature.
4. Under the **VLAN** tab, choose the required VLANs.
5. (Optionally) Under the **Local Authentication** tab, choose the desired server group from the **Local Accounting RADIUS Server Group** drop-down list. Also, choose the **RADIUS** check box.
6. Click **Update & Apply to Device**.

The flex profile is successfully configured and updated in the system.

Configure a Flex profile (CLI)

Configure a Flex Profile using commands.

Configure a Flex Profile using commands on a network device to enable specific features and settings.

Use these steps to configure a flex profile using the CLI.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a flex profile and enters flex profile configuration mode using the **wireless profile flex** *flex-profile* command.

Example

```
Device(config)# wireless profile flex new-flex-profile
```

3. Enable ARP caching.

   Example

   ```
   Device(config-wireless-flex-profile)# arp-caching
   ```

4. Enable default parameters for the Flex profile using the **description** *description* command.

   Example

   ```
   Device(config-wireless-flex-profile)# description "new flex profile"
   ```

5. Configure native vlan-id information.

   Example

   ```
   Device(config-wireless-flex-profile)# native-vlan-id 2660
   ```

6. Enable the resilient feature.

   Example

   ```
   Device(config-wireless-flex-profile)# resilient
   ```

7. Configure VLAN name using the **vlan-name** *vlan_name* command.

   Example

   ```
   Device(config-wireless-flex-profile)# vlan-name VLAN2659
   ```

8. Configure the VLAN ID.

   Example

   ```
   Device(config-wireless-flex-profile)# vlan-id 2659
   ```

   The valid VLAN ID ranges from 1 to 4096.

9. Exit the configuration mode and return to the privileged EXEC mode.

   Example

   ```
   Device(config-wireless-flex-profile)# end
   ```

The Flex profile is configured on the network device with the specified settings and features.

Configure a site tag (CLI)

Configure a Site Tag using commands.

Configure a site tag on a network device to manage wireless networks efficiently and map APs to specific site tags.

Use these steps to configure a site tag using CLI.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a site tag and enter site tag configuration mode using the **wireless tag site** *site-name* command.

   Example

   ```
   Device(config)# wireless tag site new-flex-site
   ```

3. Configure a flex profile using the **flex-profile** *flex-profile-name* command.

   Example

   ```
   Device(config-site-tag)# flex-profile new-flex-profile
   ```

4. Remove **Local site** configured from the site tag.

   Example

   ```
   Device(config-site-tag)# no local-site
   ```

5. Map a site tag to an AP using the **site-tag** *site-tag-name* command.

   Example

   ```
   Device(config-site-tag)# site-tag new-flex-site
   ```

6. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-site-tag)# end
   ```

The site tag is configured and mapped to the specified AP, facilitating effective management of wireless networks within the defined site tag.

Configure a mesh profile (CLI)

Configure a mesh profile using commands.

Configure a mesh profile on a network device to enable communication between mesh nodes, ensuring VLAN awareness and connectivity management.

Use these steps to configure a mesh profile using CLI commands.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a mesh profile and enter the mesh profile configuration mode using the **wireless profile mesh** *profile-name* command.

   Example

   ```
   Device(config)# wireless profile mesh Mesh_Profile
   ```

3. Disable VLAN transparency to ensure that the bridge is VLAN aware.

   Example

   ```
   Device(config-wireless-profile-mesh)# no ethernet-vlan-transparent
   ```

4. Exit the configuration mode and return to the privileged EXEC mode.

   Example

   ```
   Device(config-wireless-profile-mesh)# end
   ```

The mesh profile is configured, VLAN transparency is disabled to ensure VLAN awareness, and the device returns to privileged EXEC mode.

Associate the wireless mesh to an AP profile (CLI)

Associate Wireless Mesh to an AP Profile using commands.

Associate a wireless mesh to an AP profile to ensure proper configuration and management of AP.

Use these steps to associate the wireless mesh to an AP profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the AP profile and enter AP profile configuration mode using the **ap profile** *ap-profile-name* command.

Example

```
Device(config)# ap profile new-ap-join-profile
```

3. Configure the mesh profile in AP profile configuration mode using the **mesh-profile** *mesh-profile-name* command.

Example

```
Device(config-ap-profile)# mesh-profile Mesh_Profile
```

4. Configure the Secure Shell (SSH).

Example

```
Device(config-ap-profile)# ssh
```

5. Specify the AP management username and password for managing APs using the **mgmtuser username** *username* **password {0 | 8}** *password* command.

Example

```
Device(config-ap-profile)# mgmtuser username Cisco password 0 Cisco secret 0
 Cisco
```

- **0**: Specifies an UNENCRYPTED password.
- **8**: Specifies an AES encrypted password.

> **Note**
>
> While configuring a username, ensure that special characters are not used as they can lead to configuration errors.

6. Exit configuration mode and return to privileged EXEC mode.

Example

```
Device(config-ap-profile)# end
```

The AP profile is configured with the wireless mesh, allowing for proper management and operation of the AP.

Attach site tag to an access point (CLI)

Attach site tag to an AP using commands.

Attach a site tag for AP identification and configuration management.

Use these steps to attach a site tag to an AP:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure APs and enters ap-tag configuration mode using the **ap** *mac-address* command.

   Example

   ```
   Device(config)# ap F866.F267.7DFB
   ```

3. Map a site tag to the AP using the**site-tag** *site-tag-name* command.

   Example

   ```
   Device(config-ap-tag)# site-tag new-flex-site
   ```

   Note

   Associating Site Tag causes the associated AP to reconnect.

4. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-tag)# end
   ```

The AP is now associated with the specified site tag and will reconnect.

Configure switch interface for APs (CLI)

Configure Switch Interface for APs using commands.

Configure a switch interface for APs using CLI to ensure proper VLAN assignment and trunk settings.

Use these steps to configure switch interface for APs:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the interface to be added to the VLAN.

   Example

   ```
   Device(config)# interface <int-id>
   ```

3. Assign the allowed VLAN ID to the port using the **switchport trunk native vlan** *vlan-id* command.

   Example

   ```
   Device(config-if)# switchport trunk native vlan 2660
   ```

4. Assign the allowed VLAN ID to the port using the **switchport trunk allowed vlan** *vlan-id* command.

   Example

   ```
   Device(config-if)# switchport trunk allowed vlan 2659,2660
   ```

5. Sets the trunking mode to trunk unconditionally.

   Example

   ```
   Device(config-if)# switchport mode trunk
   ```

   > **Note**
   >
   > When the controller works as a host for spanning tree, ensure that you configure portfast trunk, using **spanning-tree portfast trunk** command, in the uplink switch to ensure faster convergence.

6. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-if)# end
   ```

The switch interface is successfully configured for AP connectivity, enabling effective VLAN management and ensuring faster network convergence for connected APs.

Flex resilient with flex and bridge mode AP configuration verification

Provides commands to verify Flex Resilient with Flex and Bridge Mode AP Configuration.

Use the verification commands to view details about AP mode and model, MAP mode details, RAP mode details, and the status of the Flex Profile–Resilient feature.

Use this command to view details about the AP mode and model.

```
Device# show ap name <ap-name> config general | inc AP Mode
AP Mode                                          : Flex+Bridge
AP Model                                         : AIR-CAP3702I-A-K9
```

Use this command to view the MAP mode details.

```
Device# show ap name MAP config general | inc AP Mode
AP Mode                                          : Flex+Bridge
AP Model                                         : AIR-CAP3702I-A-K9
```

Use this command to view the RAP mode details.

```
Device# show ap name RAP config general | inc AP Mode
AP Mode                                          : Flex+Bridge
AP Model                                         : AIR-AP2702I-A-K9
```

Use this command to check the status of the Flex Profile–Resilient feature.

```
Device# show  wireless profile flex detailed FLEX_TAG | inc resilient
Flex resilient                 : ENABLED
```

SuiteB ciphers in FlexConnect mode

Details SuiteB-1X and SuiteB-192-1X security enhancements in FlexConnect, providing configuration procedures for SuiteB ciphers via GUI and CLI, support for advanced encryption algorithms, and verification steps for SuiteB cipher status.

A SuiteB cipher is a cryptographic suite used in FlexConnect mode that:

- enhances enterprise authentication key management by supporting SuiteB-192-1X (AKM 192-1X) and SuiteB-1X (AKM 11)
- operates within the constraints of Galois Counter Mode Protocol (GCMP-128 and GCMP-256) and Counter Cipher Mode with Block Chaining Message Authentication Code Protocol (CCMP-256) for secure key derivation, and
- is incorporated in Cisco WLAN FlexConnect mode from Cisco IOS XE 17.15.1, extending support already available in local mode.

Additional reference information

- WPA2 FlexConnect Mode:
  - SUITEB192-1X ciphers are CCMP-256 and GCMP-256.
  - SUITEB-1X ciphers are GCMP-128.
- WPA3 FlexConnect Mode:
  - SUITEB192-1X cipher is GCMP-256.
  - SUITEB-1X cipher is GCMP-128.

Wi-Fi 7 compliant AP constraints

Wi-Fi 7 WPA3 Security Constraints:

- Open authentication as Wi-Fi 7 is not permitted.

- WPA1 as Wi-Fi 7 is not permitted.
- WPA2 as Wi-Fi 7 is not permitted.
- WPA3 is permitted with restrictions:
  - SAE(24/25) is permitted with GCMP-256.
  - SAE(8/9) is permitted. (It is beaconed as a Wi-Fi 7 client. This is a deviation from the actual security constraint.)
  - WPA2 with PMF is permitted.
  - 802.1x-SHA256 with PMF is permitted.
  - Suite-B-192 with PMF is permitted.

Note

When you enable multi-ciphers (GCMP-128 + GCMP-256) and multi-AKMs (SuiteB + SuiteB-192) on a WLAN, clients that support WPA3 security may not support GCMP-128 encryption. If clients only support GCMP-128 encryption, they will not be able to join the combination of GCMP-128 + GCMP-256 ciphers with SuiteB and SuiteB-192 AKMs.

Configure SuiteB ciphers (GUI)

Follow these steps to configure SuiteB ciphers using the GUI.

Configure SuiteB ciphers for WLAN security using the system GUI to ensure compliance with SuiteB security standards.

Use these steps to configure SuiteB ciphers:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.

   The **Add WLAN** window is displayed.
3. In the **General** tab, enter the **Profile Name**, **SSID**, and the **WLAN ID**.
4. Choose **Security** > **Layer2**, select one of the following options:
   - **WPA + WPA2**
   - **WPA2 + WPA3**
   - **WPA3**

   The **Auth Key Mgmt (AKM)** section will be populated with the possible AKMs supported by the cipher that is selected in the **WPA2/WPA3 Encryption** section. Valid cipher and AKM combinations are displayed in the **Auth Key Mgmt (AKM)** section.
5. In the **WPA2 Encryption** section, select one of the following ciphers:
   - **CCMP256**
   - **GCMP128**
   - **GCMP256**

> **Note**
>
> The **AES(CCMP128)** cipher is selected by default. Multiple ciphers are not currently supported. Clear the **AES(CCMP128)** cipher check box and then select the desired cipher.

Valid cipher and AKM combinations are displayed in the **Auth Key Mgmt (AKM)** section.

6. In the **Fast Transition** section and in the **Status** drop-down list, select **Disabled**.

> **Note**
>
> Disable **Fast Transition** when Suite-B cipher (GCMP256/CCMP256/GCMP128) is configured.

7. In the **Auth Key Mgmt (AKM)** section, check the **SUITEB-1X** check box.
8. Click **Apply to Device**.

The SuiteB ciphers are configured and applied successfully to the selected WLAN profile.

Configure Suite-B ciphers (CLI)

Describes the CLI configuration steps to configure Suite-B ciphers.

Configure Suite-B ciphers for WLAN using CLI to enhance wireless security.

Use these steps to configure Suite-B ciphers:

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN profile and SSID using **wlan** *wlan-profile-name wlan-id ssid-name* command.

   Example

   ```
   Device(config)# wlan suiteb-profile 17 suiteb-ssid01
   ```

   Enters the WLAN configuration mode.

3. Configure the CCMP-128 support using **security wpa wpa2 ciphers** {**aes** | **ccmp256** | **gcmp128** | **gcmp256**} command.

Example

```
Device(config-wlan)# security wpa wpa2 ciphers aes
```

Suite-B ciphers are configured for the specified WLAN, enhancing the security protocols available for wireless connections.

Configure GCMP-128, GCMP-256, or CCMP-256 (CLI)

This task guides you through configuring WPA2 support with advanced ciphers on a WLAN profile to enhance wireless security.

Use these steps to configure WPA2 support with advanced ciphers on a WLAN profile.

Procedure

1. Configure the WPA2 support for a WLAN profile.

   Example

   ```
   Device(config-wlan)# security wpa wpa2
   ```

2. Disable security AKM for 802.1X.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

3. Disable the SuiteB CCMP-128 cipher.

   Example

   ```
   Device(config-wlan)# no security wpa wpa2 ciphers ccmp128
   ```

4. Configure either the CCMP-256 cipher, the GCMP-128 cipher, or the GCMP-256 cipher.

   Example

   ```
   Device(config-wlan)# security wpa wpa2 ciphers gcmp256
   ```

5. Set the authentication list for IEEE 802.1X.

   Example

   ```
   Device(config-wlan)# security dot1x authentication-list suiteb-authlist
   ```

The WLAN profile is configured with enhanced WPA2 support using advanced ciphers, improving the security of wireless communications on your network.

Verify SuiteB cipher status

Describes the verification of the suiteB cipher status with th commands.

Verify SuiteB cipher in a WLAN profile

To verify the SuiteB cipher status in a WLAN profile, use this command:

```
Device# show wlan id 3
saurabh-vwlc#show wlan id 3
WLAN Profile Name     : FIPS
================================================
Identifier                                      : 3
Network Name (SSID)                             : FIPS
Status                                          : Enabled
.
.
.
Security
    802.11 Authentication                       : Open System
    Static WEP Keys                             : Disabled
    802.1X                                      : Disabled
    Wi-Fi Protected Access (WPA/WPA2)           : Enabled
        WPA (SSN IE)                            : Disabled
        WPA2 (RSN IE)                           : Enabled
            AES Cipher                          : Enabled
            CCMP256 Cipher                      : Enabled
            GCMP128 Cipher                      : Disabled
            GCMP256 Cipher                      : Disabled
        Auth Key Management
            802.1x                              : Enabled
            PSK                                 : Disabled
            CCKM                                : Disabled
            FT dot1x                            : Disabled
            FT PSK                              : Disabled
            PMF dot1x                           : Disabled
            PMF PSK                             : Disabled
            SUITEB-1X                           : Disabled
            SUITEB192-1X                        : Enabled
.
.
.
```

Verify SuiteB cipher status using MAC address

To verify the SuiteB cipher status using a MAC address, use this command:

```
Device# show wireless client mac-address H.H.H detail
Client MAC Address : a8XX.ddXX.05XX
Client IPv4 Address : 169.254.175.214
…….
…….
Policy Type : WPA2
Encryption Cipher : CCMP256
Authentication Key Management : SUITEB192-1X
```

40 OEAP Link Test

Topics:

- OfficeExtend Access Point link test

OfficeExtend Access Point link test

Introduces OEAP Link Test fundamentals, and provides instructions for configuring the feature through CLI and GUI, including steps to check link test information and interpret results for effective troubleshooting.

An OfficeExtend Access Point Link (OEAP) Test is a diagnostic feature that

- helps determine the Datagram Transport Layer Security (DTLS) upload speed of the link between an OEAP and a controller
- assists in identifying network bottlenecks and reasons for functionality failures, and
- allows administrators to estimate link quality by running tests on demand.

Feature history for OfficeExtend Access Point link test

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 55: Feature history

Release	Feature	Feature information
Cisco IOS XE 17.5.1	OEAP Link Test	The Cisco OEAP Link Test feature allows you to determine the DTLS upload, link latency, and jitter of the link between an AP and the controller.

Feature scenarios

OEAP users may experience poor performance when connected to a teleworker AP. Running an OEAP link test can diagnose and address these issues.

- The test involves the AP sending synthetic packets to the controller. The controller returns them, and the AP measures link quality.

Use cases

You can use this feature to troubleshoot issues such as low throughput from the Cisco Catalyst 9800 Controller GUI. The OEAP link test provides crucial metrics like DTLS upload speed, link latency, and jitter, facilitating precise issue identification.

Configure OEAP link test (CLI)

Configure OEAP link test using commands.

Perform network diagnostics on an OEAP using CLI to troubleshoot network connectivity.

Procedure

1. Enter privileged EXEC mode.

Example

```
Device> enable
```

2. Trigger network diagnostics on an OfficeExtend AP.

Example

```
Device# ap name ap18 network-diagnostic
```

The OEAP starts the network diagnostics process, allowing you to assess connectivity and performance.

Perform OEAP link test (GUI)

Perform a link test for the OEAP to ensure optimal connection quality and operational efficiency through the GUI.

Procedure

---

1. Choose **Monitoring > Wireless > AP Statistics**.

   In the list of APs, a **Link Test** icon is displayed in the **AP Name** column for OEAP-capable APs.

   > **Note**
   >
   > The **Link Test** icon is displayed only if an AP is OEAP capable and is configured to operate as OEAP.

2. Click **Link Test**.

   The controller runs the link test and displays the results.

The link test results are displayed after selecting the OEAP.

Verify OEAP link test

Verify OEAP Link Test.

To verify network diagnostics information, use this command:

```
Device# show FlexConnect office-extend diagnostics

Summary of OfficeExtend AP Link Latency

CAPWAP Latency Heartbeat

Current: current latency (ms)
Min: minimum latency (ms)
Max: maximum latency (ms)

Link Test
```

```
Upload: DTLS Upload (Mbps)
Latency: DTLS Link Latency (ms)
Jitter: DTLS Link Jitter (ms)

AP Name Last Latency Heartbeat from AP Current Max Min Last Link Test Run
Upload Latency  Jitter
-------------------------------------------------------------------------------------------------
ap-18   1 minute 1 second              0       0   0   12/04/20 09:19:48  8
    2         0
```

41 Cisco OEAP Split Tunneling

Topics:

- Cisco OEAP split tunneling

Cisco OEAP split tunneling

Introduces Cisco OEAP split tunneling, outlining traffic management concepts, prerequisites, restrictions, and use cases, and provides configuration workflows, detailed steps to create and apply IP and URL ACLs, policy profile setup, and verification of split tunnel operation.

Cisco OEAP split tunneling is a feature that

- provides secure communications from a controller to an AP at a remote location
- seamlessly extends the corporate WLAN over the internet to an employee's residence, and
- provides segmentation of home and corporate traffic using the split tunneling feature.

Routing all traffic through traditional VPNs increases volume, slows resource access, and negatively impacts remote user experience. Split tunneling allows for home device connectivity without security risks to corporate policy.

Feature history for Cisco OEAP split tunneling
Table 56: Feature history

Release	Feature	Feature information
Cisco IOS XE 17.8.1	IPv6 Support	IPv6 addressing is supported on the Cisco OEAP Split Tunneling feature.
Cisco IOS XE 17.7.1	Cisco OEAP split tunneling	The split tunneling feature in Cisco OfficeExtend Access Point (OEAP) provides a mechanism to classify client traffic, based on packet content, using access control lists (ACLs).

IPv6 address support for Cisco OEAP split tunneling

From Cisco IOS XE 17.8.1, IPv6 addressing is supported. You can disable IPv6 addressing only by disabling the feature.

The end-to-end network should support IPv6. Both the corporate network (controller, corporate gateway, and other related components) and the home network (wireless clients, home router, and others) should support IPv6.

Traffic to Software as a Service (SaaS) applications such as Cisco WebEx, Microsoft SharePoint, Microsoft Office365, Box, and Dropbox, which are required as part of the work routine, do not need to go through the corporate network when using the split tunneling feature.

Cisco OEAP split tunneling traffic management

Cisco OEAP split tunneling traffic management information.

A Cisco OEAP split tunnel is a network feature that

- classifies client traffic based on packet content using Access Control Lists (ACLs)
- switches matching packets locally from Cisco OEAP, and
- centrally switches other packets over Control and Provisioning of Wireless Access Points (CAPWAP).

Cisco OEAP provides seamless connectivity by broadcasting distinct Service Set Identifiers (SSIDs) for corporate use and personal use allowing for differentiated handling and prioritization of network traffic. Corporate SSID clients obtain their IP addresses from the central DHCP server within the corporate network. With split tunneling enabled, when a client connected to the corporate SSID attempts to access a device within a home network, the OEAP efficiently manages

network traffic by performing Network Address Translation (NAT) or Port Address Translation (PAT) between the client's internal network and the home network.

These examples explains Cisco OEAP split tunneling manages and differentiates network traffic for various use cases:

- VPN Split Tunnel Example: Corporate data can be sent through the secure corporate VPN while allowing personal data to be routed directly to the internet for enhanced performance.
- Home Network Example (SSID): Devices connected to the home SSID receive IP addresses either from the local AP DHCP server or directly from home network equipment when the firewall feature is switched off.

By segmenting traffic, OEAP split tunneling ensures optimized use of WAN bandwidth, improved network performance, and increased security by distinguishing between corporate and personal data streams.

Prerequisites for Cisco OEAP split tunneling

Cisco OEAP split tunneling prerequisites.

Hardware Requirements

Cisco Wave 2 APs or Cisco Catalyst 9100AX Series APs

Configuration Requirements

URL filter list that matches the ACL name configured in split tunneling

Restrictions for Cisco OEAP split tunneling

Cisco OEAP Split Tunneling restrictions.

These requirements outline the restrictions applicable to Cisco OEAP split tunneling:

- Cisco OfficeExtend Access Points (OEAPs) are not supported when Embedded Wireless Controller on Catalyst Access Points (EWC) is used as a controller.
- Mesh topology is not supported.
- Clients connected on a personal SSID or the home network (AP native VLAN) will not be able to discover devices.
- Split tunneling is not supported in standalone mode.
- URL split tunneling supports only up to 512 URLs.
- Specify actions, like deny or permit, only on the URL filter list, not for individual entries.
- If the URL-based ACL contains wildcard URLs, only ten URLs are supported.
- Use up to 128 IP address ACEs (rules) in the IP ACL for split tunneling.
- URL-based split tunnelling only works with IPv4 addresses.

DNS IP addresses restrictions

These requirements limit the amount of DNS IP addresses that can be snooped:

- An AP can snoop 4095 IP addresses per DNS response, if IP addresses are less than 150,000.
- An AP can snoop 10 IP addresses per DNS response, if IP addresses are between 150,000 and 200,000.
- An AP can snoop 5 IP addresses per DNS response, if IP addresses are between 200,000 and 250,000.
- An AP can snoop one IP address per DNS response, if IP addresses are greater than 250,000.

IPv6 Addressing restrictions

These restrictions apply to IPv6 addressing for Cisco OEAP split tunneling:

- Multihoming, which involves multiple router advertisement prefixes, is not supported. If a home network receives multiple prefixes, the AP connected to the controller uses one prefix.
- The system does not support roaming.
- Filtering is not supported on the upstream traffic towards the wireless client.
- Split tunneling is disabled for clients with duplicate IPv6 addresses. Traffic for these clients is forwarded centrally to the controller.
- DHCPv6 prefix delegation is not supported for wireless clients.
- If the corporate prefix length is smaller than the home prefix length, split tunneling for a particular client is disabled.

Use cases for Cisco OEAP split tunneling

Cisco OEAP Split Tunneling Use Cases.

Before Cisco IOS XE 17.7.1, you could use IP ACLs for split tunneling. Cloud services, such as Cisco Webex, could be accessed directly. These services bypassed the corporate network. As a network administrator, you maintained the list of IP addresses that Cisco Webex used.

Starting with Cisco IOS XE 17.7.1, when you use the Cisco OEAP Split Tunneling feature, provide only the DNS names that Cisco Webex uses. The AP then routes traffic from these DNS names directly to the internet instead of through the corporate network.

How Cisco OEAP split tunneling works

Configure Cisco OEAP split tunneling.

Summary

This process involves configuring Cisco OEAP split tunneling by performing multiple steps that include creating ACLs, adding them to profiles, enabling split tunnelling, and verifying the configuration.

Workflow

The process involves these stages:

1. Defining ACLs: Create IP address ACL or URL ACL to specify allowed network paths.
2. Profile association: The administrator adds these ACLs to the FlexConnect Profile to prepare for policy enforcement.
3. Policy activation: Enable split tunneling on the policy profile to segment and direct data flows.
4. Configuration confirmation: The administrator verifies successful configuration to ensure policy compliance, and network functionality.

Result

You have configured Cisco OEAP Split Tunneling, allowing effective management of network traffic and enhanced security for remote devices.

Create an IP address ACL (CLI)

Create an IP Address ACL using commands.

You can configure an IP address-based ACL on network devices to control and secure traffic flow.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Define an extended IPv4 access list using a name.

Example

```
Device(config)# ip access-list extended vlan_oeap
```

> **Note**
>
> An IP ACL can define a default action if no matches exist in the URL ACL.

3. Deny IP traffic from any host.

Example

```
Device(config-ext-nacl)# 10 deny ip any 10.10.0.0 0.0.255.255
```

4. Permit IP traffic from any destination host.

Example

```
Device(config-ext-nacl)# 20 permit ip any any
```

5. Exit configuration mode and return to privileged EXEC mode.

Example

```
Device(config-ext-nacl)# end
```

The IP address ACL effectively filters traffic according to the specified rules on the network device.

Create a URL ACL (CLI)

Create a URL ACL using commands.

Create a URL Access Control List (ACL) on a network device using CLI, enabling control over which URLs can be accessed based on security policies.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the URL filter list.

   Example

   ```
   Device(config)# urlfilter list vlan_oeap
   ```

   Your list name must not exceed 32 alphanumeric characters.

3. Configure the action: Permit (traffic is allowed directly on the home network) or Deny (traffic is directed to the corporate network).

   Example

   ```
   Device(config-urlfilter-params)# action permit
   ```

4. Configure the URL list as post authentication filter.

   Example

   ```
   Device(config-urlfilter-params)# filter-type post-authentication
   ```

5. Configure a URL.

   Example

   ```
   Device(config-urlfilter-params)# url wiki.cisco.com
   ```

6. Optional: Configure a URL.

   Example

   ```
   Device(config-urlfilter-params)# url example.com
   ```

   Use this option when you want to add multiple URLs.

7. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-urlfilter-params)# end
   ```

You have configured the URL ACL successfully, allowing specific URLs to be permitted or denied access through the network device according to the parameters set during configuration.

Add an ACL to a FlexConnect profile (GUI)

Associate an Access Control List (ACL) with a FlexConnect profile, applying filtering and control policies to FlexConnect APs.

Use this procedure to apply an ACL and URL filtering to a FlexConnect profile, enabling traffic control and optional OfficeExtend mode for remote APs.

Before you begin

Use these steps to add an ACL to a FlexConnect profile:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the target FlexConnect profile.

   Example

   ```
   Device(config)# wireless profile flex default-flex-profile
   ```

3. Define the ACL policy to associate with the profile.

   Example

   ```
   Device(config-wireless-flex-profile)# acl-policy vlan_oeap
   ```

4. Configure a URL filter list, if required.

   Example

   ```
   Device(config-wireless-flex-profile-acl)# urlfilter list vlan_oeap
   ```

5. Exit ACL configuration mode to return to the FlexConnect profile.

   Example

   ```
   Device(config-wireless-flex-profile-acl)# exit
   ```

6. Enable OfficeExtend mode for the FlexConnect AP, if applicable.

   Example

   ```
   Device(config-wireless-flex-profile)# office-extend
   ```

7. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-flex-profile)# end
   ```

The ACL and associated URL filter are applied to the selected FlexConnect profile, with OfficeExtend mode enabled if configured. The changes take effect for FlexConnect APs using this profile.

Enable split tunneling in a policy profile

Enable Split Tunnelling in a Policy Profile using commands.

Enable split tunneling in a policy profile to optimize network traffic and enhance performance by allowing specified traffic to bypass the central network and directly access the internet.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a FlexConnect profile.

   Example

   ```
   Device(config)# wireless profile flex default-flex-profile
   ```

3. Disable central association and enable local association for locally switched clients.

   Example

   ```
   Device(config-wireless-flex-profile)# no central association
   ```

4. Configure a split MAC ACL name.

   Example

   ```
   Device(config-wireless-flex-profile)# flex split-mac-acl vlan_oeap
   ```

   Note

   Ensure that you use the same *acl-policy-name* in the FlexConnect profile.

5. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-flex-profile)# end
   ```

Enable split tunneling in the policy profile so that traffic defined in the ACL can locally switch, which improves bandwidth use and network performance.

Verify the Cisco OEAP split tunnel configuration

Verify the Cisco OEAP split tunnel configuration.

To verify the split tunneling DNS ACLs per wireless client on the AP side, use this command:

```
Device# show split-tunnel client 00:11:22:33:44:55 access-list

Split tunnel ACLs for Client: 00:11:22:33:44:55

IP ACL: SplitTunnelACL

Tunnel packets Tunnel bytes NAT packets NAT bytes
             1          242           3       768

URL ACL: SplitTunnelACL

Tunnel packets Tunnel bytes NAT packets NAT bytes
             3          778           0         0

Resolved IPs for Client: 00:11:22:33:44:55 for Split tunnel

HIT-COUNT       URL             ACTION  IP-LIST

1               base1.com       deny.   203.0.113.0
                                        203.0.113.1

2               base2.com       deny.   203.0.113.2

3               base3.com       deny.   203.0.113.5
```

To verify the current binding between a WLAN and an ACL, use this command:

```
Device# show split-tunnel mapping

VAP-Id       ACL Name
    0        SplitTunnelACL
```

To verify the content of the current URL ACL, use this command:

```
Device# show flexconnect url-acl

ACL-NAME          ACTION      URL-LIST

SplitTunnelACL    deny        base.com
```

42 Access Points Survey Mode

Topics:

- AP survey modes

AP survey modes

Outlines AP survey modes, describing their purpose for RF parameter configuration and site surveys, provides login and SSID access details, recommends browser usage, and guides you through enabling survey mode on supported Cisco Catalyst APs.

An AP survey mode is a specialized operational state that

- enables the AP GUI for configuring RF parameters
- facilitates site survey investigation at customer sites, and
- is introduced for Cisco Catalyst 9136 Series APs and other upcoming AP models.

Additional reference information

- Accessing the GUI: Enter 'admin' as the default login and 'admin' as the default password to access AP survey mode from the GUI. Both usernames and passwords are case sensitive.
- SSID broadcast and connection: When the AP is in survey mode, it broadcasts an SSID by default. The default password to connect to this SSID is 'password' (case sensitive).
- Recommended browser: Use Google Chrome to access the AP GUI in survey mode.

Enable survey mode

Enable survey mode on an AP by running the **ap-type site-survey** command from the AP CLI.

Restrictions for access points survey mode

- When survey mode is active, certain GUI features such as WAN, Firewall, and Network Diagnostics are hidden.
- To restore visibility of hidden features on the AP GUI, switch the AP to CAPWAP mode by running the **ap-type capwap** command from the AP CLI. In CAPWAP mode, the AP GUI becomes accessible when the **OfficeExtend AP** field is enabled in the FlexConnect profile page linked to that AP.

43 AP Deployment Mode

Topics:

- AP deployment mode

AP deployment mode

Introduces AP deployment mode concepts, detailing relevant use cases, provides procedures for configuring deployment modes via both GUI and CLI, guides users in viewing deployment status, and explains verification steps to ensure proper AP operation.

An AP deployment mode is a configuration feature that

- enables Cisco Catalyst 9124AX Series Outdoor AP to operate in indoor mode within the -E regulatory domain
- expands available channels to include U-NII-1 and U-NII-2 in addition to U-NII-2C channels, and
- applies specifically to the -E regulatory domain for increased channel availability.

Supporting reference information

- For more information on the regulatory domain, see Countries and Regulations.

  For more information on U-NII-1 and U-NII-2, see https://en.wikipedia.org/wiki/Unlicensed_National_Information_Infrastructure.

Restrictions for AP deployment mode

AP deployment mode use case.

- This feature applies to Cisco Catalyst 9124AX Series Outdoor APs only.
- We recommend operating Cisco Catalyst 9124AX Series Outdoor APs in Indoor mode in environments such as greenhouses or walk-in freezers to ensure optimal network performance and equipment longevity.

Configure an AP deployment mode (GUI)

Configure AP deployment mode using the GUI.

Configure the deployment mode of an AP to fit the intended environment using GUI.

Use these steps to configure the AP deployment mode:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.

   To add a new AP join profile, see *Configuring an AP Profile (GUI)*. To modify an existing AP join profile, select the required AP join profile.

2. Click **General** tab.
3. From the **Deployment mode** drop-down list, select one of these options:
   - *Default or Outdoor*: Select this option if you want to configure the AP in outdoor mode. By default, Cisco Catalyst 9124AX Series APs are configured in outdoor mode.
   - *Indoor*: Select this option if you want to configure the AP in indoor mode for enclosed spaces like greenhouses or walk-in freezers.

> **Note**
>
> When the deployment mode is changed, the system prompts you to confirm the change. Select **Yes** to accept the change.

4. Click **Apply to Device**.

The AP deployment mode is successfully configured to match its environment, ensuring optimal performance.

View deployment status

View deployment status using GUI.

View the deployment status after configuring the AP deployment mode.

Before you begin

Use these steps to view deployment status:

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. On the **All Access Points** tab, click on a Cisco Catalyst 9124AX Series Access Point access point.
3. In the **Edit AP** window, select the **Advanced** tab to view the default and current mode of the AP.

The deployment status is displayed.

Configure AP deployment mode (CLI)

Configure AP deployment mode using commands.

Configure the deployment mode for an AP using CLI commands to ensure it operates effectively in the desired mode.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile1
   ```

3. Configure the outdoor AP to operate in indoor mode.

Example

```
Device(config-ap-profile)# dual-mode-ap-deployment-mode indoor
```

4. Exit configuration mode and return to privileged EXEC mode.

Example

```
Device(config-ap-profile)# end
```

The AP is successfully set to the specified indoor mode, ensuring that it operates under the correct deployment configuration.

Verify AP deployment mode

Verify AP deployment mode.

To verify whether the AP indoor mode is enabled or not, use this command:

```
Device# show ap name APXXXX.31XX.83XX config general
Cisco AP Name   : APXXXX.31XX.83XX
================================================

Cisco AP Identifier                              : 4ca6.4d22.f140
Country Code                                     : Multiple Countries : CZ,US
Regulatory Domain Allowed by Country             : 802.11bg:-AE   802.11a:-ABE
   802.11 6GHz:-BE
Radio Authority IDs                              : None
AP Country Code                                  : CZ  - Czech Republic
AP Regulatory Domain
  802.11bg                                       : -E
  802.11a                                        : -E
.
.
.AP Indoor Mode                                  : Enabled
```

To verify the available channel list in AP console, use this command:

```
AP# show rrm receive configuration
RRM configuration slot 1
==================================
Group Id
Switch Id            :0904640500ff
Group Cnt            :57454
IP address           :9.4.100.5
Encrypted            :0
Version              :1
Key                  :ff3fff55ffffff42ffff2cff6d0affff
Domain               :default
Key Name             :Channel Count        :19
TX Chans             :36 40 44 48 52 56 60 64 100 104 108 112 116 120 124 128
 132 136 140
```

To view the indoor deployment details in AP console, use this command:

```
AP# show capwap client configuration
AdminState                         : ADMIN_ENABLED(1)
```

```
Name                                 : AP3C57.31C5.9478
Location                             : default location
Primary controller name              : Rack10_katar
Primary controller IP                : 9.4.100.5
Secondary controller name            :
Tertiary controller name             :
.
.
.Indoor Deployment                  : 2!Indoor Deployment: 2 signifies that the
 AP is in Indoor mode.
!Indoor Deployment: 0 signifies that the AP is in Outdoor mode.
```

Part V

Smart Spaces

Topics:

- Access Point Location
- SS IOT: Implement Wi-Fi Coexistence with ESL
- Smart Spaces - AnyLocate capabilities

44 Access Point Location

Topics:

- AP auto location support

AP auto location support

Introduces AP Auto Location Support, covering advanced features in Cisco IOS XE 17.12.1, ultra wide band ranging, GNSS raw data streaming via gRPC, configuration of geolocation and ranging (GUI and CLI), fine time measurement, air pressure reporting, and verification of AP location settings.

Access points auto location support is a wireless network feature that

- automates the identification and assignment of geographic locations to APs
- uses technologies such as the Global Positioning System (GPS) or a Global Navigation Satellite System (GNSS), and
- enhances the accuracy and efficiency of wireless network management.

Feature history for AP auto location support
Table 57: Feature history

Release	Feature information
Cisco IOS XE 17.12.1	The Access Point Auto Location Support feature helps to effectively self-locate APs in a global coordinate by combining various ranging technologies and algorithms.
Cisco IOS XE 17.13.1	The improved Access Point Auto Location Support feature helps wireless clients to leverage Fine Timing Measurement (FTM) and AP GNSS for indoor navigation.
Cisco IOS XE 17.15.2	Ultra Wide Band (UWB) technology uses various methods to determine the position and distance between APs. UWB is supported in the Cisco Wireless 9178I Series Wi-Fi 7 APs, Cisco Wireless 9176I Series Wi-Fi 7 APs, and Cisco Wireless 9176D1 Series Wi-Fi 7 APs to perform AP to AP ranging.

The AP auto location solution mitigates common challenges in enterprise environments where manual AP location entry is often inaccurate or neglected. This system

- automates AP location detection using GNSS and Fine Timing Measurement (FTM) technologies, when accessible
- uses manual anchors to support location determination when GNSS signals are unavailable, and
- requires adequate AP density to ensure inter-AP communication at maximum power for precise geolocation.

Process Overview

Auto location support enhances efficient, precise AP management in settings where traditional GNSS solutions are impractical.

- Segmentation and Labeling: Dividing a large number of APs into smaller, floor-specific segments to determine geolocations.
- Ranging and Geolocation: Leveraging strong GNSS signals to assist APs with weak signals, through inter-AP ranging data and FTM technology.
- Fallback with Manual Anchors: Implementing manual anchors as needed to maintain location accuracy in areas without GNSS reception.

Use Case

Supported Access Points

Auto location support is effectively applied for self-locating APs within a building or a specific floor, aiding seamless network integration and management.

For information about APs that support this feature see https://www.cisco.com/c/en/us/td/docs/wireless/access_point/feature-matrix/ap-feature-matrix.html.

Advanced functionalities in Cisco IOS XE 17.12.1

Advanced functionalities in Cisco IOS XE 17.12.1

- Client FTM: It enhances indoor navigation by allowing AP range checks from unassociated clients through designated WLANs on 5-GHz and 6-GHz bands.
- Air Pressure Reporting and Updates: APs send air pressure data to the controller, with RPC and Privileged EXEC mode commands to enable collection. Samples are collected at regular intervals, such as every 30 seconds for 10 minutes.
- Band Filters: Optimizes AP-to-AP ranging by enhancing site-tag efficiency on specified bands like 5 GHz or 6 GHz.
- Ranging Completion Notification: The controller oversees APs for ranging completion, relaying status to Cisco Spaces upon completion.
- AP Movement Alarm: When an AP is disconnected and then reconnected at a new location, the system reports an alarm to the controller, which updates the system with syslog entries and alerts Cisco Spaces.

Ultra-wideband ranging

Ultra-wideband ranging infromation.

Ultra-wide band ranging is a technology-based method that

- determines position and distance between APs using a range of techniques
- employs double-sided two-way ranging between UWB-capable APs and UWB-enabled devices to enhance accuracy, and
- differentiates multipath components through short pulse duration and wide bandwidth.

UWB for increased accuracy

UWB technology, combined with precise time measurement, enhances accuracy in environments with heavy multipath conditions, including concrete structures or metallic surfaces. This technology optimizes performance and reliability by reducing signal reflection and interference, thereby enhancing the precision of AP AnyLocate in specific Cisco Wireless AP series.

In particular, UWB radios in Cisco Wireless 9178I Series Wi-Fi 7 APs, 9176I Series Wi-Fi 7 APs, and 9176D1 Series Wi-Fi 7 APs are utilized to conduct AP-to-AP ranging. When combined with Wi-Fi-based ranging, UWB yields superior results in specific environments.

Group ranging

Group ranging is a technique where ranging is exclusively conducted within a group of identified APs.

- These APs can span different site tags.
- A list of AP radio MAC addresses must be provided as input to execute group ranging.

Group ranging does not have CLI support and relies on Network Configuration Protocol (NETCONF) and Remote Procedure Calls (RPC) for implementation.

Global Navigation Satellite System raw data streaming through gRPC

GNSS raw Ddata streaming through gRPC info.

Global Navigation Satellite System (GNSS) data streaming is a feature that

- streams high-volume raw GNSS data from APs directly to Cisco Spaces
- uses the Google Remote Procedure Call (gRPC) protocol for efficient data handling, and
- reduces the load on network controllers.

Supported equipment and features

- Internal GNSS antennas
- External GNSS data through Gypsum USB dongle
- Cisco Catalyst 9120 Series AP, Cisco Catalyst 9130 Series AP, Cisco Catalyst 9136 Series AP, Cisco Catalyst 9162 Series AP, Cisco Catalyst 9164 Series AP, Cisco Catalyst 9166 Series AP, Cisco Catalyst 9166D Series AP, Cisco Wireless 9178 Series AP, and Cisco Wireless 9176 Series AP.

Configuration Information

- There is no configuration on the controller.
- The AP sends GNSS data to Cisco Spaces through gRPC once the connection is established.

GNSS raw data streaming through gRPC

Summary

This process describes how GNSS data is transmitted from APs to Cisco Spaces using gRPC technology.

Workflow

1. The APs receive GNSS data either internally or through the Gypsum module.
2. After the controller is added to the Cisco Spaces connector, the AP establishes a gRPC tunnel to the Cisco Spaces connector.
3. Cisco Spaces pushes the configuration to set the GNSS reporting interval on the AP.
4. The AP sends GNSS data to Cisco Spaces at the configured interval.

Result

The process successfully transfers GNSS data from the APs to Cisco Spaces, ensuring seamless communication.

Restrictions for GNSS raw data streaming through gRPC

- AP Join Profile Preparation: You must create the AP join profile before adding the controller to the Cisco Spaces connector.
- Interval Reset Upon Reload: If the AP reloads, note that the interval duration resets. The AP must then wait for 24 hours for the connector to push the configuration again.
- Data Sending Restriction: Ensure that GNSS raw data streaming cannot be explicitly disabled.

AP commands

Outline the available AP commands executable via the controller for effective network management.

These are the AP commands. These commands can be run from the controller using the **remote** command.

- **show spaces endpoint key access**
- **show spaces endpoint key authentication**
- **show spaces endpoint connection detail**
- **debug spaces <info/debug/error/warning/critical>**

Configure AP geolocation derivation using ranging (GUI)

Configure access point geolocation derivation using ranging GUI.

This task explains how to enable geolocation derivation for access points using ranging to determine their location.

Use these steps to configure access point geolocation derivation using ranging:

Procedure

1. Choose **Configuration** > **Wireless** > **Wireless Global**.
2. In the **AP Geolocation** section, click the **Geolocation Derivation Using Ranging** toggle button to enable geolocation derivation.

   Enabling **Geolocation Derivation Using Ranging** allows the AP to participate in location services using ranging for geolocation.
3. Click **Apply**.

> **Note**
>
> Once enabled, **Geolocation Derivation Using Ranging** updates the AP ranging capability on Cisco Spaces within approximately 30 minutes.

The AP is configured for geolocation derivation and is ready for use in location services employing ranging techniques.

Configure AP geolocation derivation using ranging (CLI)

Configure access point geolocation derivation using ranging using commands.

This task allows you to configure geolocation derivation for an access point using ranging, which can improve location accuracy.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enable geolocation derivation using ranging.

Example

```
Device(config)# ap geolocation derivation ranging
```

> **Note**
>
> Use the **no** command to disable the feature.

Geolocation derivation is enabled for AP, which potentially improves its location estimation capabilities. If the configuration fails, verify the commands and check the device logs for errors.

Configure the AP Ranging Parameters through the GUI

Configure AP Ranging Parameters using the graphical user interface (GUI).

This task guides you through configuring the ranging parameters on an access point using the controller's graphical user interface.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. Click **Add**.
3. In the **General** tab, enter the name and description of the corresponding AP join profile.
4. Click the **Geolocation** tab.
5. In the **Fine Time Management (FTM)** section, complete these steps:
   - Click the **FTM** toggle button to allow APs to use FTM for inter AP ranging.
   - In the **FTM Initiator Burst Size** field, specify the burst size value. The burst size determines the size of the transmitted frames. The valid range is between two and 31 frames. The default value is 16 frames per burst.
   - From the **FTM Initiator Burst Duration** drop-down list, choose a value. The burst duration determines the interval of the transmitted frames. The default value is 32 microseconds.
6. Click **Apply to Device**.

Upon completion, the access point will be configured with the specified ranging parameters, enhancing wireless communication by allowing accurate distance measurements.

Configure AP ranging parameters (CLI)

Set up AP geolocation ranging parameters to enable Fine Timing Measurement (FTM) for precise location tracking.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile1
   ```

3. Enable geolocation Fine Timing Measurement (FTM).

   Example

   ```
   Device(config-ap-profile)# geolocation ftm
   ```

4. Configure the geolocation FTM burst size.

   Example

   ```
   Device(config-ap-profile)# geolocation ftm initiator burst-size 8
   ```

   The burst size values are 4, 8, 16, 32 and 64 frames. The default value is 8 frames per burst.

5. Configure the geolocation FTM burst duration.

   Example

   ```
   Device(config-ap-profile)# geolocation ftm initiator burst-duration 32ms
   ```

   The default value is 32 microseconds.

The AP is configured with the specified geolocation ranging parameters for FTM.

Configure AP location (CLI)

Configure AP Coordinates and Floor Information using commands.

Accurately configure AP with their physical location and floor information for precise network management and location-based services.

Use these steps to configure the AP coordinates and the corresponding floor information.

> **Note**
>
> There are no corresponding GUI steps for this configuration.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

   Enter your password, if prompted.

2. Configure the longitude and latitude of the AP.

   Example

   ```
   Device# ap name cisco-ap1 geolocation coordinates 90 45
   ```

   The value range for longitude is from -180 to 180 degrees. The value range for latitude is from -90 to 90 degrees.

3. Configure the floor ID for the AP.

   Example

   ```
   Device# ap name cisco-ap1 floor 2147483647
   ```

   The floor ID range is from -2147483648 to 2147483647.

The AP has updated geographical coordinates and floor information, allowing accurate mapping and integration with location-based services.

Configure on-demand AP ranging (CLI)

Configure on-demand access point ranging using commands.

Improve the geolocation accuracy of APs through on-demand ranging configuration.

Use these steps to configure on-demand AP ranging:"

> **Note**
>
> There are no corresponding GUI steps for this configuration.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

   Enter your password, if prompted.

2. Enable accurate ranging using the 5 GHz or 6 GHz bands on APs under the configured site tag.

   Example

   ```
   Device# ap geolocation ranging site <cite>ap-site-tag</cite> accurate 5ghz
   ```

   > **Note**
   >
   > Client connections may be disrupted.

3. Enable accurate ranging using the 5 GHz or 6 GHz bands on all APs.

   Example

   ```
   Device# ap geolocation ranging all accurate 5ghz
   ```

   > **Caution**
   >
   > Client connections may be disrupted.

APs are configured for enhanced geolocation accuracy through on-demand ranging.

Enable fine time measurement (802.11mc) responder (GUI)

Enable fine time measurement (802.11mc) responder using GUI.

Enable the AP to respond to fine time measurement queries for indoor location determination.

Follow these steps to configure the AP to respond to fine time measurement queries:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.

   The **Add WLAN** window is displayed.
3. In the **General** tab, enter the **Profile Name**, the **SSID**, and the **WLAN ID**.
4. In the **Advanced** tab, under the **Geolocation** section, click the **Fine Time Measurement (FTM) Responder** toggle button to enable the AP to respond to time measurement queries sent from a client.

   The client sends time measurement queries to measure its distance to the AP and determine their indoor location.

   The **Advertise AP Location** field is displayed.
5. Click the **Advertise AP Location** toggle button to enable the AP to advertise its geolocation coordinates.
6. Click **Apply to Device** to save the configuration.

The AP is configured to respond to fine time measurement queries.

Configure fine time measurement (802.11mc) responder

Configure fine time measurement (802.11mc) responder using commands.

Configure a fine time measurement (802.11mc) responder to ensure it communicates accurate location information.

Follow these steps to configure a fine time measurement responder:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Specify the WLAN name and ID.

   Example

   ```
   Device(config)# wlan wlan-profile 36 ssid1
   ```

   - *profile-name* is the WLAN name. Valid WLAN names can contain up to 32 alphanumeric characters
   - *wlan-id* is the wireless LAN identifier. The valid range is from 1 to 4096.
   - *ssid-name* is the SSID which can contain 32 alphanumeric characters.
3. Configure the FTM responder.

   Example

   ```
   Device(config-wlan)# geolocation ftm-responder
   ```
4. Configure the advertisement of AP location to clients.

Example

```
Device(config-wlan)# geolocation ftm-responder advertise-ap-location
```

5. Optional: Clear the AP geolocation best known FTM report.

Example

```
Device# clear ap geolocation ranging
```

The fine time measurement responder is configured successfully, allowing communication of accurate location information to compatible devices.

Configure air pressure reporting (CLI)

Configure air pressure reporting using commands.

This task allows the configuration of air pressure data reporting to optimize environmental monitoring.

Procedure

---

1. Enable privileged EXEC mode.

Example

```
Device> enable
```

Enter your password if you are prompted.

2. Configure sensor air pressure data collection either for all APs or specifically for APs within a site tag.

Example

```
Device# ap sensor air-pressure site sitetag1 duration 10
```

The value range of the duration of the air pressure reporting is from 1 minute to 1,440 minutes. The default is 10 minutes.

From Cisco IOS XE 17.15.1, the default air pressure sample interval is changed from 30 seconds to 60 seconds. For example, if the duration is set to 10 minutes, the APs send 10 samples, spaced at 60-second intervals each.

Air pressure data collection is configured for the specified access points, allowing for effective environmental monitoring and reporting.

Configure AP ultra wide band parameters (GUI)

Configure access point ultra wide band parameters using GUI.

Configure the ultra wide band parameters on an access point using the GUI.

Follow these steps to configure the UWB parameters:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. Click **Add**.
3. In the **General** tab, enter the name and description of the corresponding AP join profile.
4. Click the **Geolocation** tab.
5. In the **Ultra Wide Band (UWB)** section, complete these steps:
   - Click the **UWB** toggle button to allow APs to use UWB for inter AP ranging.
   - In the **UWB Burst Size** field, specify the burst size value. The size of the transmitted frames is determined by the burst size. The range is from 10 ro 100 frames. The default value is 32 frames per burst.
   - From the **UWB Burst Duration** drop-down list, choose a value. The interval of the transmitted frames is determined by the burst duration. The valid range is from 5 to 30 seconds. The default value is 10 seconds.
   - Click **Apply to Device**.

The AP is configured with the specified UWB parameters, allowing for optimized inter AP ranging and communication.

Configure access point ultra wide band parameters (CLI)

Configure access point ultra wide band ranging parameters using commands.

This task guides you through configuring ultra wide band (UWB) parameters for an AP, optimizing geolocation accuracy and efficiency.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile default-ap-profile
   ```

3. Enable default.

   Example

   ```
   Device(config-ap-profile)# geolocation uwb
   ```

4. Configure the geolocation UWB burst size.

   Example

   ```
   Device(config-ap-profile)# geolocation uwb initiator burst-size 32
   ```

   The range of burst size values is from 10 to 100 frames. The default value is 32 frames per burst.

Important

We recommend that you do not alter the burst size and the burst duration.

5. Configure the geolocation UWB burst duration.

Example

```
Device(config-ap-profile)# geolocation uwb initiator burst-duration 10s
```

The range of burst duration values is from 5 to 30 seconds. The default value is 10 seconds.

The AP has optimized UWB parameters, which enhance its geolocation functionality for improved accuracy.

Verify AP geolocation information

Verify AP geolocation information.

Verify AP geolocation ranging report

To verify the AP geolocation ranging report, run this command:

```
Device# show ap geolocation ranging report
RequestID     AP Name                          NeighborAP(Name/MAC)
     Type    Method  Dist(cm)  Channel   Band      Width(MHz)  AOA(elev)
AOA(azim)     RSSIAvg  Frames       Time
-------------------------------------------------------------------------------
1             AP6849-92D3-87E0                  AP6849-92D4-1470
     BEST    FTM     298       36        5 GHz     80          NA
NA            -27          6/1      04/30/2024 09:44:03 UTC
1             AP6849-92D3-87E0                  AP6849-92D4-1470
     LATEST  FTM     251       36        5 GHz     80          NA
NA            -28         12/3      05/09/2024 12:03:33 UTC
1             AP6849-92D4-1470                  AP6849-92D3-87E0
     BEST    FTM     229       36        5 GHz     80          NA
NA            -32          5/2      05/06/2024 14:32:28 UTC
1             AP6849-92D4-1470                  AP6849-92D3-87E0
     LATEST  FTM     227       36        5 GHz     40          NA
NA            -37         15/0      05/09/2024 12:03:31 UTC
1             AP6849-92D4-1470                  AP6849-92D3-87E0
     LATEST  FTM     237       36        5 GHz     80          NA
NA            -36          9/1      05/09/2024 12:03:36 UTC
```

> **Note**
>
> AoA which refers to the angle of arrival (azimuth and elevation) is included in the show command output from the Cisco IOS XE 17.15.2 release.

To view an AP geolocation ranging request, run this command:

```
Device# show ap geolocation ranging request
Request ID    SiteTag/Group/All APs             Mode          Method
Band      Requests  Responses  Reports    Start Time                End Time

------------------------------------------------------------------------------------------------
1             ALL APs                           Accurate      ALL
All       4         0          0          05/15/2024 14:33:33 UTC
```

> **Note**
>
> The method column is introduced from Cisco IOS XE 17.15.2 release.

To view the AP geolocation summary, run this command:

```
Device# show ap geolocation summary
AP Name                Radio MAC          Location  Location Longitude
Latitude    Major-axis Minor-axis Orientation Height Height   Height
Height
                                          Type      Source   (degrees)
(degrees)   (meters)   (meters)   (degrees)   Type   (meters) Uncertainty
Source

                                                             (meters)

------------------------------------------------------------------------------------------------
APCC9C.3EF1.0F30      10f9.20fd.f640     Ellipse   Manual   90.000000
90.000000    0          0          0.000000    NA     NA       NA
NA
```

To view the AP geolocation statistics, run this command:

```
Device# show ap geolocation statistics
Num APs with GNSS                  : 1
Num APs with manual height         : 0
Num APs with derived geolocation   : 0
Last geolocation derivation run    : 07/21/2023 08:54:21
```

To view the AP geolocation GNSS-capable summary, run this command:

```
Device# show ap geolocation gnss-capable summary
------------------------------------------------------------------------------------------------
AP Name                      Radio MAC          GPS Coverage    Antenna Type
 Last GPS fix
------------------------------------------------------------------------------------------------
```

```
APCC9C.3EF4.CF00             10f9.20fd.b6e0     No              Internal
 NA
```

Note

APs do not store the GNSS location. The location is recalculated after a reboot unless it is configured manually.

To view the AP geolocation ranging status, run this commands:

```
Device# show ap geolocation ranging status
```

```
Device# show ap name geolocation ranging status
```

To view the ranging capability of APs, run this command:

```
Device# show ap geolocation ranging capability
AP Name               FTM Responder   FTM Initiator
-----------------------------------------------------------------
AP0001.Cisco.CF00      Yes             Yes
AP0002.Cisco.0F30      Yes             Yes
AP-2800                No              No
AP0003.Cisco.82a0      No              No
```

45 SS IOT: Implement Wi-Fi Coexistence with ESL

Topics:

- Wi-Fi-IoT radio coexistence

Wi-Fi-IoT radio coexistence

Introduces Wi-Fi-IoT radio coexistence mechanisms, detailing collaborative and non-collaborative interference mitigation approaches, explaining PTA coordination for airtime sharing, and outlining performance improvements in environments with dense IoT devices.

Wi-Fi-IoT radio coexistence is a collaborative interference-reduction mechanism that

- minimizes signal interference between Wi-Fi and IoT radios operating on the 2.4 GHz ISM band
- utilizes Packet Traffic Arbitration (PTA) to coordinate airtime sharing between collocated radios, and
- improves the performance and reliability of wireless networks in environments with high IoT device density.

Packet Traffic Arbitration (PTA)

Packet Traffic Arbitration (PTA) is a coordination entity that provides per-packet authorization for transmissions. It dynamically manages the wireless medium based on traffic load, ensuring that high priority packets from either radio are granted access while preventing simultaneous transmissions that would cause collisions.

Collaborative and Non-collaborative Mechanisms

- Non-collaborative: Radios do not exchange information.
- Collaborative: Radios exchange real-time state information to negotiate medium access.

Difference between collaborative and non-collaborative mode

In non-collaborative mode, the IoT radio and the Wi-Fi radio each transmit whenever they have data, regardless of whether the other radio is currently in use. As a result, if a collision occurs when the Wi-Fi radio is busy because it is being used by the Wi-Fi chipset, the IoT radio cannot send its packet. In collaborative mode, the IoT radio asks the Wi-Fi radio for permission to transmit. Once permission is granted, the IoT radio transmits over the radio.

Feature history for Wi-Fi-IoT radio coexistence

This section explains the feature history details of the Wi-Fi-IoT radio coexistence feature.

- This table provides release and related information for the feature explained in this module.
- This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 58: Feature history for Wi-Fi-IoT radio coexistence

Feature Name	Release Information	Feature Description
Wi-Fi-IoT radio coexistence	Cisco IOS XE 26.1.1	Wi-Fi-IoT radio coexistence is a collaborative interference-reduction mechanism that minimizes signal interference between Wi-Fi and IoT radios operating on the 2.4 GHz ISM band. It utilizes Packet Traffic Arbitration (PTA) to coordinate airtime sharing between collocated radios, improving the performance and reliability of wireless networks in environments with high IoT device density.

Best practices for Wi-Fi-IoT coexistence

This section explains the best practices for Wi-Fi-IoT coexistence.

- IOx application dependency: For Electronic Shelf Labeling (ESL) use cases, the coexistence feature requires an active IOx application. The IOx application must be installed and running on the AP to manage the IoT radio firmware.
- Use supported 6E platforms: During initial deployment, use the Cisco Catalyst 9162, 9164, or 9166 series APs.
- Controller priority: The controller's configuration commands take precedence over requests from local IOx applications.

How Wi-Fi-IoT coexistence works with IOx applications

This section explains how Wi-Fi-IoT coexistence works with IOx applications.

Wi-Fi-IoT coexistence enables the controller, the AP, and collocated third-party applications to manage radio airtime by facilitating a synchronized communication flow. The key components involved in the process are:

- The controller defines the global administrative state (enable or disable) using RF profiles.
- The AP manages communication between the Wi-Fi chip, the IoT chip, and the IOx application.
- The IOx application, such as those for ESL, requests IoT radio access and provides specific IoT performance counters.

Summary

The process involves these stages:

- Administrative Configuration: The controller sends the desired coexistence state to the AP.
- Application Subscription: After installation, the IOx application establishes gRPC connection to the AP and subscribes to the coexistence service.
- Enablement Request: The IOx application asks the AP to enable coexistence on the IoT chip.
- Priority Verification: The AP checks the administrative state from the controller and, if enabled, activates coexistence on the Wi-Fi and IoT chips.
- Status Synchronization: The IOx application confirms coexistence is enabled on the IoT chip, and the AP reports the 'Enabled' state to the controller.
- Periodic Reporting: The AP collects and forwards IoT statistics from the IOx application to the controller.

Workflow

When the administrative and application states are aligned, the system reliably transmits both Wi-Fi and IoT data without requiring manual intervention.

Configuration priority for Wi-Fi-IoT coexistence

This section explains the configuration priority details for Wi-Fi-IoT coexistence.

Controller configuration has higher priority than IOx application requests.

- If you disable coexistence on the controller, the AP disables the feature on the Wi-Fi chip and instructs the IOx application to disable the feature on the IoT chip, regardless of the IOx application's requirements.
- The IOx application can only enable coexistence if the administrative state in the AP's RF profile is set to Enabled.

Create an RF profile for Wi-Fi-IoT coexistence (CLI)

Use this procedure to create an RF profile for Wi-Fi-IoT coexistence using commands.

Provide a consistent RF environment that supports both Wi-Fi and IoT devices by defining an appropriate RF profile using commands.

RF profiles allow you to manage and tune radio parameters for different deployment needs, such as ensuring reliable operation of Wi-Fi and IoT devices in the 2.4 GHz band.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure Cisco AP and 802.11 parameters.

   Example

   ```
   Device(config)# ap dot11
   ```

3. Configure 802.11b parameters.

   Example

   ```
   Device(config)# ap dot11 24ghz
   ```

4. Set the RF-profile name and specify the name of the RF profile to configure.

   Example

   ```
   Device(config)# ap dot11 24ghz rf-profile default-rf-profile
   ```

5. Specify and enter the description of the RF profile.

Example

```
Device(config-rf-profile)# description description-detail
```

6. Negate a command. Shut down the profile and disable network.

Example

```
Device(config-rf-profile)# no shutdown
```

7. Exit the sub-mode.

Example

```
Device(config-rf-profile)# exit
```

Configure Wi-Fi-IoT coexistence (CLI)

Use this procedure to configure Wi-Fi-IoT coexistence using commands.

Enable coexistence between Wi-Fi and IoT radios to ensure reliable operation and minimize interference on supported Cisco APs using commands.

Wi-Fi-IoT coexistence allows the AP to support both Wi-Fi and IoT devices simultaneously on the 2.4 GHz band.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure Cisco AP and 802.11 parameters.

Example

```
Device(config)# ap dot11
```

3. Configure 802.11b parameters.

Example

```
Device(config)# ap dot11 24ghz
```

4. Set the RF-profile name and specify the name of the RF profile to configure.

Example

```
Device(config)# ap dot11 24ghz rf-profile default-rf-profile
```

5. Configure the RF Profile IoT settings.

   Example

   ```
   Device(config-rf-profile)# iot
   ```

6. Configure Wi-Fi-IoT radio coexistence.

   Example

   ```
   Device(config-rf-profile)# iot coexistence
   ```

7. Exit the sub-mode.

   Example

   ```
   Device(config-rf-profile)# exit
   ```

Tag the RF profile to an AP (CLI)

Use this procedure to tag the RF profile to an AP using commands.

Assign a specific RF profile to customize the radio configuration of an AP using commands.

Perform this task when you need to apply a distinct RF profile to a particular AP for more granular wireless control.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure wireless and tag parameters.

   Example

   ```
   Device(config)# wireless tag
   ```

3. Configure and enter the name of the RF tag.

   Example

   ```
   Device(config)# wireless tag rf default-rf-tag
   ```

4. Configure and name the dot11b policy.

   Example

   ```
   Device(config-wireless-rf-tag)# 24ghz-rf-policy default-dot11b-policy
   ```

5. Exit the sub-mode.

   Example

```
Device(config-wireless-rf-tag)# exit
```

6. Configure Cisco AP and enter the ethernet MAC address of the AP.

   Example

```
Device(config)# ap 9.12.15
```

7. Map a RF tag to the AP and enter the name of the RF tag.

   Example

```
Device(config-ap-tag)# rf-tag default-rf-tag
```

8. Exit the sub-mode.

   Example

```
Device(config-ap-tag)# exit
```

Verify Admin state on RF profile

This section explains how to verify Admin state on RF profile.

Use this command to verify the Admin state on RF profile.

- **show ap rf-profile name <RF_PROFILE> detail**

To verify the Admin state on RF profile, enter this command from the console:

```
Device# show ap rf-profile name iot-coex-custom-test-profile detail

Description                       : custom RF profile for IoT COEX testing
RF Profile Name                   : iot-coex-custom-test-profile
Band                              : 2.4 GHz
Transmit Power Threshold v1       : -70 dBm
Min Transmit Power                : -10 dBm
Max Transmit Power                : 30 dBm
Operational Rates
  802.11b 1M Rate                 : Mandatory
  802.11b 2M Rate                 : Mandatory
  802.11b 5_5M Rate               : Mandatory
  802.11b 11M Rate                : Mandatory
  802.11g 6M Rate                 : Supported
  802.11g 9M Rate                 : Supported
  802.11g 12M Rate                : Supported
  802.11g 18M Rate                : Supported
  802.11g 24M Rate                : Supported
  802.11g 36M Rate                : Supported
  802.11g 48M Rate                : Supported
  802.11g 54M Rate                : Supported
```

```
Max Clients                          : 200
Trap Threshold
  Clients                            : 12 clients
  Interference                       : 10%
  Noise                              : -70 dBm
  Utilization                        : 80%
Multicast Data Rate                  : auto
Rx SOP Threshold                     : auto
Band Select
  Probe Response                     :  Disabled
  Cycle Count                        :  2 cycles
  Cycle Threshold                    :  200 milliseconds
  Expire Suppression                 :  20 seconds
  Expire Dual Band                   :  60 seconds
  Client RSSI                        :  -80 dBm
  Client Mid RSSI                    :  -80 dBm
High Speed Roam
  hsr mode                           :  Disabled
Load Balancing
  Window                             :  5 clients
  Denial                             :  3 count
Coverage Data
  Data                               : -80 dBm
  Voice                              : -80 dBm
  Minimum Client Level               : 3 clients
  Exception Level                    : 25%
RSSI Settings
  RSSI Low Check                     : Disabled
  RSSI Threshold                     : -127 dbm
DCA Channel List                     : 1,6,11
Unused Channel List                  : 2,3,4,5,7,8,9,10
DCA Foreign AP Contribution          : Enabled
802.11n MCS Rates
  MCS  0                             : Enabled
  MCS  1                             : Enabled
  MCS  2                             : Enabled
  MCS  3                             : Enabled
  MCS  4                             : Enabled
  MCS  5                             : Enabled
  MCS  6                             : Enabled
  MCS  7                             : Enabled
  MCS  8                             : Enabled
  MCS  9                             : Enabled
  MCS 10                             : Enabled
  MCS 11                             : Enabled
  MCS 12                             : Enabled
  MCS 13                             : Enabled
  MCS 14                             : Enabled
  MCS 15                             : Enabled
  MCS 16                             : Enabled
  MCS 17                             : Enabled
  MCS 18                             : Enabled
  MCS 19                             : Enabled
  MCS 20                             : Enabled
  MCS 21                             : Enabled
  MCS 22                             : Enabled
  MCS 23                             : Enabled
  MCS 24                             : Enabled
  MCS 25                             : Enabled
  MCS 26                             : Enabled
  MCS 27                             : Enabled
  MCS 28                             : Enabled
```

```
  MCS 29                          : Enabled
  MCS 30                          : Enabled
  MCS 31                          : Enabled
State                             : Up
Client Aware FRA                  : Disabled
FRA Action                        : 2.4GHz/5GHz/Monitor
Client Network Preference         : default
Airtime Fairness Mode             : Disable
Airtime Stealing                  : N/A
Mesh Client Access Airtime Alloc  : Disabled
802.11ax
  OBSS PD                         : Disabled
  Non-SRG OBSS PD Maximum         : -62 dBm
  SRG OBSS PD                     : Disabled
  SRG OBSS PD Minimum             : -82 dBm
  SRG OBSS PD Maximum             : -62 dBm
NDP mode                          : Auto
Guard Interval                    : 800ns
A-MPDU Window                     : 255
Packets Retries Settings:
  Management Retries              : -1
  Aggregate Retries               : -1
  Non_Aggregate Retries           : -1
IOT Coexistence Admin State       : Enabled
```

Verify coexistence configuration applied to an AP on controller

This section explains how to verify coexistence configuration applied to an AP on controller.

Use this command to verify coexistence configuration applied to an AP on controller.

- **show ap name <AP_NAME>config general | include IOT**

To verify the coexistence configuration applied to an AP on controller, enter this command from the console:

```
Device# show ap name AP8C88.8152.CEE0 config general
Cisco AP Name   : AP8C88.8152.CEE0
================================================

Cisco AP Identifier                             : 940d.4b9c.24e0
Country Code                                    : US
Regulatory Domain Allowed by Country            : 802.11bg:-A   802.11a:-AB
 802.11 6GHz:-B
Radio Authority IDs                             : None
AP Country Code                                 : US  - United States
Country Code Resolution Method                  : Proximity
AP Regulatory Domain
  802.11bg                                      : -A
  802.11a                                       : -B
  802.11 6GHz                                   : -B
License Type                                    : CW
License State                                   : Non Compliant
Non Compliance Reason                           : Never Licensed
MAC Address                                     : 8c88.8152.cee0
IP Address Configuration                        : DHCP
IP Address                                      : 100.12.99.185
IP Netmask                                      : 255.255.0.0
Gateway IP Address                              : 100.12.0.1
CAPWAP Path MTU                                 : 1485
Capwap Active Window Size                       : 1
Telnet State                                    : Disabled
```

```
CPU Type                                          :  ARMv8 Processor rev 4 (v8l)
Memory Type                                        : DDR4
Memory Size                                        : 1771520 KB
SSH State                                          : Enabled
Serial Console State                               : Enabled
Cisco AP Location                                  : default location
AP Floor ID                                        : 0
AP Location Mode                                   : Unknown
Site Tag Name                                      : default-site-tag
RF Tag Name                                        : iot-coex-custom-test-tag
Policy Tag Name                                    : default-policy-tag
AP join Profile                                    : default-ap-profile
Flex Profile                                       : default-flex-profile
Tag Source                                         : Static
Static name                                        : FALSE
Primary Cisco Controller Name                      : katar
Primary Cisco Controller IP Address                : 100.12.45.46
Secondary Cisco Controller Name                    : Not Configured
Secondary Cisco Controller IP Address              : 0.0.0.0
Tertiary Cisco Controller Name                     : Not Configured
Tertiary Cisco Controller IP Address               : 0.0.0.0
Administrative State                               : Enabled
Operation State                                    : Registered
NAT External IP Address                            : 100.12.99.185
AP Certificate type                                : Manufacturer Installed
Certificate
AP Certificate Expiry-time                         : 08/09/2099 22:58:26
AP Certificate issuer common-name                  : High Assurance SUDI CA
AP Certificate Policy                              : Default
AP CAPWAP-DTLS LSC Status
    Certificate status        : Not Available
AP 802.1x LSC Status
    Certificate status        : Not Available
AP LSC authentication state                        : CAPWAP-DTLS
AP Mode                                            : Local
AP VLAN tagging state                              : Disabled
AP VLAN tag                                        : 0
CAPWAP Preferred mode                              : IPv4
CAPWAP UDP-Lite                                    : Not Configured
AP Submode                                         : Not Configured
Office Extend Mode                                 : Disabled
Link-Encryption                                    : Disabled
Dhcp Server                                        : Disabled
Remote AP Debug                                    : Disabled
Logging Trap Severity Level                        : information
Logging Syslog facility                            : kern
Software Version                                   : 26.2.0.4
Boot Version                                       : 1.1.2.4
Mini IOS Version                                   : 0.0.0.0
Stats Reporting Period                             : 180
LED State                                          : Enabled
MDNS Group Id                                      : 0
MDNS Rule Name                                     :
MDNS Group Method                                  : None
PoE Pre-Standard Switch                            : Disabled
PoE Power Injector MAC Address                     : Disabled
Power Type/Mode                                    : DC/Full Power
Number of Slots                                    : 3
AP Model                                           : CW9174I
IOS Version                                        : 26.2.0.4
Reset Button                                       : Disabled
AP Serial Number                                   : FVH2849A988
```

```
Management Frame Validation                      : Capable
Management Frame Protection                      : Not capable
AP User Name                                     : lab
AP 802.1X User Mode                              : Global
AP 802.1X User Name                              : Not Configured
Cisco AP System Logging Host                     : 255.255.255.255
Cisco AP Secured Logging TLS mode                : Disabled
AP Up Time                                      : 6 hours 3 minutes 37 seconds
AP CAPWAP Up Time                                : 6 minutes 16 seconds
Join Date and Time                               : 12/04/2025 12:04:53
Join Taken Time                                  : 3 seconds
Join Priority                                    : 1
AP Link Latency                                  : Disable
AP Lag Configuration Status                      : Disabled
Lag Support for AP                               : No
AP runs Dot11k/v LocalMAC                        : Yes
Rogue Detection                                  : Enabled
Rogue Containment auto-rate                      : Disabled
Rogue PMF Denial                               : Disabled
Rogue Containment of standalone flexconnect APs : Disabled
Rogue Detection Report Interval                  : 10
Rogue AP minimum RSSI                            : -90
Rogue AP minimum transient time                  : 0
AP TCP MSS Adjust                                : Enabled
AP TCP MSS Size                                  : 1250
AP IPv6 TCP MSS Adjust                           : Enabled
AP IPv6 TCP MSS Size                             : 1250
Hyperlocation Admin Status                       : Disabled
Retransmit count                                 : 5
Retransmit interval                              : 3
Kernel core dump :
  Fulldump capable                               : No
  Minidump capable                               : Yes
Macsec Capability                                : Capable
  Port 0
    Macsec Status                                : NA
    Macsec Ciphersuite                           : NA
  Port 1
    Macsec Status                                : NA
    Macsec Ciphersuite                           : NA
MLO Capability                                   : Capable
MLD Base MAC                                     : 960d.4b9c.24f0
Fabric status                                    : Disabled
FIPS status                                      : Disabled
WLANCC status                                    : Disabled
USB Module Type                                  : USB Module
USB Module State                                 : Disabled
USB Module Connected                             : No
USB Override                                     : Disabled
GAS rate limit Admin status                      : Disabled
WPA3 Capability                                  : Enabled
EWC-AP Capability                                : Disabled
AWIPS Capability                                 : Enabled
AID Management Capability                        : Enabled
Proxy Hostname                                   : Not Configured
Proxy Port                                       : Not Configured
Proxy NO_PROXY list                              : Not Configured
Proxy Username                                   : Not Configured
GRPC server status                               : Disabled
Unencrypted Data Keep Alive                      : Enabled
Local DHCP Server                                : Disabled
Traffic Distribution Statistics Capability       : Enabled
```

```
Dual DFS Statistics                                : Disabled
AP Upgrade Out-Of-Band Capability                  : Enabled
AP statistics                                      : Disabled
AP power derate Capability                         : Enabled
AP PMK Propagation Capability                      : Enabled
AP FTM Responder Capability                        : Enabled
AP FTM Initiator Capability                        : Enabled
AP UWB Responder Capability                        : Disabled
AP UWB Initiator Capability                        : Disabled
AP Client FTM capability                           : Enabled
URWB Capability                                    : Not Capable
6GHz Standard-Power mode                           : Not Allowed
Mesh DCA Run Status                                : N/A
Last Mesh DCA Run                                  :
Meraki Capable AP                                  : Yes
AP Console Speed                                   : 115200
Tilt Angle                                         : 176 Degree(s)
Radio Reset Reason Statistics
Slot 0
  Unknown Reason                                                    : 1
Slot 1
  Unknown Reason                                                    : 1
Slot 2
  Unknown Reason                                                    : 1
Slot 3

Radio Failure Reason Statistics
Slot 0
Slot 1
Slot 2
Slot 3
AP image integrity
  Time                                             : 12/28/2024 02:00:01
  Alterative image loaded                          : No
  Backup image status
    Version                                        : 26.1.0.40
    Partition                                      : part2
    Kernel                                         : Good
    Root FS                                        : Good
    IOX                                            : Good
  Primary image status
    Version                                        : 26.2.0.4
    Partition                                      : part1
    Kernel                                         : Good
    Root FS                                        : Good
    IOX                                            : Good
IOT Coexistence Admin State                        : Enabled
IOT Coexistence Operational State                  : Enabled
```

Verify coexistence summary of all the APs

This section explains how to verify coexistence summary of all the APs.

Use this command to verify coexistence summary of all the APs.

- **show wireless iot-coexistence summary**

To verify the coexistence summary of all the APs, enter this command from the console:

```
Device# show wireless iot-coexistence summary
```

```
------------------------------------------------------------------------------------------------------------------------------------

                                                                                                                    |
                                        IOT COUNTERS
AP Name                                Radio MAC       Coex Admin  Coex Oper    |
 Tx pkts    Rx pkts        CRC    Low Pri   High Pri    Low Pri   High Pri
 Low Pri   High Pri
                                                       State       State        |
                          errors   requests   requests     denied     denied
Tx abort   Tx abort
------------------------------------------------------------------------------------------------------------------------------------
AP8C88.8152.CEE0                  940d.4b9c.24e0  Enabled     Enabled
     359        344          0         72          72          7          3
       1          0
AP24D7.9CBD.F290                  c828.e526.a1c0  Disabled    Not supported
```

Verify coexistence Wi-Fi statistics

This section explains how to verify coexistence Wi-Fi statistics.

Use this command to verify coexistence Wi-Fi statistics.

- **show ap name <AP_NAME> iot coexistence statistics**

To verify the coexistence Wi-Fi statistics, enter this command from the console:

```
Device# show ap name AP8C88.8152.CEE0 iot coexistence statistics

AP name                    : AP8C88.8152.CEE0
AP MAC Address             : 940d.4b9c.24e0
IOT Coex Admin state       : Enabled
IOT Coex Oper state        : Enabled
Last AP response code      : SUCCESS
IOT get stats err cnt      : 0
IOT Tx packets             : 580
IOT Rx packets             : 561
IOT CRC errors             : 0
IOT low pri requests       : 116
IOT high pri requests      : 116
IOT low pri denied         : 11
IOT high pri denied        : 5
IOT low pri tx aborted     : 3
IOT high pri tx aborted    : 0
WiFi get stats err cnt     : 0
WiFi Tx packets            : 0
WiFi Rx packets            : 0
WiFi CRC errors            : 0
WiFi low pri requests      : 0
WiFi high pri requests     : 0
WiFi low pri denied        : 0
WiFi high pri denied       : 0
WiFi low pri tx aborted    : 0
WiFi high pri tx aborted   : 0
Last Rcvd Time             : 12/04/2025 12:15:01
```

46 Smart Spaces - AnyLocate capabilities

Topics:

- AnyLocate feature set

AnyLocate feature set

Introduces the AnyLocate feature set, outlining capabilities for enhancing indoor location services using UWB and Wi-Fi FTM, and details technologies and workflows for indoor wayfinding, asset tracking, and AP auto-locate across supported Cisco APs.

AnyLocate capabilities are a suite of features that

- enhance indoor location services
- leverage technologies like ultra-wideband (UWB) and Wi-Fi fine timing measurement (FTM), and
- enable precise indoor wayfinding and asset tracking.

AnyLocate aims to transform the network into a sensor, eliminating the need for additional sensors or gateways by utilizing APs to facilitate location services. This umbrella term encompasses various location-based functionalities, including client location, RFID tags, BLE tags, AP auto-locate, UWB based indoor wayfinding and asset tracking.

Key features:

Indoor Wayfinding: Provides turn-by-turn directions within a building, similar to GPS for outdoor navigation.

Asset Tracking: Monitors the location of assets equipped with tracking tags.

AP Auto-locate: Automatically places APs on network maps using FTM and UWB based inter-AP ranging and GNSS data.

Wi-Fi fine timing measurement

Wi-Fi fine timing measurement (FTM) is a 802.11 standard feature that

- allows devices to measure their distance from Wi-Fi APs
- enhances indoor location accuracy, and
- uses time-stamped signal exchanges for ranging.

FTM improves upon traditional Wi-Fi signal strength (RSSI) based location by using precise time-stamping to calculate distances more accurately.

Supported APs

- Wi-Fi 6: Cisco Catalyst 9130 and 9120 Series APs
- Wi-Fi 6E APs
- Wi-Fi 7 APs

> **Note**
>
> For more information, check the Infrastructure Feature Matrix (IOS XE) tables for Wi-Fi 6 and Wi-Fi 7 APs.

Indoor wayfinding

Indoor wayfinding is a highly sought-after feature that

- allows clients to compute their own location within a building
- provides turn-by-turn directions to a desired destination, and

- functions as an indoor equivalent to outdoor GPS navigation.

Indoor wayfinding operates using these protocols.

- Client-side computation: The client device computes its own location.
- Technology leverage: Utilizes UWB DL-TDOA and Wi-Fi FTM for precise positioning.

Asset tracking

Asset Tracking is a feature that

- enables the precise monitoring of physical assets equipped with location tags
- leverages ultra-wideband (UWB) and bluetooth low energy (BLE) technologies, and
- allows the network infrastructure to compute and display the asset's location.

Unlike wayfinding, where the client computes its own location, asset tracking involves the network infrastructure actively listening for signals from specialized tags attached to assets. The network then calculates the tag's position, making it ideal for inventory management, equipment location, and other operational use cases.

Asset tracking operates using these protocols.

- Tag-initiated blinks: UWB-enabled tags periodically transmit short "blinks."
- Network-side listening: APs listen for these blinks, record their reception timestamps, and send this data to a central location engine.
- Infrastructure computation: The network infrastructure computes the tag's precise location based on the time differences of arrival at multiple APs.
- Hybrid tag support: Tags often combine BLE for onboarding and traditional location, with UWB for higher accuracy when available.

Feature history for Smart Spaces - AnyLocate capabilities

This section describes the feature history details for the Smart Spaces - AnyLocate capabilities feature.

- This table provides release and related information for the feature explained in this module.
- This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 59: Feature history for Smart Spaces - AnyLocate capabilities

Feature Name	Release Information	Feature Description
Smart Spaces - AnyLocate capabilities	Cisco IOS XE 17.18.2	This is a suite of features that enhance indoor location services leveraging technologies like ultra-wideband (UWB), Wi-Fi fine timing measurement (FTM), and enable precise indoor wayfinding and asset tracking.

Prerequisites for asset tracking

This section describes the prerequisites for asset tracking.

There are four very basic device level prerequisites for asset tracking.

- Enable the global configuration knob for ranging.
- Enable UWB radio in the AP profile (the default setting is enabled).
- Allow and ensure a functional gRPC connection to Cisco Spaces Connector.
- Add the AP to the Cisco Spaces floor map and enable IoT services.

> **Note**
>
> Enabling IoT services through Cisco Spaces Connect feature is not supported yet.

After you meet these prerequisites, go to Cisco Spaces to enable the asset tracking feature, onboard tags, and use the detect and locate functions to track assets.

To verify the prerequisites for asset tracking, check Verify the prerequisites for asset tracking on page 684.

Time difference of arrival (TDOA)

This topic describes time difference of arrival (DOA) details.

Time difference of arrival (TDOA) operates in two ways

- Downlink time difference of arrival
- Uplink time difference of arrival

Downlink time difference of arrival (DL-TDOA)

Downlink time difference of arrival (DL-TDOA) is a UWB-based positioning method that

- allows client devices to compute their own location
- leverages time-synchronized UWB signals transmitted by APs, and
- ensures client privacy because location computation occurs on the device.

In a DL-TDOA system, APs transmit time-stamped UWB signals. Client devices passively listen to these signals, measure the time differences of arrival from multiple APs, and use this information to calculate their precise indoor position. This method is highly scalable and ensures user privacy because the network does not compute or store the client's location.

Uplink time difference of arrival (UL-TDOA)

Uplink time difference of arrival (UL-TDOA) is a UWB-based positioning method that

- enables the network infrastructure to compute the location of tracking tags
- relies on tags periodically transmitting UWB blinks, and
- is primarily used for asset tracking applications.

In a UL-TDOA system, UWB-enabled tags (attached to assets) periodically transmit short "blinks." APs listen to these blinks, record their reception timestamps, and forward this data to a central location engine. The network infrastructure then computes the tag's precise location based on the time differences of arrival at multiple APs.

Automatic reclustering for clock synchronization

This topic describes automatic reclustering for clock synchronization details.

Summary

Automatic reclustering for clock synchronization is the automated process of reconfiguring the UWB network's clusters and roles when an AP joins, leaves, or undergoes configuration changes, ensuring continuous and optimized location services.

- Wireless LAN Controller : Detects AP status changes and initiates reclustering requests.
- Cisco Spaces: Receives reclustering requests, initiates AP to AP ranging over UWB if necessary, computes new cluster configurations, and pushes them to the controller.
- APs: Apply new cluster configurations and participate in UWB location services.

Workflow

Reclustering behavior helps you anticipate network adaptation when APs change.

Prerequisites for wayfinding configurations

This topic describes the prerequisites for wayfinding configurations.

Ensure these prerequisites are met for UWB wayfinding functionality.

- Cisco Spaces integration: The controller must be onboarded to Cisco Spaces using a connector.
- AP placement on maps: Place all APs accurately on a map and associate them with a floor map in Cisco Spaces.
- Geo-coordinates: Enter geo-coordinates for each AP. Using AP Autolocate is preferred.
- Height and floor level: Record each AP’s height above the floor and specify its floor level.
- WLAN beacon broadcast: Enable beacon broadcast for at least one WLAN on each AP to support UWB Out-of-Band (OOB) communication with mobile devices.
- UWB coverage density: Deploy APs so that each AP covers 800 to 1000 square feet and provides adequate UWB coverage.

> **Note**
>
> Perform a proper RF design. Position each AP with equal importance as the AP density.

Impact of AP downtime on location services

This topic describes impact of AP downtime on location services.

An AP going down can affect UWB location service coverage.

- Localized blackout: If an AP goes down in your environment, you may lose location services in that area.
- Coverage resilience: In the event that one AP in your deployment becomes non-operational while four other anchors remain active, your location services continue to function. If there are fewer than four anchors in a specific area, computed location might be degraded or inaccurate.
- Maintenance window: Expect AP downtime during maintenance windows or network events. The system automatically recovers by reclustering when these events occur.

Enable UWB wayfinding (GUI)

Use this procedure to enable UWB wayfinding using the GUI.

Activate UWB wayfinding to enhance location services on an AP.

Before you begin

UWB must be enabled to proceed with this task.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. Select the **AP Join Profile**.
3. Click on the **Geolocation** tab.
4. In the **Ultra-Wide Band (UWB)** section, toggle the button to enable **UWB Wayfinding**.
5. The **UWB Wayfinding Block Duration** value by default is one second.

Enable UWB wayfinding (CLI)

Use this procedure to enable UWB wayfinding using commands.

Enable UWB wayfinding and configure its block duration on a wireless AP profile using commands.

Procedure

1. Enter global configuration guide.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile1
   ```

3. Enable geolocation UWB wayfinding.

   Example

   ```
   Device(config-ap-profile)# [no] geolocation uwb wayfinding
   ```

   Use the **no** form of the command to disable UWB wayfinding.

4. Configure the geolocation UWB wayfinding block duration.

   Example

   ```
   Device(config-ap-profile)# geolocation uwb wayfinding block-duration 1second
   ```

The **UWB Wayfinding Block Duration** value by default is one second. The **UWB Wayfinding Block Duration** has two values: one second and 500 ms.

Configure height above the floor (GUI)

Use this procedure to configure height above the floor using the GUI.

Set the physical mounting height of an AP in centimeters as part of the AP provisioning process using the GUI.

Use this task during bulk AP provisioning to ensure accurate floor mapping and optimal wireless coverage by specifying a realistic height above floor.

Procedure

1. Choose **Configuration** > **Wireless** > **Bulk AP Provisioning**.
2. Click on **Select a workflow to create a AP Provisioning task.**
3. Select the AP and click **Next**.
4. In the **Floor** section, enter the **Height Above Floor (centimeters)** value.
   The **Height Above Floor (centimeters)** value's valid range is from 1 to 4000.
5. Click **Next**.

What to do next

To verify the configuration, check Verify AP location information on page 684.

Configure height above the floor (CLI)

Use this procedure to configure height above the floor using commands.

Specify the height of an AP above the floor to improve the accuracy of location services using commands.

The height above the floor (HATF) information is crucial for 3D location computations, especially for wayfinding and asset tracking use cases. This configuration can be done per AP.

Before you begin

- Ensure the AP is operational and managed by the controller.
- Know the precise height of the AP from the floor in centimeters.

Procedure

Configure the geolocation height above the floor for a Cisco AP.

Example

```
Device# ap name cisco-ap-name geolocation height-above-the-floor 1-4000
```

The valid range is from 1 to 4000 cm. Use the **no** form of the command to disable height above the floor for a Cisco AP.

The AP's height above the floor is set for use by location services.

What to do next

To verify the configuration, check Verify AP location information on page 684.

Verify AP UWB wayfinding status

This section provides the verifying AP UWB wayfinding status details.

Use these commands to verify the status of AP UWB wayfinding.

- **Device# show ap name <ap_name> geolocation uwb wayfinding**
- **Device# show ap geolocation uwb wayfinding detail**
- **Device# show ap geolocation uwb wayfinding status**
- **Device# show ap geolocation uwb wayfinding group <group name>**

To verify the AP geolocation UWB wayfinding, enter this command from the console:

```
Device# show ap name <ap_name> geolocation uwb wayfinding

AP MAC Address: f0d8.054c.12e0
Group ID: new_curl_group1
Status: Active
Number of Clusters: 1
Role: Initiator
Round Index: 0
Number of Responder Anchor: 1
Responder MAC    Slot Index
--------------------------
ecf4.0c92.b340   1
```

To verify the AP geolocation UWB wayfinding detail when AP is active, enter this command from the console:

```
Device# show ap geolocation uwb wayfinding detail
AP Name AP MAC Site Tag Group ID Status Block
Duration
NAlpine2 ecf4.0c92.b340 default-site-tag new_curl_group Active 1sec
NAlpine3 ecf4.0c92.b980 default-site-tag new_curl_group Active 1sec
NAlpine1 ecf4.0c9c.bd60 default-site-tag new_curl_group Active 1sec
NZealand1 f0d8.054c.12e0 default-site-tag new_curl_group Active 1sec
```

To verify the AP geolocation UWB wayfinding detail when the AP is in paused state, enter this command from the console:

```
Device# show ap geolocation uwb wayfinding detail
AP Name AP MAC Site Tag Group ID Status Block
Duration
NAlpine2 ecf4.0c92.b340 default-site-tag new_curl_group Paused 1sec
NAlpine3 ecf4.0c92.b980 default-site-tag new_curl_group Paused 1sec
NAlpine1 ecf4.0c9c.bd60 default-site-tag new_curl_group Paused 1sec
NZealand1 f0d8.054c.12e0 default-site-tag new_curl_group Paused 1sec
```

To verify the AP geolocation UWB wayfinding detail when the AP is in pending state, enter this command from the console:

```
Device# show ap geolocation uwb wayfinding detail
AP Name AP MAC Site Tag Group ID Status Block
Duration
NAlpine2 ecf4.0c92.b340 default-site-tag new_curl_group Pending 1sec
NAlpine3 ecf4.0c92.b980 default-site-tag new_curl_group Pending 1sec
NAlpine1 ecf4.0c9c.bd60 default-site-tag new_curl_group Pending 1sec
NZealand1 f0d8.054c.12e0 default-site-tag new_curl_group Pending 1sec
```

To verify the AP geolocation UWB wayfinding status, enter this command from the console:

```
Device# show ap geolocation uwb wayfinding status
Group-ID                        Clustered APs State        Active APs

1760584021797-1                 6             Configured   6
```

To verify the AP geolocation UWB wayfinding group, enter this command from the console:

```
Device# show ap geolocation uwb wayfinding group <group name>
Group ID: 1760584021797-1
Max Round: 7
Cluster List
Round Initiator Anchor                 Stratum Parent Anchor
  List of Responder Anchors (Slot-Index:Responder)

0     Alpine3                          0
  1:Zealand1 2:Alpine2 3:Alpine1 4:Alpine4 5:Zealand2
```

Verify AP UWB asset-tracking status

This section provides the verifying AP UWB asset-tracking status details.

Use these commands to verify the status of AP UWB asset-tracking.

- **Device# show ap name <ap_name> geolocation uwb asset-tracking**
- **Device# show ap geolocation uwb asset-tracking status**
- **Device# show ap geolocation uwb asset-tracking detail**

To verify the AP geolocation UWB asset-tracking, enter this command from the console:

```
Device# show ap name <ap_name> geolocation uwb asset-tracking


AP MAC Address: f0d8.054c.12e0
Group ID: ABC1231
Status: Active
Sync slot index: 0
Primary: 1
RX Bitmap: 0X00000000FFFFFFFF
```

To verify the AP geolocation UWB asset-tracking status, enter this command from the console:

```
Device# show ap geolocation uwb asset-tracking status


Group-ID                        Clustered APs State        Active APs


------------------------------------------------------------------------


ABC1231                         4             Configured   4
ne2 3:Alpine1 4:Alpine4 5:Zealand2
```

To verify the AP geolocation UWB asset-tracking detail, enter this command from the console:

```
Device# show ap geolocation uwb asset-tracking detail
```

```
AP Name                          AP MAC           Site Tag
      Group ID                         Status


_____________________________________________________________________________________


NAlpine2                         ecf4.0c92.b340   default-site-tag
      ABC1231                          Active


NAlpine3                         ecf4.0c92.b980   default-site-tag
      ABC1231                          Active


NAlpine1                         ecf4.0c9c.bd60   default-site-tag
      ABC1231                          Active


NZealand1                        f0d8.054c.12e0   default-site-tag
      ABC1231
```

Verify AP location information

This section provides verifying AP location information details.

Verify AP location information for these parameters:

- Verify height above the floor and latitude and longitude parameters from Cisco Spaces autoplacement or on the controller's console or GUI.

To verify height above the floor and latitude and longitude parameters from Cisco Spaces autoplacement or on the controller's console or GUI, enter these commands:

```
Device# ap name <ap_name> geolocation coordinate latitude longitude

Device# show ap name <ap_name> geolocation detail

AP Name                          : Alpine1
GNSS present                     : Yes
Location type                    : Ellipse
 HDOP                            : 0.000000
 Source                          : GNSS
 Center ellipse - longitude      : -121.928826
 Center ellipse - latitude       : 37.407565
 Ellipse major-axis (meters)     : 86
 Ellipse minor-axis (meters)     : 86
 Ellipse orientation             : 0.000000
 Last update                     : 12/03/2025 22:32:15
Height                           : NA
Height above the floor (cms)     : 200
```

Verify the prerequisites for asset tracking

This section provides the verifying the prerequisites for asset tracking details.

Verify the prerequisites for asset tracking for these parameters:

- Enable the global configuration knob for ranging.

- Enable UWB radio in the AP profile (the default setting is enabled).
- Allow and ensure a functional gRPC connection to Cisco Spaces Connector.
- Verify the controller to connector connection status

To enable the global configuration knob for ranging, use this command: **Device(config)# ap geolocation derivation ranging**

To check the UWB radio enablement in the AP profile, see Enable UWB wayfinding (GUI) on page 680 or Enable UWB wayfinding (CLI) on page 680.

Allow and ensure a functional gRPC connection to Cisco Spaces Connector by checking the controller connector connection.

To verify the controller to connector connection status, enter this command:

```
Device# show telemetry connection all
Telemetry connections

Index Peer Address               Port  VRF Source Address              State
    State Description
----- ------------------------- ----- --- ------------------------- ----------
 --------------------
    0 70.0.0.132                  8184 0   70.0.0.2                   Active
    Connection up
    1 70.0.0.132                  8004 0   70.0.0.2                   Active
    Connection up
  395 172.19.20.125             25103 1 M 172.20.209.40              Connecting
 Connection request made to transport handler
```

IoT service enablement is done from Cisco Spaces.

Part VI

Radio Resource Management

Topics:

- Radio Resource Management
- Coverage Hole Detection
- Optimized Roaming
- Cisco Flexible Radio Assignment
- XOR Radio Support
- Cisco Receiver Start of Packet
- Client Limit
- IP Theft
- Unscheduled Automatic Power Save Delivery
- Target Wake Time
- Enabling USB Port on Access Points
- Dynamic Frequency Selection
- Cisco Access Points with Tri-Radio
- Cisco Catalyst Center Assurance Wi-Fi 6 Dashboard
- Antenna Disconnection Detection
- Neighbor Discovery Protocol Mode on Access Points
- 6-GHz Band Operations

47 Radio Resource Management

Topics:

- Radio resource management
- How to configure RRM
- Monitor RRM parameters and RF group status
- Event-Driven Radio Resource Management
- Rogue detection - rogue channel width

Radio resource management

Introduces radio resource management, covering real-time RF parameter control, AP monitoring, automated optimization, and key functions like radio resource monitoring, power control, dynamic channel assignment, coverage hole detection, and RF grouping to enhance wireless network performance..

A Radio Resource Management (RRM) is a system that

- consistently manages real-time RF parameters of a wireless network
- monitors associated APs for traffic load, interference, noise, coverage, and other metrics,
- performs critical functions like radio resource monitoring, power control transmission, dynamic channel assignment, and coverage hole detection and correction.

Feature history

Table 60: Feature history table for radio resource management

Feature name	Release information	Feature description
Radio resource management	Cisco IOS XE 16.10.1	Radio Resource Management (RRM) is a feature that automates and optimizes the management of radio frequencies in a wireless network. It enables continuous monitoring of access points (APs) for metrics such as traffic load, interference, noise, and coverage.

Functions of radio resource management

- Radio Resource Monitoring: Ensures optimal allocation of network resources.
- Power Control Transmission: Adjusts power levels to maintain network performance.
- Dynamic Channel Assignment: Allocates channels dynamically to reduce interference and optimize network performance.
- Coverage Hole Detection and Correction: Identifies and rectifies gaps in coverage to ensure consistent connectivity.
- RF Grouping: Groups RF resources effectively to manage interference and optimize performance.

Radio resource monitors

Radio resource monitor details.

Radio resource monitor is a system that

- detects and configures new devices and APs automatically
- adjusts associated APs for optimal coverage and capacity, and
- supports noise and interference monitoring.
- APs scan all the valid channels for the country of operation as well as for channels available in other locations.
- The APs in local mode go *offchannel* for a period not greater than 70 ms to monitor these channels for noise and interference.

- Packets collected during this time are analyzed to detect rogue APs, rogue clients, ad-hoc clients, and interfering APs.
- In the presence of voice traffic or other critical traffic (in the last 100 ms), APs can defer off-channel measurements. The APs also defer off-channel measurements based on the WLAN scan priority configurations.
- Each AP spends only 0.2 percent of its time off channel. This activity is distributed across all the APs so that adjacent APs are not scanning at the same time, which could adversely affect wireless LAN performance.

Mobility controller and mobility agent

Radio frequency groups

A radio frequency group is a collection of controllers that

- coordinate RRM globally
- support separate networks for 2.4 GHz and 5 GHz networks, and
- optimize network calculations on a per-radio basis.

Clustering Cisco Catalyst 9800 Series Wireless Controller into a single RF group enables the RRM algorithms to scale beyond the capabilities of a single Cisco Catalyst 9800 Series Wireless Controller.

RF group creation

Create an RF group using these parameters:

- User-configured RF network name.
- Neighbor discovery performed at the radio level.
- Country list configured on the controller.

RF grouping function run between controllers .

> **Note**
>
> RF groups and mobility groups are similar, in that, they both define clusters of controllers , but they are different in terms of their use. An RF group facilitates scalable, system-wide dynamic RF management, while a mobility group facilitates scalable, system-wide mobility and controller redundancy.

RF neighborhood

APs periodically send out neighbor messages over the air. APs using the same RF group name validate messages from each other.

When APs on different controllers hear validated neighbor messages at a signal strength of -80 dBm (or stronger), the controllers dynamically form an RF neighborhood in auto mode. In static mode, the leader is manually selected and the members are added to the RF Group.

RF group leader

Analyzes real-time radio data collected by the system, calculating the power and channel assignments.

An RF group leader is a designated device that

- analyzes real-time radio data collected by the system

- calculates power and channel assignments, and
- sends them to each controller in the RF group.

The RRM algorithms ensure system-wide stability, and restrain channel and power scheme changes to the appropriate local RF neighborhoods.

RF Group Leader is selected based on the controller with the greatest AP capacity (platform limit). If multiple controllers have the same capacity, the leader is selected based on the Group ID, which is a combination of the management IP address, AP capacity, random number, and so on. The one with the highest Group ID is selected as the leader.

RF Group Leader can be configured in two ways as follows:

- Auto Mode: In this mode, the members of an RF group elect an RF group leader to maintain a *primary* power and channel scheme for the group. The RF grouping algorithm dynamically chooses the RF group leader and ensures that an RF group leader is always present. Group leader assignments can and do change (for instance, if the current RF group leader becomes inoperable or RF group members experience major changes).
- Static Mode: In this mode, a user selects a controller as an RF group leader manually. In this mode, the leader and the members are manually configured and fixed. If the members are unable to join the RF group, the reason is indicated. The leader tries to establish a connection with a member every minute if the member has not joined in the previous attempt.

Note

- When a controller becomes both leader and member for a specific radio, you get to view the IPv4 and IPv6 address as part of the group leader.

  When a Controller A becomes a member and Controller B becomes a leader, the Controller A displays either IPv4 or IPv6 address of Controller B using the address it is connected.

  So, if both leader and member are not the same, you get to view only one IPv4 or IPv6 address as a group leader in the member.

Pinning and cascading

If Dynamic Channel Assignment (DCA) needs to use the worst-performing radio as the single criterion for adopting a new channel plan, it can result in pinning or cascading problems.

The main cause of both pinning and cascading is that any potential channel plan changes are controlled by the RF circumstances of the worst-performing radio. The DCA algorithm does not do this; instead, it does the following:

- Multiple local searches: The DCA search algorithm performs multiple local searches initiated by different radios in the same DCA run rather than performing a single global search that is driven by a single radio. This change addresses both pinning and cascading, while maintaining the desired flexibility and adaptability of DCA and without jeopardizing stability.
- Multiple Channel Plan Change Initiators (CPCIs): Previously, the single worst radio was the sole initiator of a channel plan change. Now each radio in an RF group is evaluated and prioritized as a potential initiator. Intelligent randomization of the resulting list ensures that every radio is eventually evaluated, which eliminates the potential for pinning.
- Limiting the propagation of channel plan changes (Localization): For each CPCI radio, the DCA algorithm performs a local search for a better channel plan, but only the CPCI radio itself and its one-hop neighboring access points are actually allowed to change their current transmit channels. The impact of an access point triggering a channel plan change is felt only to within two RF hops from that access point, and the actual channel plan changes are confined to within a one-hop RF neighborhood. Because this limitation applies across all CPCI radios, cascading cannot occur.

- Non-RSSI-based cumulative cost metric: A cumulative cost metric measures how well an entire region, neighborhood, or network performs with respect to a given channel plan. The individual cost metrics of all the access points in that area are considered in order to provide an overall understanding of the channel plan's quality. These metrics ensure that the improvement or deterioration of each single radio is factored into any channel plan change. The objective is to prevent channel plan changes in which a single radio improves, but at the expense of multiple other radios experiencing a considerable performance decline.

The RRM algorithms run at a specified updated interval, which is 600 seconds by default. Between update intervals, the RF group leader sends keepalive messages to each of the RF group members and collects real-time RF data.

Note

Several monitoring intervals are also available. See the Configuring RRM section for details.

RF grouping failure reason codes

RF Grouping failure reason codes and their explanations are listed below:

Table 61: RF Grouping Failure Reason Codes

Reason Code	Description
1	Maximum number (20) of controllers are already present in the group.
2	If the following conditions are met: • The request is from a similar powered controller and, • Controller is the leader for the other band, OR • Requestor group is larger.
3	Group ID do not match.
4	Request does not include source type.
5	Group spilt message to all member while group is being reformed.
6	Auto leader is joining a static leader, during the process deletes all the members.
9	Grouping mode is turned off.
11	Country code does not match.
12	Controller is up in hierarchy compared to sender of join command (static mode). Requestor is up in hierarchy (auto mode).
13	Controller is configured as static leader and receives join request from another static leader.
14	Controller is already a member of static group and receives a join request from another static leader.
15	Controller is a static leader and receives join request from non-static member.

Reason Code	Description
16	Join request is not intended to the controller. Controller name and IP do not match.
18	RF domain do not match.
19	Controller received a Hello packet at incorrect state.
20	Controller has already joined Auto leader, now gets a join request from static leader.
21	Group mode change. Domain name change from CLI. Static member is removed from CLI.
22	Max switch size (350) is reached

Additional Reference

Radio Resource Management White Paper: https://www.cisco.com/c/en/us/td/docs/wireless/controller/technotes/8-3/b_RRM_White_Paper/b_RRM_White_Paper_chapter_011.html

RF group name

An RF group name is a user-configured identifier assigned to a group of wireless LAN controllers that

- is sent to all the access points joined to the controller
- coordinate to perform Radio Resource Management (RRM) in a globally optimized manner,
- serves as a shared secret for generating hashed MIC in neighbor messages, and
- helps in avoiding RF interference and contention by ensuring system-wide RRM.

To create an RF group, you configure all the controllers to be included in the group with the same RF group name. If there is any possibility that an AP joined to a controller might hear RF transmissions from an AP on a different controller , you should configure the controller with the same RF group name.

If RF transmissions between APs can be heard, then system-wide RRM is recommended to avoid 802.11 interference and contention as much as possible.

Rogue AP detection in RF groups

Rogue AP Detection in RF Groups.

A rogue AP detection in an RF group is a wireless security feature that

- enables APs in the group to identify unauthorized neighboring APs
- uses beacon and probe-response frames to check for matching authentication elements, and
- automatically reports and records unauthorized devices for network monitoring and response.

After you create an RF group of controllers, you must configure connected access points to detect rogue APs. APs analyze neighbor messages for matching authentication elements. If a match is found, the AP authenticates the frame. If the authentication elements do not match, the AP reports the neighboring AP as rogue, logs its BSSID, and sends the log to the controller or embedded controller.

Secure RF groups

Secure RF groups enable encryption and security for RF grouping and RRM message exchanges over a DTLS tunnel.

Secure RF groups are a wireless LAN controller feature that

- encrypt radio frequency grouping and RRM message exchanges over a DTLS tunnel
- authenticate controllers using wireless management trust-point certificates during the DTLS handshake, and
- require all participating controllers to belong to the same mobility group.

RF profile configuration recommendation

RF group configuration recommendations.

The RF profile configuration recommendations are:

- Ensure that the country code configuration on both leader and member controllers match.
- RF grouping formation will not form if:
  - The group leader has IPv4 address and the group member has IPv6 address.
  - The group leader has IPv6 address and the group member has IPv4 address.
- Configure the IPv4 address on the group leader if RF grouping occurs between leader and member using IPv6 address.
- We recommend having the same RF profile configurations on the RF group leader and member controllers.
  - Since the RRM algorithms run on the group leader, they refer to the RF profile configurations of the leader controller. For example, additional channels in the member controller's list are not included by the group leader when it assigns a channel to the radio. Similarly, if there is a difference in configuration for DBS channel width between the leader and member controller, the algorithm only refers to the configurations of the group leader.
- For premises with over 20 controllers, we recommend that you have a common (non-default) RF network name for up to 20 controllers. The rest of the controllers can be mapped under different RF network names.
- For better network control, set your controller as static leader.

Examples: RF group configuration

RF Group Configuration examples.

These are examples of RF group name configuration.

```
Device# configure terminal
Device(config)# wireless rf-network test1
Device(config)# ap dot11 24ghz shutdown
Device(config)# end
Device# show network profile 5
```

This example demonstrates how to configure rogue AP sdetection within RF groups.

```
Device# ap name ap1 mode clear
Device# end
Device# configure terminal
Device(config)# wireless wps ap-authentication
Device(config)# wireless wps ap-authentication threshold 50
Device(config)# end
```

Transmit power control

A transmit power control is an automation algorithm that:

- increases and decreases an access point's power dynamically
- responds to changes in the RF coverage environment, and
- provides enough RF power to achieve the required coverage levels while avoiding channel interference.

This feature is different from coverage hole detection, which is primarily concerned with clients.

- TPC provides enough RF power to achieve the required coverage levels while avoiding channel interference between APs. We recommend that you select TPCv1; TPCv2 option is deprecated.
- With TPCv1, you can select the channel aware mode; we recommend that you select this option for 5 GHz, and leave it unchecked for 2.4 GHz.

Override the TPC algorithm with minimum and maximum transmit power settings

Override the TPC Algorithm with Minimum and Maximum Transmit Power Settings.

A Transmit Power Control (TPC) minimum and maximum transmit power setting is a wireless network configuration option that

- defines the allowed range of RF transmit powers for APs
- overrides the automatic power adjustment recommendations of the TPC algorithm, and
- applies settings globally to APs through RF profiles.

The TPC (Transmit Power Control) algorithm automatically balances RF power in various environments. Occasionally, site or architectural constraints require manually overriding TPC recommendations. With minimum and maximum power settings, you ensure APs do not exceed or fall below specific transmit powers, regardless of TPC or automatic adjustments or coverage hole detection.

Each AP model and each regulatory domain has its own allowed power levels. The increments vary: Cisco APs use 3 dB increments, but settings can be chosen in 1 dB increments and rounded.

To set the Maximum Power Level Assignment and Minimum Power Level Assignment, enter the maximum and minimum transmit power used by RRM in the fields in the **Tx Power Control** window. The range for these parameters is from -10 to 30 dBm. The minimum value cannot be greater than the maximum value; the maximum value cannot be less than the minimum value.

If you configure a maximum transmit power, RRM does not allow any access point attached to the controller, to exceed this transmit power level (whether the power is set by RRM TPC or by coverage hole detection). For example, if you configure a maximum transmit power of 11 dBm, no AP will transmit above 11 dBm unless the AP is configured manually.

Cisco APs support power level changes in 3 dB granularity. TPC Min and Max power settings allow for values in 1 dB increments. The resulting power level will be rounded to the nearest value supported in the allowed powers entry for the AP model and the current serving channel.

Each AP model has its own set of power levels localized for its regulatory country and region. Moreover, the power levels for the same AP model will vary based on the band and channel it is set to. For more information about Allowed Power Level versus Actual Power (in dBm), use the **show ap name <name> config slot <0|1|2|3>** command. This command displays the specific number of power levels, the allowed range of power levels, and the current power level setting on the AP.

Dynamic channel assignment

Dynamic Channel Assignment (DCA) minimizes adjacent channel interference between APs.

A dynamic channel assignment (DCA) is a wireless LAN management technique that

- automatically evaluates radio frequency (RF) conditions and network utilization
- dynamically allocates channels among APs to minimize interference and maximize performance, and
- continuously updates channel assignments based on system-wide RF analytics and policies.

Features of DCA

Features of DCA are:

- **Dynamic channel allocation**: DCA dynamically assigns channels to APs to avoid conflicts and interference, improving network capacity and performance. Two adjacent APs on the same channel can cause signal contention or collision. In a collision, data is not received by the AP. For example, reading an e-mail in a café can affect the performance of an AP in a neighboring business.

  Even though these are separate networks, someone sending traffic to the café on channel 1 can disrupt communication in an enterprise using the same channel. Devices can dynamically allocate AP channel assignments to avoid conflict and increase capacity and performance.

- **Channel reuse**: Efficiently reuses channels by assigning the same channel to APs that are physically far apart, maximizing scarce RF resources. In other words, channel 1 is allocated to a different AP far from the café, which is more effective than not using channel 1 altogether.

- **Adjacent channel separation**: The device's DCA capabilities are also useful in minimizing adjacent channel interference between APs.

  For example, two overlapping channels in the 802.11b/g band, such as 1 and 2, cannot simultaneously use 11 or 54 Mbps. By effectively reassigning channels, the device keeps adjacent channels separated.

Channel assignments

The device examines a variety of real-time RF characteristics to efficiently handle channel assignments.

- AP received energy: The received signal strength measured between each AP and its nearby neighboring AP. Channels are optimized to give you the highest network capacity.
- Noise: Noise can limit signal quality for your devices and APs. Increased noise reduces cell size and degrades user experience. By optimizing channels to avoid noise sources, the device helps you maintain coverage and system capacity. If a channel is unusable due to excessive noise, that channel can be avoided.
- 802.11 interference: Interference is any 802.11 traffic that is not a part of your wireless LAN, including rogue APs and neighboring wireless networks. Lightweight APs automatically scan all channels to detect interference sources. If the amount of 802.11 interference exceeds a predefined configurable threshold (the default is 10 percent), the AP sends an alert to the device. Using the RRM algorithms, the device may then dynamically rearrange channel assignments to increase system performance in the presence of the interference. Such an adjustment could result in adjacent lightweight APs being on the same channel, but this setup provides better performance than keeping APs on a channel made unusable by interference.

  In addition, if other wireless networks are present, the device shifts the usage of channels to complement the other networks. For example, if one network is on channel 6, an adjacent wireless LAN is assigned to channel 1 or 11. This arrangement increases the capacity of the network by limiting the sharing of frequencies. If a channel has virtually no capacity remaining, the device may choose to avoid this channel. In huge deployments in which all nonoverlapping channels are occupied, the device does its best, but you must consider RF density when setting expectations.

- Load and utilization: When utilization monitoring is enabled, capacity calculations can consider that some APs are deployed in ways that carry more traffic than other APs, for example, a lobby versus an engineering area. The device can then assign channels to improve the AP that has performed the worst. The load is taken into account when changing the channel structure to minimize the impact on the clients that are currently in the wireless LAN. This metric

keeps track of every AP's transmitted and received packet counts to determine how busy the APs are. New clients avoid an overloaded AP and associate to a new AP. This *Load and utilization* parameter is disabled by default.

The device combines this RF characteristic information with RRM algorithms to make system-wide decisions. Conflicting demands are resolved using soft-decision metrics that guarantee the best choice for minimizing network interference. The result is optimal channel configuration across three dimensions. APs located on different floors play an important role in your wireless LAN configuration.

RRM startup mode

The RRM startup mode is invoked under these conditions:

- In a single- device environment, the RRM startup mode is invoked after the device is upgraded and rebooted.
- In a multiple- device environment, the RRM startup mode is invoked after an RF Group leader is elected.
- You can trigger the RRM startup mode using the **ap dot11 {24ghz | 5ghz | 6ghz} rrm dca restart** command.

The RRM startup mode runs for 100 minutes (ten iterations at ten-minute intervals). The duration of the RRM startup mode is independent of the DCA interval, sensitivity, and network size. The startup mode consists of ten DCA runs with high sensitivity (making channel changes easy and sensitive to the environment) to converge to a steady-state channel plan. DCA continues to run at the specified interval and sensitivity after the startup mode is finished.

Dynamic bandwidth selection

Provides a smooth transition for various configurations while upgrading from 11n to 11ac.

Dynamic bandwidth selection is a wireless network algorithm that

- maximizes network throughput by dynamically varying the channel width during operation
- fine-tunes channel assignments by monitoring channel and Base Station Subsystem (BSS) statistics, and
- adapts to changes in network conditions, such as transitions from 11n to 11ac, and variable client mixes.

During upgrades from 11n to 11ac, dynamic bandwidth selection enables a smooth transition between configurations by adjusting channel widths and orientation (such as choosing between 40 MHz and 80 MHz bandwidths) in response to fast-changing wireless statistics.

Limitations for DCA

DCA limitations include the following items.

- DCA supports only 20-MHz channels in the 2.4-GHz band.
- In a Dynamic Frequency Selection (DFS)-enabled AP environment, enable the UNII2 channels option under the DCA channel to allow 100-MHz separation for the dual 5-GHz radios.
- The calculation methods for interference and channel utilization differ between 9120 (and older) AP models and the newer 916x and 917x AP models. The 9120 and older APs estimate interference and channel utilization using Wi-Fi data packets received from non-neighbor Wi-Fi sources during scanning. In contrast, the 916x and 917x APs use the radio chip vendor’s hardware-level snapshots to determine channel activity, providing improved accuracy over the older models.
- Use only nonoverlapping channels (1, 6, 11, and others) for reassigning channels to minimize interference.
- Channel change does not require you to shut down the radio.
- The DCA algorithm interval is set to one hour. However, the DCA algorithm always runs at the default interval of 10 minutes. Channel allocation occurs at 10-minute intervals for the first 10 cycles, and channel changes occur every 10 minutes as determined by the DCA algorithm. After these cycles, the DCA algorithm returns to the configured interval.

- Invoking channel update will not result in any immediate changes until the next DCA interval is triggered.
- If DCA or Transmit Power Control (TPC) is turned off on the RF group member, and automatic settings are enabled on the RF group leader, the channel or transmit power on the member changes according to the algorithm run on the RF group leader.

Coverage hole detection and correction

Details the coverage hole detection and correction algorithm, including alerting administrators to insufficient wireless coverage, discriminating between correctable and uncorrectable holes, and adjusting AP transmit power to mitigate coverage issues.

A coverage hole detection and correction algorithm is a wireless LAN management mechanism that

- identifies areas with insufficient radio coverage for reliable performance
- alerts administrators when access points fail to provide adequate coverage, and
- adjusts AP transmit power to mitigate correctable coverage holes.

If clients on a lightweight AP are detected at threshold levels such as RSSI, failed client count, percentage of failed packets, and number of failed packets that are lower than those specified in the RRM configuration, the AP sends a “coverage hole” alert to the device. The alert indicates that clients cannot connect to a usable AP because of poor signal coverage.

The device discriminates between coverage holes that can and cannot be corrected. For coverage holes that can be corrected, the device mitigates the coverage hole by increasing the transmit power level for that specific AP.

The device does not mitigate coverage holes caused by clients that are unable to increase their transmit power or are statically set to a power level. Increasing downstream transmit power could increase interference in the network.

Cisco AI enhanced RRM

The AI Enhanced RRM is the next evolution of Cisco's Radio Resource Management (RRM).

A Cisco AI enhanced RRM is a wireless radio resource management technology that

- applies artificial intelligence and machine learning to optimize RF environments
- operates through distributed data collection from Cisco wireless controllers and cloud-based analytics, and
- automates and adapts RF parameter tuning for Cisco wireless networks.

The RRM runs as a service in a Cisco Catalyst 9800 Series Wireless Controller. The Cisco RRM manages the RF Group (the components making up the RF Network) based on dynamic measurements between every AP and its neighbors stored in a local database for the entire RF Group. At runtime, the RRM draws the last 10 minutes of the collected data, and gently optimizes based on the current network conditions.

The AI Enhanced RRM integrates the power of Artificial Intelligence and Machine Learning to the reliable and trusted Cisco RRM product family algorithms in the Cloud.

Note

The AI enhanced RRM is coordinated through the Cisco Catalyst Center (on-prem appliance) as a service. The current RRM sites are seamlessly transitioned to an intelligent centralized service. AI enhanced RRM along with other Cisco Catalyst Center services brings a host of new features with it.

Cisco AI Enhanced RRM operates as a distributed RRM service. RF telemetry is collected from the Cisco Access Points by the controller, and passed through the Catalyst Center to the Cisco AI Analytics Cloud where the data is stored. The RRM Algorithms run against this telemetry data stored in the cloud. AI analyzes the solutions, and passes any configuration change information back to the Catalyst Center . The Catalyst Center maintains the control connection with the enrolled controller and passes any individual AP configuration changes back to the APs.

Note

The RRM algorithms run in the cloud against the telemetry data available in the cloud.

The following RRM algorithms run in the cloud while the remaining work in the controller:

- DCA
- TPC
- DBS
- FRA

Cisco AI enhanced RRM supporting releases

The table covers the controller and Cisco Catalyst Center release versions that support Cisco AI Enhanced RRM support:

Table 62: Controller and Cisco Catalyst Center releases supporting Cisco AI enhanced RRM support

Controller release	Cisco Catalyst Center release	Cisco AI enhanced RRM support
Cisco IOS XE Cupertino 17.9.x	• Cisco Catalyst Center , Release 2.3.2 or Cisco Catalyst Center , Release 2.3.3 • Cisco Catalyst Center , Release 2.3.4	• 2.4GHz and 5GHz • 2.4GHz, 5GHz, and 6GHz
Cisco IOS XE Cupertino 17.8.x	• Cisco Catalyst Center , Release 2.3.2 Cisco Catalyst Center , Release 2.3.3 • Cisco Catalyst Center , Release 2.3.4	2.4GHz and 5GHz
Cisco IOS XE Cupertino 17.7.x	Cisco Catalyst Center , Release 2.3.2 or Cisco Catalyst Center , Release 2.3.3	2.4GHz and 5GHz

If the location of controller, and APs are provisioned previously, assigning a location enrolls the AI Enhanced RRM Services and the profile to be pushed to the controller. Thus, AI Enhanced RRM becomes the RF Group Leader for the subscribed controller.

For more information on the Cisco Catalyst Center, see Cisco Catalyst Center User Guide .

Restrictions

Restrictions for Radio Resource Management (RRM).

The restrictions for RRM are:

- The maximum number of APs supported in an RF group depends on the capacity of the controller platform acting as the RF group leader. For platform-specific RF-group limits, refer the C9800 Radio Resource Management Deployment Guide.
- If an AP tries to join the RF-group that already holds the maximum number of APs it can support, the device rejects the application and throws an error.
- Disabling all data rates for **default rf-profile** or **custom rf-profile** affects the ISSU upgrade and client join process after the software upgrade (ISSU or non-ISSU). To prevent this, you must enable at least one data rate (for example, **ap dot11 24 rate RATE_5_5M enable**) on the **default rf-profile** or **custom rf-profile**. We recommend that you enable the lowest data rate if efficiency is of prime concern.
- Keywords like **secure** are not permissible as RF group names.
- RRM grouping does not occur when an AP operates in a static channel that is not in the DCA channel list. The Neighbor Discovery Protocol (NDP) is sent only on DCA channels; therefore, when a radio operates on a non-DCA channel, it does not receive NDP on the channel.

How to configure RRM

Guides you through configuring neighbor discovery type, RF groups, static RF groups, transmit power control, 802.11 RRM parameters and 802.11 advanced RRM, using GUI steps and CLI commands for effective Radio Resource Management.

Configure neighbor discovery type (GUI)

Configure Neighbor Discovery Type using GUI.

Configure the Neighbor Discovery Type for your network devices by leveraging the GUI interface.

Use this task to set the Neighbor Discovery Type within the Radio Resource Management settings to enhance the network's ability to detect and manage neighbor devices efficiently.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **RRM** .
2. On the **Radio Resource Management** page, click either the **5 GHz Band** , **2.4 GHz Band** or the **6 GHz Band** tab.
3. In the **General** tab, under each section enter the corresponding field details:
   a) Under the **Profile Threshold For Traps** section, enter the:
      1. **Interference Percentage**: The foreign interference threshold range is from 0 to 100 %. The default is 10 %.
      2. **Clients**: The client threshold range is from 1 to 75 clients. The default is 12.
      3. **Noise**: The foreign noise threshold range is from -127 dBm to 0dBm. The default is -70 dBm.

4. **Utilization Percentage**: The RF utilization threshold between 0 and 100 %. The default is 80 %.
5. **Throughput**: The average rate of successful messages delivery over a communication channel. Value ranges from 1000 to 10000000 bps. The default value is 1000000.

b) Under the **Noise/Interference/Rogue/CleanAir/SI Monitoring Channels** section, choose these options:

1. **Channel List** from the drop-down list:
   - All Channels
   - Country Channels
   - DCA Channels
2. **RRM Neighbor Discover Type** from the drop-down list:
   - **Transparent**: Packets are sent as is.
   - **Protected**: Packets are protected.
3. **RRM Neighbor Discovery Mode**:
   - **AUTO**: If the NDP mode configured is AUTO, the controller selects On-Channel as the NDP mode. The default is set as AUTO.
   - **OFF-CHANNEL**: If the NDP mode configured is Off-Channel, the controller selects Off-Channel as the NDP mode.

c) Under the **Monitor** section, set these options:

- **Neighbor Packet Frequency (seconds)**: Frequency (in seconds) in which the Neighbor Discovery Packets are sent. The default is 180 seconds.
- **Reporting Interval (seconds)**: The default is 180 seconds. Each channel dwell has to be completed within 180 seconds.
- **Neighbor Timeout factor**: Value in seconds used to determine when to prune access points from the neighbor list that have timed out. The default is 20 seconds.

4. Click **Apply** to save your configuration.

Neighbor discovery for your network devices is complete.

Configure neighbor discovery type (CLI)

Configure Neighbor Discovery Type using commands.

Specify how neighbor discovery packets are handled on each radio band.

Procedure

---

1. Enter global configuration mode.

   Example

```
Device# configure terminal
```

2. Configure the neighbor discovery type for the desired radio band.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm ndp-type {protected |
   transparent}
   ```

   The NDP types are:

   - **protected**: Use **protected** to encrypt discovery packets.
   - **transparent**: Use **transparent** to send packets as is (default).

3. Return to privileged EXEC mode by ending the configuration mode.

   Example

   ```
   Device(config)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The neighbor discovery type for the specified band is set.

Configure RF groups

Configure RF Groups through either the GUI or the CLI.

RF groups are configured through either GUI or CLI. You can also configure RF groups through

> **Note**
>
> When the multiple-country feature is being used, all controllers intended to join the same RF group must be configured with the same set of countries, configured in the same order.

Configure RF group selection mode (GUI)

Configure RF Group Selection Mode using GUI.

Set the RF Group Selection Mode for a wireless controller to manage radio resource management (RRM) and group configuration.

Use this procedure when you need to designate or update the RF Group Selection Mode on your controller using the web interface.

Before you begin

Ensure you have administrator access to the wireless controller GUI.

To configure the RF Group Selection Mode, use the steps in this section:

Procedure

---

1. Choose **Configuration** > **Radio Configurations** > **RRM**.

2. On the **RRM** page, click the relevant band tab: either **6 GHz Band**, **5 GHz Band**, or **2.4 GHz Band**.
3. Click the **RF Grouping** tab.
4. Choose the appropriate **Group Mode** from these options:
   - Automatic: Sets the 802.11 RF group selection to automatic update mode.
   - Leader: Sets the 802.11 RF group selection to leader mode.
   - Off: Disables the 802.11 RF group selection.

   Note

   When AI Enhanced RRM is enabled on a controller and Cisco Catalyst Center is connected to a wireless network, Cisco Catalyst Center is assigned the group role as leader. Controllers managed by Cisco Catalyst Center with AI Enhanced RRM enabled are assigned the group role as remote members regardless of the group mode previously assigned. The **Group Role** field displays **Remote Member**, and the **Group leader** field displays the IP address of the Cisco Catalyst Center.

5. Save the configuration.

The controller applies the selected RF Group Selection Mode to the chosen bands.

Configure RF group selection mode (CLI)

Configure RF Group Selection Mode using commands.

Set the RF group selection mode for specific 802.11 radio bands on a Cisco device using CLI commands.

Use this procedure to configure how access points are grouped for RF management purposes by specifying the RF group selection mode for 2.4 GHz, 5 GHz, or (if supported) 6 GHz bands.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the RF group selection mode for your 802.11 band.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} rrm group-mode {auto | leader | off}
   ```

   - **auto**: Sets the 802.11 RF group selection to automatic update mode.
   - **leader**: Sets the 802.11 RF group selection to leader mode.
   - **off**: Disables the 802.11 RF group selection.

3. Configure RF group selection mode for 802.11 bands.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm group-mode {auto | leader
 | off}
```

- **auto**: Sets the 802.11 RF group selection to automatic update mode.
- **leader**: Sets the 802.11 RF group selection to leader mode.
- **off**: Disables the 802.11 RF group selection.

4. Exit configuration mode to return to privileged EXEC mode.

Example

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The system applies the selected RF group selection mode to the specified 802.11 band.

Configure an RF group name (CLI)

Configure an RF Group Name using commands.

Set up a radio frequency (RF) group for wireless controllers. This action optimizes radio resource management.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Create an RF group. Enter a group name in ASCII format, up to 19 characters long. The name is case sensitive.

Example

```
Device(config)# wireless rf-network network-name
```

> **Note**
>
> Repeat this procedure for each controller you want to add to the RF group.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

You can also press **Ctrl-Z** to exit global configuration mode.

The RF group is created. Controllers in the group share RF information to optimize radio performance.

```
Device# configure terminal
Device(config)# wireless rf-network network1
Device(config)# end
Device# show network profile 100
```

Configure a secure RF group (CLI)

Configure a Secure RF Group using commands.

Set up a secure RF group for optimal wireless device communication and management.

With a secure RF group, your access points coordinate radio resources to improve network performance.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a secure RF group and enable the secure RF group functionality.

   Example

   ```
   Device(config)# wireless rf-network secure
   ```

3. Exit to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

4. Optional: Verify the RF group configuration and statistics for the desired frequency band , such as 2.4 GHz, 5 GHz, or 6 GHz.

   Example

   ```
   Device# show ap dot11 {24ghz | 5ghz | 6ghz} group
   ```

The secure RF group is created and operational on the selected wireless controller.

Configure members in an 802.11 static RF group (GUI)

Configure Members in an 802.11 Static RF Group using GUI.

Add controllers to a static RF group. This configuration enables coordinated radio management and optimizes wireless network performance.

When you configure static RF groups, you can manually designate RF group leaders and members. This approach supports predictable behavior and enhances control over radio resource management.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **RRM** .
2. On the **RRM** page, click either the **6 GHz Band** , **5 GHz Band**, or **2.4 GHz Band** tab.
3. Click the **RF Grouping** tab.
4. Choose the appropriate **Group Mode** from these options:
   - **Automatic (default)**: Members of an RF group elect a leader to maintain a primary power and channel scheme. The RF grouping algorithm selects the group leader dynamically and ensures continuous leadership. Group leader assignments can change. For example, assignments change if the current leader becomes inoperable or if group members experience significant changes.
   - **Leader**: A device as an RF group leader, manually. In this mode, the leader and members are manually configured. Their assignments remain fixed. If a member is unable to join the RF group, the system indicates the reason. The system uses the member's management IP address and system name to request that the member join the leader. The leader attempts to connect with a member every minute if the member did not join during the previous attempt.
   - **Off**: No RF group is configured.
5. Under **Group Members** section, click **Add**.
6. In the **Add Static Member** window that is displayed, enter the controller name and the IPv4 or IPv6 address of the controller.
7. Click **Save & Apply to Device**.

The new member is added to the static RF group and joins the manually configured group for radio parameter coordination.

Configure members in an 802.11 static RF group (CLI)

Configure Members in an 802.11 Static RF Group using commands.

Define APs as members of a static RF group using the CLI, enabling coordinated radio management.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure members in an 802.11 static RF group.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm group-member group_name
   ip_addr
   ```

Set the group mode to leader to activate the group member.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

You can also press **Ctrl-Z** to exit global configuration mode.

The AP is added as a member of the 802.11 static RF group.

```
Device# configure terminal
Device(config)# ap dot11 24ghz rrm group-member Grpmem01 10.1.1.1
Device(config)# end
```

Configure transmit power control

-

Configure transmit power (GUI)

Configure Transmit Power using GUI.

Configure transmit power settings for wireless APs.

Procedure

---

1. Choose **Configuration** > **Radio Configurations** > **RRM** .
2. On the **6 GHz Band** , **5 GHz Band** , or **2.4 GHz Band** tab, click the **TPC** tab.
3. Choose one of these dynamic transmit power assignment modes:
   - **Automatic** (default): The transmit power is periodically updated for all APs that permit this operation.
   - **On Demand**: The transmit power is updated on demand. If you choose this option, you get to view the **Invoke Power Update Once** . Click **Invoke Power Update Once** to apply the RRM data successfully.
   - **Fixed**: No dynamic transmit power assignments occur and values are set to their global default.
4. Enter the maximum and minimum power level assignment on this radio. If you configure maximum transmit power, RRM does not allow any AP attached to the device to exceed this transmit power level. This applies whether the power is set by RRM TPC or by coverage hole detection.

   For example, if you configure a maximum transmit power of 11 dBm, then no AP would transmit above 11 dBm, unless the AP is configured manually. The range is from -10 dBm to 30 dBm.
5. In the **Power Threshold** field, enter the cutoff signal level used by RRM when determining whether to reduce an access point's power.

   The default value for this parameter varies depending on the TPC version you choose. For TPCv1, the default value is -70 dBm, and for TPCv2, the default value is -67 dBm. The default value can be changed when APs are transmitting at higher (or lower) than desired power levels. The range is from -80 to -50 dBm.

   Increasing this value (between -65 and -50 dBm) causes the APs to operate at higher transmit power rates. Decreasing the value has the opposite effect. In applications with a dense population of APs, it may be useful to decrease the threshold to -80 or -75 dBm in order to reduce the number of BSSIDs (APs) and beacons seen by the wireless clients. Some wireless clients might have difficulty processing a large number of BSSIDs or a high beacon rate and might exhibit problematic behavior with the default threshold.

6. Click **Apply** .

Transmit power settings are updated according to your configuration.

Configure Tx-power control threshold (CLI)

Configure the Tx-Power Control Threshold using commands.

Set the Tx-power control threshold to define the minimum received signal strength at which the device adjusts its transmit power.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the Tx-power control threshold used by RRM for auto power assignment.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm tpc-threshold
   threshold_value
   ```

   The range is from -80 dBm to -50 dBm.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   You can also press **Ctrl-Z** to exit global configuration mode.

The Tx-power control threshold is updated, enabling the device to adjust its transmit power according to the specified threshold.

```
Device# configure terminal
Device(config)# ap dot11 24ghz rrm tpc-threshold -60
Device(config)# end
```

Configure the Tx-power level (CLI)

Configure the Tx-Power Level using commands.

Set the transmit power level of the wireless AP to improve wireless coverage and signal strength.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the 802.11 Tx-power level.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm txpower {trans_power_level | auto
 | max | min | once}
```

The Tx-power parameters are:

- **trans_power_level**: Sets the transmit power level.
- **auto**: Enables auto-RF.
- **max**: Configures the maximum auto-RF Tx-power.
- **min**: Configures the minimum auto-RF Tx-power.
- **once**: Enables one-time auto-RF.

3. Configure the 802.11 6-GHz Tx-power level.

Example

```
Device(config)# ap dot11 6ghz rrm txpower trans_power_level auto
```

The Tx-power parameters are:

- *trans_power_level*: Sets the transmit power level. The range is 1 to 5.
- **auto**: Enables auto-RF.

> **Note**
>
> The 6-GHz band uses constant-PSD instead of constant-EIRP, which allows you to transmit at higher power as the channel width increases. The system derives the power levels based on the configured channel width. At higher power levels between 1 and 3, these power values exceed the limit for legacy rate frames, such as beacon frames. The beacon power does not change at higher levels in the 6-GHz band, unlike in the 2.4-GHz and 5-GHz bands.

4. **end**

Example

```
Device(config)# end
```

Return to privileged EXEC mode.

Configure 802.11 RRM parameters

-

Configure advanced 802.11 channel assignment parameters (GUI)

Configure Advanced 802.11 Channel Assignment Parameters using GUI.

Configure the RRM settings to optimize channel assignment for 802.11 wireless networks.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **RRM** .
2. In the **DCA** tab, complete these details:
   a) Select a **Channel Assignment Mode** to specify the DCA mode:
      - **Automatic** (default): Causes the device to periodically evaluate and, if necessary, update the channel assignment for all joined APs.
      - **Freeze**: Causes the device to evaluate and update the channel assignment for all joined APs. If you choose this option, you get to view the Invoke Channel Update Once. Click **Invoke Channel Update Once** to apply the RRM data successfully.
      - **Off**: Turns off DCA and sets all AP radios to the first channel of the band, which is the default value. If you choose this option, you must manually assign channels on all radios.

   b) From the **Interval** drop-down list, choose the interval that tells how often the DCA algorithm is allowed to run. The default interval is 10 minutes.
   c) From the **Anchortime** drop-down list, choose a number to specify the time of day when the DCA algorithm must start. The options are numbers between 0 and 23 (inclusive) representing the hour of the day from 12:00 a.m. to 11:00 p.m.
   d) Check the **Avoid Foreign AP Interference** check box to cause the device's RRM algorithms to consider 802.11 traffic from foreign APs (those not included in your wireless network) when assigning channels to lightweight APs, or uncheck it to disable this feature. For example, RRM may adjust the channel assignment to have APs avoid channels close to foreign APs. By default, this feature is in enabled state.
   e) Check the **Avoid Cisco AP Load** check box to cause the device's RRM algorithms to consider 802.11 traffic from Cisco lightweight APs in your wireless network when assigning channels. For example, RRM can assign better reuse patterns to APs that carry a heavier traffic load. By default, this feature is in disabled state.
   f) Check the **Avoid Non-802.11a Noise** check box to cause the device's RRM algorithms to consider noise (non-802.11 traffic) in the channel when assigning channels to lightweight APs. For example, RRM may have APs avoid channels with significant interference from non-AP sources, such as microwave ovens. By default, this feature is in enabled state.
   g) Check the **Avoid Persistent Non-Wi-Fi Interference** check box to enable the device to take into account persistent non-Wi-Fi interference in DCA calculations. A persistent interfering device is any device from the following categories, which has been seen in the past 7 days - Microwave Oven, Video Camera, Canopy, WiMax Mobile, WiMax Fixed, Exalt Bridge. With **Avoid Persistent Non-Wi-Fi Interference** enabled, if a Microwave Oven is detected, that interference from the Microwave Oven is taken into account in the DCA calculations for the next 7 days. After 7 days, if the interfering device is not detected anymore, it is no longer considered in the DCA calculations.
   h) From the **DCA Channel Sensitivity** drop-down list, choose one of the following options to specify how sensitive the DCA algorithm is to environmental changes such as signal, load, noise, and interference when determining whether to change channels:
      - **Low**: The DCA algorithm is not particularly sensitive to environmental changes. The DCA threshold is 30 dB.
      - **Medium** (default): The DCA algorithm is moderately sensitive to environmental changes. The DCA threshold is 15 dB.

- **High**: The DCA algorithm is highly sensitive to environmental changes. The DCA threshold is 5 dB.

i) Set the **Channel Width** as required. You can choose the RF channel width as 20 MHz, 40 MHz, 80 MHz, 160 MHz, or Best. This is applicable only for 802.11a/n/ac (5 GHz) radio.

3. The **Auto-RF Channel List** section shows the channels that are currently selected. To select a channel, check the corresponding check box.

> **Note**
>
> If you disable the serving radio channel of the root AP from the **Auto-RF Channel List**, you will not be able to view the neighboring APs in the root APs.

4. In the **Event Driven RRM** section, check the **EDRRM** check box to run RRM when CleanAir-enabled AP detects a significant level of interference. If enabled, set the sensitivity threshold level at which the RRM is invoked, enter the custom threshold, and check the **Rogue Contribution** check box to enter the rogue duty-cycle.
5. Click **Apply** .

The wireless controller updates 802.11 channel assignments according to your configured parameters.

Configure 802.11 channel assignment parameters (CLI)

Configure Advanced 802.11 Channel Assignment Parameters using commands.

Configure DCA and related parameters on 802.11 radios.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure CleanAir event-driven RRM parameters.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel cleanair-event sensitivity
 {high | low | medium}
```

The types of sensitivity are:

- **High** : Specifies the most sensitivity to non-Wi-Fi interference as indicated by the air quality (AQ) value.
- **Low** : Specifies the least sensitivity to non-Wi-Fi interference as indicated by the AQ value.
- **Medium** : Specifies medium sensitivity to non-Wi-Fi interference as indicated by the AQ value.

3. Configure 802.11 6GHz DCA algorithm parameters.

Example

```
Device(config)# ap dot11 6ghz rrm channel dca {anchor-time <0-23> | global
auto | interval <0-24> sensitivity {high | low | medium}}
```

The DCA algorithm parameters include:

- **anchor-time**: Configures the anchor time for the DCA. The range is from 0 to 23 hours.
- **global**: Configures the DCA mode for all 802.11 Cisco APs.
  - **auto**: Enables auto-RF.
- **interval**: Configures the DCA interval value. The values are 1, 2, 3, 4, 6, 8, 12 and 24 hours, and the default value 0 denotes 10 minutes.
- **sensitivity**: Configures the DCA sensitivity level to changes in the environment.
  - **high**: Specifies the most sensitivity.
  - **low**: Specifies the least sensitivity.
  - **medium**: Specifies medium sensitivity.

4. Configure the DCA channel bandwidth for all 802.11 radios in the 5-GHz band.

   Example

   ```
   Device(config)# ap dot11 5ghz rrm channel dca chan-width {20 | 40 | 80 | best}
   ```

   The channel bandwidth can be set to 20 MHz, 40 MHz, or 80 MHz, . The default value for channel bandwidth is 20 MHz (80 MHz is the default value for Best). Set the channel bandwidth to Best before configuring the constraints.

5. Configure the maximum channel bandwidth that can be assigned to a channel.

   Example

   ```
   Device(config)# ap dot11 5ghz rrm channel dca chan-width width-max {WIDTH_20MHz
    | WIDTH_40MHz | WIDTH_80MHz |WIDTH_MAX}
   ```

   WIDTH_80MHz assigns the channel bandwidth to 20 MHz, 40 MHz, or 80 MHz but not greater than that.

6. Configure the maximum channel bandwidth that can be assigned to a channel.

   Example

   ```
   Device(config)# ap dot11 6ghz rrm channel
   dca chan-width width-max {WIDTH_200MHz | WIDTH_40MHz | WIDTH_80MHz | WIDTH_MAX}
   ```

   WIDTH_80MHz assigns the channel bandwidth to 20 MHz, 40 MHz, or 80 MHz but not greater than that.

The 802.11 channel assignment parameters are configured.

What to do next

Configure the advanced channel assignment parameters.

Configure the advanced channel assignment parameters (CLI)

Procedure

1. Configure the persistent non-Wi-Fi device avoidance in the 802.11 channel assignment.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} rrm channel device
   ```

2. Configure the foreign AP 802.11 interference avoidance in the channel assignment.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} rrm channel foreign
   ```

3. Configure the Cisco AP 802.11 load avoidance in the channel assignment.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} rrm channel load
   ```

4. Configure noise avoidance in 802.11 channel assignment.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} rrm channel noise
   ```

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   You can also press **Ctrl-Z** to exit global configuration mode.

The 802.11 advanced channel assignment parameters are configured.

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm channel noise
  Device(config)# end
```

Configure 802.11 coverage hole detection (GUI)

Configure 802.11 Coverage Hole Detection using GUI.

Enable 802.11 coverage hole detection to optimize wireless radio coverage and identify areas with inadequate connectivity.

Use this task when you want to detect and troubleshoot Wi-Fi coverage gaps on 6 GHz, 5 GHz, or 2.4 GHz radio bands in your wireless network.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **RRM** to configure Radio Resource Management parameters for 802.11ax (6 GHz), 802.11a/n/ac (5 GHz) and 802.11b/g/n (2.4 GHz) radios.
2. On the **Radio Resource Management** page, click the **Coverage** tab and complete these steps:
   a) To enable coverage hole detection, check the **Enable Coverage Hole Detection** check box.
   b) In the **Data Packet Count** field, enter the number of data packets.
   c) In the **Data Packet Percentage** field, enter the percentage of data packets.
   d) In the **Data RSSI Threshold** field, enter the actual value in dBm. The range is from -60 dBm to -90 dBm. The default value is -80 dBm.
   e) In the **Voice Packet Count** field, enter the number of voice data packets.
   f) In the **Voice Packet Percentage** field, enter the percentage of voice data packets.
   g) In the **Voice RSSI Threshold** field, enter the actual value in dBm. The range is from -60 dBm to -90 dBm. The default value is -80 dBm.
   h) In the **Minimum Failed Client per AP** field, enter the minimum number of clients on an AP with a signal-to-noise ratio (SNR) below the coverage threshold. The range is from 1 to 75, and the default value is 3.
   i) In the **Percent Coverage Exception Level per AP** field, enter the maximum desired percentage of clients on an access point's radio operating below the desired coverage threshold. The range is from 0 to 100%, and the default value is 25%.
3. Click **Apply**.

Your configuration is saved and the system begins monitoring for coverage holes based on the specified parameters.

Configure 802.11 coverage hole detection (CLI)

Configure 802.11 Coverage Hole Detection using commands.

Set up coverage hole detection for your wireless network.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the 802.11 6 GHz coverage hole detection for data packets.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm coverage data
   {fail-percentage | packet-count} value
   ```

   The attributes for the data packets are:

   - **fail-percentage**: Configures the 802.11 6-GHz coverage failure rate threshold for uplink data packets as a percentage that ranges from 1 to 100 percent.
   - **packet-count**: Configures the 802.11 6-GHz coverage minimum failure count threshold for uplink data packets that ranges from 1 to 255.

3. Configure the 802.11 AP coverage exception level as a percentage that ranges from 0 to 100 percent.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz} rrm coverage exception global
exception-level
```

4. Set the minimum exception level for AP clients.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} rrm coverage level global cli_min
   exception level
   ```

   The value range is from 1 to 75 clients.

5. Configure the 802.11 6 GHz coverage hole detection for voice packets.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm coverage voice
   {fail-percentage | packet-count} value
   ```

   The attributes for voice packets are:

   - **fail-percentage**: Configures the 802.11 6-GHz coverage failure rate threshold for uplink voice packets as a percentage that ranges from 1 to 100 percent.
   - **packet-count**: Configures the 802.11 6-GHz coverage minimum failure count threshold for uplink voice packets that ranges from 1 to 255.

6. Return to privileged EXEC mode to complete configuration.

   Example

   ```
   Device(config)# end
   ```

   ress **Ctrl-Z** to exit global configuration mode.

After you configure coverage hole detection thresholds, the wireless APs monitor your network and send alerts about coverage gaps.

```
Device# configure terminal
Device(config)# ap dot11 24ghz rrm coverage data fail-percentage 60
Device(config)# ap dot11 6ghz rrm coverage data fail-percentage 60
Device(config)# ap dot11 24ghz rrm coverage exception global 50
Device(config)# ap dot11 24ghz rrm coverage level global 10
Device(config)# ap dot11 24ghz rrm coverage voice packet-count 10
Device(config)# ap dot11 6ghz rrm coverage voice packet-count 10
Device(config)# end
```

Configure 802.11 event logging (CLI)

Configure 802.11 Event Logging using commands.

Enable and customize event logging for 802.11 wireless network parameters.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure event-logging for various parameters.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm logging {channel | coverage
    | foreign | load | noise | performance | txpower}
   ```

   The event-logging parameters include:

   - **channel**: Configures the 802.11 channel change logging mode.
   - **coverage**: Configures the 802.11 coverage profile logging mode.
   - **foreign**: Configures the 802.11 foreign interference profile logging mode.
   - **load**: Configures the 802.11 load profile logging mode.
   - **noise**: Configures the 802.11 noise profile logging mode.
   - **performance**: Configures the 802.11 performance profile logging mode.
   - **txpower**: Configures the 802.11 transmit power change logging mode.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   You can also press **Ctrl-Z** to exit global configuration mode.

You have enabled 802.11 event logging for the specified parameters.

```
Device# configure terminal

Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm logging {channel | coverage
 | foreign | load | noise | performance | txpower}
Device(config)# end
```

Configure 802.11 statistics monitoring (GUI)

Configure 802.11 Statistics Monitoring using GUI.

Enable and customize radio statistics monitoring intervals for APs.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **RRM** to configure Radio Resource Management parameters for 802.11ax (6 GHz), 802.11a/n/ac (5 GHz) and 802.11b/g/n (2.4 GHz) radios.
2. In the **Monitor Intervals (60 to 3600 secs)** section:
   a) To configure the 802.11 noise measurement interval (channel scan interval), set the **AP Noise Interval**. The valid range is from 60 to 3,600 seconds.
   b) To configure the 802.11 signal measurement interval (neighbor packet frequency), set the **AP Signal Strength Interval**. The valid range is from 60 to 3,600 seconds.
   c) To configure the 802.11 coverage measurement interval, set the **AP Coverage Interval**. The valid range is from 60 to 3,600 seconds.
   d) To configure the 802.11 load measurement, set the **AP Load Interval**. The valid range is from 60 to 3,600 seconds.
3. Click **Apply**.

802.11 statistics monitoring intervals are applied to the selected radios.

Configure 802.11 statistics monitoring (CLI)

Configure 802.11 Statistics Monitoring using commands.

Enable or customize the monitoring of 802.11 statistics on wireless APs.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Set the 802.11 monitoring channel-list for parameters such as noise/interference/rogue.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm monitor channel-list {all
    | country | dca}
   ```

   The channel list parameters include:

   - **all**: Monitors all channels.
   - **country**: Monitor channels used in configured country code.
   - **dca**: Monitor channels used by dynamic channel assignment.

3. Configure the 802.11 coverage measurement interval in seconds, which ranges from 60 to 3,600.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm monitor coverage interval
   ```

4. Configure the 802.11 load measurement interval in seconds, which ranges from 60 to 3,600.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm monitor load interval
```

5. Configure the 802.11 noise measurement interval (channel scan interval) in seconds, which ranges from 60 to 3,600.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm monitor noise interval
```

6. Configure the 802.11 signal measurement interval (neighbor packet frequency) in seconds, which ranges from 60 to 3,600.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm monitor signal interval
```

7. Configure the 802.11 RRM Neighbor Discovery RSSI normalization.

Example

```
Device(config)#ap dot11 {24ghz | 5ghz | 6ghz} rrm monitor rssi-normalization
```

The system applies your 802.11 statistics monitoring settings to the APs.

```
Device# configure terminal
Device(config)# ap dot11 6ghz rrm monitor channel-list all
Device(config)# ap dot11 6ghz rrm monitor coverage 600
Device(config)# ap dot11 6ghz rrm monitor load 180
Device(config)# ap dot11 6ghz rrm monitor noise 360
Device(config)# ap dot11 6ghz rrm monitor signal 480
Device(config)# ap dot11 6ghz rrm monitor rssi-normalization
```

Configure the 802.11 performance profile (GUI)

Configure the 802.11 Performance Profile using GUI.

Define radio frequency parameters to optimize AP performance in 802.11 networks.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. On the **AP Join** page, click the name of the profile or click **Add** to create a new one.
3. In the **Add/Edit RF Profile** window, click the **RRM** tab.
4. In the **General** tab that is displayed, enter the following parameters:
   a) In the **Interference (%)** field, enter the threshold value for 802.11 foreign interference, which ranges from 0 to 100 percent.
   b) In the **Clients** field, enter the threshold value for 802.11 Cisco AP clients, which ranges from 1 to 75 clients.
   c) In the **Noise (dBm)** field, enter the threshold value for 802.11 foreign noise, which ranges from -127 dBm to 0 dBm.

d) In the **Utilization(%)** field, enter the threshold value for 802.11 RF utilization, which ranges from 0 to 100 percent.

5. Click **Update & Apply to Device**.

The 802.11 performance profile is applied to the selected APs.

Configure the 802.11 performance profile (CLI)

This topic provides information on configuring performance profile.

Configure threshold values for 802.11 performance parameters, including clients, interference, noise, throughput, and RF utilization, on Cisco APs.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Set the threshold value for 802.11 Cisco AP clients, which ranges from 1 to 75 clients.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} rrm profile clients cli_threshold_value
   ```

3. Set the threshold value for 802.11 foreign interference, which ranges from 0 to 100 percent.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} rrm profile foreign int_threshold_value
   ```

4. Set the threshold value for 802.11 foreign noise, which ranges from -127 to 0 dBm.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} rrm profile noise
   for_noise_threshold_value
   ```

5. Enable performance profiles.

   Example

   ```
   Device(config)# ap dot11 6ghz rrm profile customize
   ```

6. Set the threshold value for 802.11 Cisco AP throughput, which ranges from 1000 to 10000000 bytes per second.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm profile throughput
   throughput_threshold_value
   ```

7. Set the threshold value for 802.11 RF utilization, which ranges from 0 to 100 percent.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} rrm profile utilization
   rf_util_threshold_value
   ```

8. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The AP's performance profile is updated with your specified threshold values, enabling automated monitoring and adjustment of wireless network conditions according to configured criteria.

```
Device# configure terminal
Device(config)# ap dot11 24ghz rrm profile clients 20
Device(config)# ap dot11 24ghz rrm profile foreign 50
Device(config)# ap dot11 24ghz rrm profile noise -65

Device(config)# ap dot11 6ghz rrm profile customize
Device(config)# ap dot11 24ghz rrm profile throughput 10000
Device(config)# ap dot11 24ghz rrm profile utilization 75
Device(config)# end
```

Configure advanced 802.11 RRM

-

Enable channel assignment (GUI)

Enable Channel Assignment using GUI.

To optimize wireless performance, assign device radio channels either automatically or manually.

Procedure

---

1. Choose **Configuration** > **Radio Configurations** > **RRM**.
2. In the **RRM** page, click the relevant band's tab: either **6 GHz Band**, **5 GHz Band** or **2.4 GHz Band**.
3. Click the **DCA** tab.
4. In the **Dynamic Channel Assignment Algorithm** section, choose the appropriate **Channel Assignment Mode**:
   - **Automatic**: Sets the channel assignment to automatic.
   - **Freeze**: Locks the channel assignment. Click **Invoke Channel Update Once** to refresh the assigned channels.
5. Click **Apply**.

Dynamic channel assignment is enabled based on your selection. You can configure radios to use automatic or manual channel allocation.

Enable channel assignment (CLI)

Enable Channel Assignment using commands.

To optimize radio resource allocation, assign wireless channels to 802.11 APs.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Enable the 802.11 channel selection update for each AP.

   Example

   ```
   Device# ap dot11 {24ghz | 5ghz | 6ghz} rrm channel-update
   ```

   Note

   After enabling the feature, the DCA algorithm assigns a token for channel assignment.

The system applies the DCA algorithm to update wireless channel assignments on all APs for the specified frequency.

```
Device# enable
Device# ap dot11 24ghz rrm channel-update
```

Restart DCA operation

Restart DCA Operation using commands.

To restore optimal channel allocation for wireless radios, restart the Dynamic Channel Assignment (DCA).

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Restart the DCA cycle for the 802.11 radio.

Example

```
Device# ap dot11 {24ghz | 5ghz  | 6ghz} rrm dca restart
```

The DCA process restarts on the wireless device.

```
Device# enable
Device# ap dot11 24ghz rrm dca restart
```

Update power assignment parameters (GUI)

Update Power Assignment Parameters using GUI.

Change the transmit power assignment settings for your AP to optimize wireless coverage.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. On the **Access Points** page, click the name of the AP from the 5 GHz or 2.4 GHz list.
3. In the **Edit Radios** > **Configure** > **Tx Power Level Assignment** section, select **Custom** from the **Assignment Method** group-down list.
4. Select the value for **Transmit Power** from the drop-down list.
5. Click **Update & Apply to Device**.

The system applies the updated transmit power parameters to your selected AP.

Update power assignment parameters (CLI)

Update Power Assignment Parameters using commands.

Adjust the wireless transmit power settings for APs to optimize coverage and performance.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Update the 802.11 6 GHz transmit power for every AP.

   Example

   ```
   Device# ap dot11 {24ghz | 5ghz | 6ghz} rrm txpower update
   ```

The system updates the transmit power configuration for the specified APs.

```
Device# enable
Device# ap dot11 24ghz rrm txpower update
Device# ap dot11 6ghz rrm txpower update
```

Monitor RRM parameters and RF group status

Monitor Radio Resource Management (RRM) parameters and RF group status using **show** commands to optimize wireless network performance.

Monitor RRM parameters

Monitor RRM Parameters using commands.

Provide a quick reference to the commands used for monitoring Radio Resource Management (RRM) parameters, enabling users to efficiently assess and troubleshoot wireless network performance.

Table 63: Commands for monitoring Radio Resource Management

Commands	Description
show ap dot11 24ghz channel	Displays the configuration and statistics of the 802.11b monitoring.
show ap dot11 24ghz coverage	Displays the configuration and statistics of the 802.11b coverage.
show ap dot11 24ghz group	Displays the configuration and statistics of the 802.11b grouping.
show ap dot11 24ghz logging	Displays the configuration and statistics of the 802.11b event logging.
show ap dot11 24ghz monitor	Displays the configuration and statistics of the 802.11b monitoring. nnn
show ap dot11 24ghz profile	Displays 802.11b profiling information for all APs.
show ap dot11 24ghz summary	Displays the configuration and statistics of the 802.11a APs.
show ap dot11 24ghz txpower	Displays the configuration and statistics of the 802.11b transmit power control.
show ap dot11 5ghz channel	Displays the configuration and statistics of the 802.11a channel assignment.
show ap dot11 5ghz coverage	Displays the configuration and statistics of the 802.11a coverage.
show ap dot11 5ghz group	Displays the configuration and statistics of the 802.11a grouping.
show ap dot11 5ghz logging	Displays the configuration and statistics of the 802.11a event logging.
show ap dot11 5ghz monitor	Displays the configuration and statistics of the 802.11a monitoring.

Commands	Description
show ap dot11 5ghz profile	Displays 802.11a profiling information for all APs.
show ap dot11 5ghz summary	Displays the configuration and statistics of the 802.11a APs.
show ap dot11 5ghz txpower	Displays the configuration and statistics of the 802.11a transmit power control.

Verify RF group status

Verify RF Group Status using commands.

This section describes the new commands for RF group status.

These commands are used to verify RF group status on the .

This table lists the commands for verifying aggressive load balancing.

Table 64: Aggressive load balancing verification commands

Command	Purpose
show ap dot11 5ghz group	Displays the controller name that is the group leader for the 802.11a RF network.
show ap dot11 24ghz group	Displays the controller name that is the group leader for the 802.11b/g RF network.
show ap dot11 6ghz group	Displays the controller name that is the group leader for the 802.11 6-GHz RF network.

To display the controller as a remote member and part of the AI Enhanced RRM, use this command:

```
Device# show ap dot11 24ghz group

Radio RF Grouping

RF Group Name : Open-RRM
RF Protocol Version(MIN) : 100(30)
RF Packet Header Version : 2
802.11b Group Mode : AUTO
802.11b Group Role : Remote-Member
802.11b Group Update Interval : 600 seconds
802.11b Group Leader : 172.19.30.39 (172.19.30.39)
Secure-RRM : Disabled


RF Group Members

Controller name Controller IP Controller IPv6 DTLS status
------------------------------------------------------------------------------------------
evwlc-188         192.0.2.188     N/A
```

Event-Driven Radio Resource Management

ED-RRM details.

A Event-Driven Radio Resource Management (ED-RRM) feature is a radio frequency management solution that

- continuously monitors air quality metrics
- automatically triggers channel changes when interference exceeds a set threshold, and
- blocks affected channels for a specified duration to prevent immediate reselection.

Spontaneous interference is interference that appears suddenly on a network, perhaps jamming a channel or a range of channels completely. The Cisco CleanAir spectrum event-driven RRM feature allows you to set a threshold for air quality (AQ) that, if exceeded, triggers an immediate channel change for the affected AP. Once a channel change occurs due to event-driven RRM, the channel is blocked list for three hours to avoid selection.

Most RF management systems can avoid interference, but this information takes time to propagate through the system. Cisco CleanAir relies on AQ measurements to continuously evaluate the spectrum and can trigger a move within 30 seconds. For example, if an AP detects interference from a video camera, it can recover by changing channels within 30 seconds of the camera becoming active.

Configure ED-RRM on the controller (CLI)

Configure ED-RRM using commands.

Configure Event-Driven Radio Resource Management (ED-RRM) on the controller using CLI commands.

Trigger spectrum event-driven RRM to run when a Cisco CleanAir-enabled AP detects a significant level of interference by entering these commands.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure CleanAir driven RRM parameters for the 802.11 APs.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm channel cleanair-event
   ```

3. Configure CleanAir driven RRM sensitivity for the 802.11 APs.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm channel cleanair-event
   sensitivity {low | medium | high | custom}
   ```

   Default selection is Medium.

4. Trigger the ED-RRM event at the set threshold value.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm channel cleanair-event
custom-threshold custom-threshold-value
```

The custom threshold range is from 1 to 99.

5. Enable rogue contribution.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm channel cleanair-event
rogue-contribution
```

6. Configure threshold value for rogue contribution.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm channel cleanair-event
rogue-contribution duty-cycle thresholdvalue
```

The range is from 1 to 99, with 80 as the default.

7. Save your changes.

Example

```
Device# write memory
```

8. Optional: Verify the CleanAir configuration for the 802.11a/n/ac or 802.11b/g/n network.

Example

```
Device# show ap dot11 {24ghz | 5ghz | 6ghz}cleanair config
```

The output is displayed.

```
CleanAir Solution................................ : Enabled
Air Quality Settings:
Air Quality Reporting........................ : Enabled
Air Quality Reporting Period (min)........... : 15
Air Quality Alarms........................... : Disabled
Air Quality Alarm Threshold.................. : 10
Unclassified Interference.................... : Disabled
Unclassified Severity Threshold.............. : 35
Interference Device Settings:
Interference Device Reporting................ : Enabled
BLE Beacon.............................. : Enabled
Bluetooth Link........................... : Enabled
Microwave Oven........................... : Enabled
802.11 FH................................ : Enabled
Bluetooth Discovery...................... : Enabled
TDD Transmitter.......................... : Enabled
Jammer................................... : Enabled
Continuous Transmitter................... : Enabled
DECT-like Phone.......................... : Enabled
Video Camera............................. : Enabled
```

```
802.15.4.............................. : Enabled
WiFi Inverted......................... : Enabled
WiFi Invalid Channel.................. : Enabled
SuperAG............................... : Enabled
Canopy................................ : Enabled
Microsoft Device...................... : Enabled
WiMax Mobile.......................... : Enabled
WiMax Fixed........................... : Enabled
Interference Device Types Triggering Alarms:
BLE Beacon............................ : Disabled
Bluetooth Link........................ : Disabled
Microwave Oven........................ : Disabled
802.11 FH............................. : Disabled
Bluetooth Discovery................... : Disabled
TDD Transmitter....................... : Disabled
Jammer................................ : Disabled
Continuous Transmitter................ : Disabled
DECT-like Phone....................... : Disabled
Video Camera.......................... : Disabled
802.15.4.............................. : Disabled
WiFi Inverted......................... : Enabled
WiFi Invalid Channel.................. : Enabled
SuperAG............................... : Disabled
Canopy................................ : Disabled
Microsoft Device...................... : Disabled
WiMax Mobile.......................... : Disabled
WiMax Fixed........................... : Disabled
Interference Device Alarms............ : Disabled
AdditionalClean Air Settings:
CleanAir Event-driven RRM State....... : Disabled
CleanAir Driven RRM Sensitivity....... : LOW
CleanAir Driven RRM Sensitivity Level. : 35
CleanAir Event-driven RRM Rogue Option. : Disabled
CleanAir Event-driven RRM Rogue Duty Cycle... : 80
CleanAir Persistent Devices state..... : Disabled
CleanAir Persistent Device Propagation. : Disabled
```

The Event-Driven Radio Resource Management (ED-RRM) on the controller is configured.

Rogue detection - rogue channel width

Rogue detection - rogue channel width details.

A rogue detection configuration is a security measure that

- allows specifying channel width and band for detecting unauthorized APs, and
- filters rogue APs based on matching channel width criteria and band.

The **condition chan-width** command is introduced in Cisco IOS XE Dublin 17.12.1 allows you to set the minimum or maximum channel width for rogue detection.

Configure rogue channel width (CLI)

Configure rogue channel width using commands.

Complete this task to configure rogue channel width.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create or enable a rule.

   Example

   ```
   Device(config)# wireless wps rogue rule rule-name priority priority
   ```

3. Configure channel width and band for rogue detection.

   Example

   ```
   Device(config-rule)# condition chan-width {160MHz | 20MHz | 40MHz | 80MHz}
   band {24ghz | 5ghz | 6ghz}
   ```

   If the classification is **Friendly**, this is the minimum channel width.

   If the classification is **Custom**, **Malicious**, or **Delete**, this is the maximum channel width.

4. Use Step 4, 5, 6, or 7.

   > **Note**
   >
   > Use only one of the Steps: 4, 5, 6, or 7 as required to classify rogue devices. Do not use all of them.

5. Optional: Classify devices matching this rule as friendly.

   Example

   ```
   Device(config-rule)# classify friendly state {alert | external | internal}
   ```

   The options are:

   - **alert**: Sets the malicious rogue access point to alert mode.
   - **external**: Acknowledges the presence of a rogue access point.
   - **internal**: Trusts a foreign access point.

6. Optional: Classify devices matching this rule as malicious.

   Example

   ```
   Device(config-rule)# classify malicious state {alert | contained}
   ```

   The options are:

   - **alert**: Sets the malicious rogue AP to alert mode.

- **contained**: Contains the rogue AP.

7. Optional: Classify devices matching this rule as custom.

   Example

   ```
   Device(config-rule)# classify custom severity-score severity-score [name name]
    state {alert | contained}
   ```

   Here the options are:

   - *severity-score* : Custom classification severity score. The range is from 1 to 100.
   - **name**: Defines the name for custom classification.
   - *name* : Specifies the custom classification name.
   - **state**: Defines the final state if rule is matched.
   - **alert**: Sets the rogue AP to alert mode.
   - **contained**: Contains the rogue AP.

8. Ignore the devices matching this rule.

   Example

   ```
   Device(config-rule)# classify delete
   ```

9. Return to privileged EXEC mode.

   Example

   ```
   Device(config-rule)# end
   ```

The rogue channel width is configured.

```
Device# configure terminal
                Device(config)# wireless wps rogue rule 1 priority 1
                Device(config-rule)# condition chan-width 20MHz band 5ghz
                Device(config-rule)# classify friendly state internal
                Device(config-rule)# classify malicious state alert
                Device(config-rule)# classify custom severity-score 12 name
rule1 state alert
                Device(config-rule)# classify delete
                Device(config-rule)# end
```

Configure rogue classification rules (GUI)

Configure rogue classification rules using GUI.

Complete this task to configure rogue classification rules.

Procedure

1. Choose **Configuration** > **Security** > **Wireless Protection Policies** > **Rogue AP Rules** to open the **Rogue Rules** window.

   Rules that have already been created are listed in priority order. The name, type, status, state, match, and hit count of each rule is provided.

   > **Note**
   >
   > To delete a rule, select the rule and click **Delete**.

2. Create a new rule with these steps:
   a) Click **Add**.
   b) In the **Add Rogue AP Rule** window, enter a name for the new rule, in the **Rule Name** field. Ensure that the name does not contain any spaces.
   c) From the **Rule Type** drop-down list, choose one of these options to classify rogue APs matching this rule:
      - **Friendly**
      - **Malicious**
      - **Unclassified**
      - **Custom**

   d) From the **State** drop-down list, configure the state of the rogue AP. This is the state when the rule matches the conditions for the rogue APs.
      - **Alert**: A trap is generated when an ad hoc rogue is detected.
      - **Internal**: A foreign ad hoc rogue is trusted.
      - **External**: The presence of an ad hoc rogue is acknowledged.
      - **Contain**: The ad hoc rogue is contained.
      - **Delete**: The ad hoc rogue is removed.

      > **Note**
      >
      > The **State** field is not displayed if you select **Unclassified** as the **Rule Type**.

   e) If you chose the **Rule Type** as **Custom**, enter the **Severity Score** and the **Custom Name**.
   f) Click **Apply to Device** to add this rule to the list of existing rules, or click **Cancel** to discard this new rule.
3. Optional: Edit a rule using these steps:
   a) Click the name of the rule that you want to edit.
   b) In the **Edit Rogue AP Rule** page that is displayed, from the **Type** drop-down list, choose one of these options to classify rogue APs matching this rule:
      - **Friendly**

- **Malicious**
- **Custom**

c) Configure the notification from the **Notify** drop-down list to **All**, **Global**, **Local**, or **None** after the rule is matched.
d) Configure the state of the rogue AP from the **State** drop-down list after the rule is matched.
e) From the **Match Operation** field, choose one of these options:

- **Match All**: The detected rogue AP must meet all of the conditions specified by the rule for the rule to be matched and the rogue AP to adopt the classification type of the rule.
- **Match Any**: The detected rogue AP must meet any of the conditions specified by the rule for the rule to be matched and the rogue AP to adopt the classification type of the rule. This is the default value.

f) To enable this rule, check the **Enable Rule** check box. The default is unchecked.
g) If you chose the **Rule Type** as **Custom**, enter the **Severity Score** and the **Classification Name**.
h) From the **Add Condition** drop-down list, choose one or more of the conditions that the rogue AP must meet:

- **None**: No condition is set for rogue AP detection.
- **client-count**: Condition requires that a minimum number of clients be associated to the rogue AP. For example, if the number of clients associated to the rogue AP is greater than or equal to the configured value, then the AP can be classified as malicious. If you choose this option, enter the minimum number of clients to be associated with the rogue AP in the **Minimum Number of Rogue Clients** field. The valid range is 1 to 10 (inclusive), and the default value is 0.
- **duration**: Condition requires that the rogue AP be detected for a minimum period of time. If you choose this option, enter a value for the minimum detection period in the **Time Duration** field. The valid range is 0 to 86400 seconds (inclusive), and the default value is 0 seconds.
- **encryption**: Condition requires that the advertised WLAN have specified encryption. Requires that the rogue AP's advertised WLAN does not have encryption enabled. If a rogue AP has encryption disabled, it is likely that more clients will try to associate with it. No further configuration is required for this option.
- **infrastructure**: Condition requires that the rogue AP's SSID (the SSID configured for the WLAN) be known to the controller. Select the **Manage SSID** check box to enable this configuration.
- **rssi**: Condition requires that the rogue AP have a minimum received signal strength indication (RSSI) value. For example, if the rogue AP has an RSSI that is greater than the configured value, then the AP could be classified as malicious. If you choose this option, enter the minimum RSSI value in the **Maximum RSSI** field. The valid range is 0 to -128 dBm (inclusive).
- **channel-width**: Condition requires that the rogue AP use the specified radio spectrum channel width for the specified radio band, as defined. The valid channel widths are 20, 40, 80, and 160MHz.
  - For APs to be classified as **Malicious**, **Custom** or **Delete**, it must match the value (equal or more) set in the **Minimum Channel Width** drop-down list.
  - For APs to be classified as **Friendly**, it must match the value (equal or less) set using an option from the **Maximum Channel Width** drop-down list.
- **ssid**: Condition requires that the rogue AP have a specific user-configured SSID. If you choose this option, enter the SSID in the **User Configured SSID** text field, and click **+** to add the SSID.
- **substring-ssid**: Condition requires that the rogue AP have a substring of the specific user-configured SSID. The controller searches the substring in the same occurrence pattern and returns a match if the substring is found in the SSID string.

4. Click **Apply to Device** to save the configuration.

5. Click **OK**.

The rogue classification rules are configured.

Verify rogue channel width

Verify rogue channel width details.

To view channel width and band information of a classification rule, use these commands.

> **Note**
>
> When the same BSSID is beaconing on multiple bands (2.4 GHz, 5 GHz, 6 GHz), the **show wireless wps rogue ap summary** command output displays information for the band with the highest RSSI.

```
Device# show wireless wps rogue rule detailed 1
Priority                                              : 1
Rule Name                                             : 1
Status                                                : Enabled
Type                                                  : Friendly
State                                                 : Alert
Match Operation                                       : Any
Notification                                          : Enabled
Hit Count                                             : 117
Condition :
  type                                                : chan-width
  Max value (MHz)                                     : 40
  Band (GHz)                                          : 5GHz
```

```
Device# wireless wps rogue ap summary
.
.
.

MAC Address     Classification  State  #APs  #Clients  Last Heard
Highest-RSSI-Det-AP  RSSI  Channel  Ch.Width  GHz
-------------------------------------------------------------------------------
002c.c849.9f00  Unclassified    Alert  2     0         10/18/2022 16:50:18
0cd0.f895.efc0       -31        11        20  2.4
0062.ecf3.e73f  Unclassified    Alert  1     0         10/18/2022 16:50:16
0cd0.f895.efc0       -46        36        80  5
4ca6.4d22.cbaf  Unclassified    Alert  3     0         10/18/2022 16:50:46
0cd0.f895.efc0       -62        36       160  5
```

48 Coverage Hole Detection

Topics:

- Coverage hole detection and correction

Coverage hole detection and correction

Details the coverage hole detection and correction algorithm, including alerting administrators to insufficient wireless coverage, discriminating between correctable and uncorrectable holes, and adjusting AP transmit power to mitigate coverage issues.

A coverage hole detection and correction algorithm is a wireless LAN management mechanism that

- identifies areas with insufficient radio coverage for reliable performance
- alerts administrators when access points fail to provide adequate coverage, and
- adjusts AP transmit power to mitigate correctable coverage holes.

If clients on a lightweight AP are detected at threshold levels such as RSSI, failed client count, percentage of failed packets, and number of failed packets that are lower than those specified in the RRM configuration, the AP sends a "coverage hole" alert to the device. The alert indicates that clients cannot connect to a usable AP because of poor signal coverage.

The device discriminates between coverage holes that can and cannot be corrected. For coverage holes that can be corrected, the device mitigates the coverage hole by increasing the transmit power level for that specific AP.

The device does not mitigate coverage holes caused by clients that are unable to increase their transmit power or are statically set to a power level. Increasing downstream transmit power could increase interference in the network.

Configure coverage hole detection (GUI)

Configure coverage hole detection using the GUI.

Enable Coverage hole detection (CHD) to configure client accounting using the GUI.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **RRM**.
2. Configure the Radio Resource Management parameters for 802.11a/n/ac (5 GHz) and 802.11b/g/n (2.4 GHz) radios, and flexible radio assignment parameters.
3. Check the **Enable Coverage Hole Detection** check box to activate the feature.

CHD is enabled. The system identifies and reports wireless coverage gaps.

Configure coverage hole detection (CLI)

Configure coverage hole detection using commands.

CHD is based on upstream RSSI metrics observed by the AP. Enable CHD on your wireless device using CLI commands.

> **Note**
>
> To revert back radios from 5-GHz to 24-GHz for CHD, ensure that the 5-GHz radio is UP and client network preference value is other than the default.

Before you begin

Disable the 802.11 network before applying the configuration.

Procedure

1. Configure the 802.11 coverage level for data packets.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm coverage
   ```

   The options are:

   - **24ghz**: Configure 802.11b parameters.
   - **5ghz**: Configure 802.11a parameters.
   - **6ghz**: Configure 802.11 6 GHz parameters.

2. Configure the 802.11 coverage level for data packets.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rrm coverage data
   {fail-percentage | packet-count | rssi-threshold} 60
   ```

   The options are:

   - **fail-percentage**: Configures 802.11b coverage failure rate threshold for uplink data packets.
   - **packet-count**: Configures 802.11b coverage minimum failure count threshold for uplinkvoice packets.
   - **rssi-threshold**: Configures 802.11b minimum receive coverage level for voice packets.

3. Configure the 802.11 6-GHz coverage hole detection for data packets.

   Example

   ```
   Device(config)# ap dot11 6ghz rrm coverage data {fail-percentage |
   packet-count} 60
   ```

   The options are:

   - **fail-percentage**: Configures 802.11b coverage failure rate threshold for uplink data packets.
   - **packet-count**: Configures 802.11b coverage minimum failure count threshold for uplinkvoice packets.

4. Configure the 802.11 Cisco AP coverage exception level as a percentage that ranges from zero to 100%.

Example

```
Device(config)# ap dot11 24ghz rrm coverage exception global 50
```

5. Configure the 802.11 Cisco AP client minimum exception level that ranges from one to 75 clients.

   Example

   ```
   Device(config)# ap dot11 24ghz rrm coverage level global 10
   ```

6. Configure the 802.11 coverage hole detection for voice packets.

   Example

   ```
   Device(config)# ap dot11 24ghz rrm coverage voice {fail-percentage |
   packet-count | rssi-threshold} 10
   ```

   The options are:

   - **fail-percentage**: Configures 802.11b coverage failure rate threshold for uplink data packets.
   - **packet-count**: Configures 802.11b coverage minimum failure count threshold for uplinkvoice packets.
   - **rssi-threshold**: Configures 802.11b minimum receive coverage level for voice packets.

7. Configure the 802.11 6-GHz coverage hole detection for voice packets.

   Example

   ```
   Device(config)# ap dot11 6ghz rrm coverage voice {fail-percentage |
   packet-count} 10
   ```

   The options are:

   - **fail-percentage**: Configures 802.11b coverage failure rate threshold for uplink data packets.
   - **packet-count**: Configures 802.11b coverage minimum failure count threshold for uplinkvoice packets.

8. Save the configuration and return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

9. Verify the CHD details.

   Example

   ```
   Device# show ap dot11 {24ghz | 5ghz | 6ghz} coverage
   ```

   This displays CHD status and statistics.

> **Note**
>
> If both the number and percentage of failed packets exceed the values entered in the **packet-count** and **fail-percentage** commands for a five-second period, the client enters a pre-alarm condition. The controller uses this information to distinguish between real and false coverage holes. False positives typically result from poor roaming logic implemented on most clients.
>
> A coverage hole is detected if both the number and percentage of failed clients meet or exceed the values entered in the coverage level global and coverage exception global commands over a 90-second period. The controller determines if the coverage hole can be corrected and, if appropriate, increases the transmit power level for that specific AP.

Configure CHD for RF tag profile (GUI)

Configure CHD for RF tag profile using GUI.

Enable and configure CHD parameters for an RF tag profile using the GUI to ensure optimal wireless coverage.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **RRM**.
2. On the **Coverage** tab, select the **Enable Coverage Hole Detection** check box.
3. In the **Data RSSI Threshold** field, enter the actual value in dBm. The valid value ranges from -60 dBm to -90 dBm; the default value is -80 dBm.
4. In the **Voice RSSI Threshold** field, enter the actual value in dBm. The valid value ranges from -60 dBm to -90 dBm; the default value is -80 dBm.
5. In the **Minimum Failed Client per AP** field, enter the minimum number of clients on an AP with a signal-to-noise ratio (SNR) below the coverage threshold. The valid value ranges from one to 75 and the default value is three.
6. In the **Percent Coverage Exception Level per AP** field, enter the maximum desired percentage of clients on an AP's radio operating below the desired coverage threshold and click Apply. The valid value ranges from zero to 100% and the default value is 25%.
7. In the **Voice Packet Count** and **Voice Packet Percentage** fields, enter the number and percentage of voice data packets.
8. In the **Data Packet Count** and **Data Packet Percentage** fields, enter the number and percentage of data packets.
9. Click **Apply**.

Configuring CHD for RF profile (CLI)

Configure CHD for RF profile using commands.

Enable CHD for a specified RF profile on your wireless device using CLI commands.

Before you begin
Ensure that the RF profile is already created.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the 802.11 coverage hole detection for data packets.

   Example

   ```
   Device(config)# ap dot11 24ghz rf-profile alpha-rfprofile-24ghz
   ```

3. Configure the minimum RSSI value for data packets received by the AP. The valid values range is from -90 to -60 in dBm.

   Example

   ```
   Device(config-rf-profile)# coverage data rssi threshold -80
   ```

4. Save the configuration and return to privileged EXEC mode.

   Example

   ```
   Device(config-rf-profile)# end
   ```

5. Verify the summary of the available RF profiles.

   Example

   ```
   Device# show ap dot11 24ghz rf-profile summary
   ```

49 Optimized Roaming

Topics:

- Optimized roaming

The Cisco Catalyst 9800 Series Wireless Controller Software Configuration Guide for Smart Roaming explains how to optimize client roaming by disassociating clients with low RSSI or data rates to prevent sticky clients and improve connection stability. It details configuration steps for enabling optimized roaming via GUI or CLI, including setting data rate thresholds and hysteresis values to enhance wireless client mobility and network performance.

Optimized roaming

Introduces optimized roaming technology, outlining its key functions and behavior, and describes deployment restrictions and use cases. Provides instructions for configuring optimized roaming using the GUI and CLI, covering both legacy and new workflows.

Optimized roaming is a Wi-Fi network feature that

- resolves the problem of sticky clients that remain associated to access points that are far away and outbound clients that attempt to connect to a Wi-Fi network without having a stable connection
- disassociates clients based on the RSSI of the client data packets and data rate when the RSSI alarm condition is met and the current data rate of the client is lower than the optimized roaming data rate threshold, and
- prevents client association when the client's RSSI is low by checking the RSSI of the incoming client against the RSSI threshold.

Additional optimized roaming information

You can disable the data rate option so that only RSSI is used for disassociating clients.

This feature checks the RSSI of the incoming client against the RSSI threshold, which is configured in the same place as the Coverage Hole Detection. This check prevents the clients from connecting to a Wi-Fi network unless the client has a viable connection. In many scenarios, even though clients can hear beacons and connect to a Wi-Fi network, the signal might not be strong enough to support a stable connection.

You can also configure the client coverage reporting interval for a radio by using optimized roaming. The client coverage statistics include data packet RSSIs, Coverage Hole Detection and Mitigation (CHDM) pre-alarm failures, retransmission requests, and current data rates.

Optimized roaming is useful in these scenarios:

- Addresses the sticky client challenge by proactively disconnecting clients.
- Actively monitors data RSSI packets.
- Disassociates client when the RSSI is lower than the set threshold.

Restrictions for optimized roaming

Lists optimized roaming restrictions.

Lists the restrictions that apply when using the optimized roaming feature on Cisco wireless controllers.

- When basic service set (*BSS*) transition is sent to 802.11v-capable clients, and if the clients are not transitioned to other BSS before the disconnect timer expires, the corresponding client is disconnected forcefully. BSS transition is enabled by default for 802.11v-capable clients.
- The Cisco Catalyst 9800 controller increments the 80211v smart roam failed counter while disconnecting the client due to optimized roaming.
- We recommend that you do not use the optimized roaming feature with RSSI low check.

Configure optimized roaming (GUI)

Configure Optimized Roaming using GUI to improve wireless client connectivity by managing disassociation thresholds and data rates.

Optimized roaming helps maintain optimal wireless client connectivity by managing when clients are disassociated based on RSSI and data rate thresholds.

Procedure

1. Choose **Configuration** > **Wireless** > **Advanced**.
2. Click the **Optimized Roaming** tab.
3. Check the **Optimized Roaming Mode** check box in the **6 GHz Band**, **5 GHz Band**, or, **2.4 GHz Band** sections to enable the feature.

   Note

   Modifying the default settings for the **Optimized Roaming data rate** and CHDM RSSI configurations could result in unintended client connectivity problems. Ensure that you cautiously make updates to the default settings.

4. Choose the required **Optimized Roaming Date Rate Threshold**. The threshold value options are different for 802.11a, 802.11b, and 802.11ax networks.

   Optimized roaming disassociates clients based on the RSSI of the client data packet and data rate. The client is disassociated if the current data rate of the client is lower than the Optimized Roaming Data Rate Threshold.

5. In the **Optimized Roaming Hysteresis** field, enter the hysteresis value. The the range is from 0 to 10. The default is 6.
6. Click **Apply** to save the configuration.

Optimized roaming is configured and will manage client disassociation based on the specified thresholds and settings.

Configure optimized roaming (CLI)

Configure optimized roaming using commands.

Enable optimized roaming to improve client connectivity and reduce roaming delays by automatically triggering roaming decisions based on data rate thresholds.

Optimized roaming helps wireless clients move seamlessly between access points when their connection quality degrades below specified thresholds. This feature is particularly useful in high-density environments where clients need to maintain optimal connectivity.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure 802.11a, 802.11b, or 802.11 6-GHz optimized roaming.

   Example

   ```
   Device(config)# ap dot11 band rrm optimized-roam
   ```

Example

```
Device(config)# ap dot11 24ghz rrm optimized-roam
```

By default, optimized roaming is disabled. The *band* can be 24-GHz, 5-GHz, or 6-GHz.

3. Configure the data rate threshold for 802.11b for optimized roaming.

Example

```
Device(config)# ap dot11 24ghz rrm monitor optimized-roam data-rate-threshold
 threshold-value
```

Example

```
Device(config)# ap dot11 24ghz rrm monitor optimized-roam data-rate-threshold
 18M
```

Valid threshold values for 24-GHz: 1M, 2M, 5_5M, 6M, 9M, 11M, 12M, 18M, 24M, 36M, 48M, 54M, or disable.

4. Configure the data rate threshold for 802.11a or 802.11 6-GHz optimized roaming.

Example

```
Device(config)# ap dot11 band rrm monitor optimized-roam data-rate-threshold
 threshold-value
```

Example

```
Device(config)# ap dot11 6ghz rrm monitor optimized-roam data-rate-threshold
 18M
```

Valid threshold values for 5-GHz and 6-GHz: 6M, 9M, 12M, 18M, 24M, 36M, 48M, 54M, or disable.

5. Display the 802.11a, 802.11b, or 802.11 6-GHz optimized roaming configurations.

Example

```
Device# show ap dot11 band optimized-roaming statistics
```

Example

```
Device# show ap dot11 24ghz optimized-roaming statistics
```

Optimized roaming is now configured with the specified data rate thresholds, enabling automatic client roaming when connection quality degrades below the configured values.

50 Cisco Flexible Radio Assignment

Topics:

- Flexible radio assignment actions

Flexible radio assignment actions

Describes FRA Action functionality, guiding users through configuration in default and 2.4-GHz RF profiles via CLI and outlining procedures to verify FRA Action deployments for optimal wireless performance.

Flexible Radio Assignment (FRA) evaluates 2.4-GHz radio coverage to identify overlapping coverage causing radio interference. If there is an overlapping coverage, the dual-band radio moves to either 5-GHz client serving or monitor role.

Feature history for flexible radio assignment action
Table 65: Feature history for FRA Action

Release	Feature	Feature information
Cisco IOS XE Dublin 17.10.1	Flexible Radio Assignment (FRA) Action	In Cisco IOS-XE 17.10.1 and earlier releases, the FRA moves the redundant dual-band radios to either 5-GHz client-serving role or monitor role.
Cisco IOS XE Dublin17.11.1	Flexible Radio Assignment (FRA) Action	From Cisco IOS-XE 17.11.1 onwards, you can select the redundant dual-band radios in a network to operate in monitor only mode.

Release Information

In Cisco IOS-XE 17.10.1 and earlier releases, the FRA moves the redundant dual-band radios to either 5-GHz client-serving role or monitor role. From Cisco IOS-XE 17.11.1 onwards, you can set redundant dual-band radios in a network to operate in monitor-only mode.

> **Note**
>
> The FRA action feature is disabled by default.

Configure FRA action in default RF profile (CLI)

Configure the FRA action in the default RF profile to optimize radio frequency management.

Procedure

1. Configure terminal

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. Configure the FRA action

Example

```
Device(config)# ap dot11 24ghz fra action monitor
```

Configures the FRA action to monitor mode, moving all redundant dual-band APs solely to the monitor role.

3. Exit configuration mode

Example

```
Device(config)# end
```

Exits configuration mode and returns to privileged EXEC mode.

Upon completion, the FRA action is set, redundancy in dual-band radios is handled efficiently, and the system returns to privileged EXEC mode.

Configure FRA action in 2.4-GHz RF profile (CLI)

Configure FRA action in 2.4-GHz RF profile using commands.

Procedure

---

1. Configure terminal

Example

```
Device# configure terminal
```

Enters global configuration mode

2. Configure the RF profile name

Example

```
Device(config)# ap dot11 24ghz rf-profile alpha-rfprofile-24ghz
```

Configures the RF profile name and enters RF profile configuration mode.

3. Configure the FRA action as monitor

Example

```
Device(config-rf-profile)# fra action monitor
```

Configures the FRA action as monitor, and moves all redundant dual-band radios to monitor role only.

4. Exit configuration mode

Example

```
Device(config-rf-profile)# end
```

Exits configuration mode and returns to privileged EXEC mode.

Verify FRA action configuration

To view the selected FRA action, use the following command:

```
Device# show ap fra
FRA State               : Enabled
FRA Freeze              : Disabled
FRA Operation State     : Up
FRA Sensitivity         : higher (85%)
FRA Interval            : 1 Hour(s)
Service Priority        : Coverage
Client Aware FRA        : Enabled
  Client Select         : 25%
  Client Reset          : 5%
FRA Action              : 2.4GHz/Monitor
  Last Run              : 3069 seconds ago
```

To view the FRA action details in an AP RF profile, use the following command:

```
Device# show ap rf-profile name madhu-rf-profile-24 detail | sec FRA
Client Aware FRA                  : Disabled
FRA Action                        : 2.4GHz/Monitor
```

To view the radio mode and role in an AP, use the following command:

```
Device# show ap name AP7872.5DED.CB74 config slot 0 | sec Attribute
Attributes for Slot 0
  Radio Type                                    : 802.11n - 2.4/5 GHz
  Radio Mode                                    : Monitor
  Radio Role                                    : Monitor
    Assignment Method                           : Auto
    Monitor Mode Reason                         : Automatically Switched by
FRA
```

51 XOR Radio Support

Topics:

- Dual-band radios in Cisco AP models

Dual-band radios in Cisco AP models

Details on dual-band radio support.

A dual-band radio is a device category that

- offers connectivity on more than one frequency band (such as 2.4 GHz and 5 GHz)
- provides flexibility in network configuration, and
- is used in multiple Cisco AP models like the 2800, 3800, 4800, and 9120 series.

Feature history table

Table 66: Feature history table for dual-band radios in Cisco AP models

Feature name	Release information	Feature description
Dual-band (XOR) radio support	Cisco IOS XE 16.10.1	The Dual-band (XOR) radio support offers the ability to serve 2.4 GHz or 5 GHz bands or passively monitor both the bands on the same AP. These APs can be configured to serve clients in 2.4 GHz and 5 GHz bands, or serially scan both 2.4 GHz and 5 GHz bands on the flexible radio while the main 5 GHz radio serves clients.

Key features of dual-band radios in Cisco APs

The XOR radio support can be steered manually or automatically.

- Manual steering of a band on a radio: The band on the XOR radio can only be changed manually.
- Automatic client and band steering on radios: This is managed by the Flexible Radio Assignment (FRA) feature that monitors and changes the band configurations as per site requirements.

Client steering

When a radio moves between bands (from 2.4 GHz to 5 GHz and vice versa), clients need to be steered to get an optimal distribution across radios. When an AP has two radios in the 5 GHz band, client steering algorithms contained in the FRA algorithms are used to steer a client between the same band co-resident radios

Cisco APs and dual-band radios

Cisco 2800, 3800, 4800, and 9120 series AP models are equipped with dual-band (XOR) radios. These models have

- radios that operate on either 2.4 GHz or 5 GHz bands, or
- the ability to passively monitor both the bands on the same AP.

These APs can be configured to serve clients in 2.4 GHz and 5 GHz bands, or serially scan both 2.4 GHz and 5 GHz bands on the flexible radio while the main 5 GHz radio serves clients.

Cisco AP models up to the Cisco 9120 APs are designed to support dual 5 GHz band operations with the *i* model supporting a dedicated Macro or Micro architecture and the *e* and *p* models supporting Macro or Macro. The Cisco 9130AXI APs support dual 5 GHz operations as Macro or Micro cell.The CW9166I supports Macro or Macro cell.

Cisco Catalyst Wireless 9166 AP (CW9166) has XOR function for a dual 5 GHz 4x4 or 5 GHz 4x4 and 6 GHz 4x4 radios. These radios can also be configured as client serving, monitor or as a sniffer interface like the earlier XOR radios.

> **Note**
>
> For all countries that do not support 6 GHz spectrum for use of Wi-Fi, when the Cisco Catalyst Wireless 9166I AP operates as dual 5 GHz, the 5 GHz channels will be locked on both the radios even if slot 2 is disabled or set up for monitoring.

Wi-Fi 7 APs compatibility

For Wi-Fi 7 APs, the XOR radio operates in 2.4 GHz or 5 GHz low band mode (UNII 1-2A). By default, the XOR radio is brought up in 2.4 GHz band. The existing controller dual-band commands are used to switch between 2.4 GHz and 5 GHz bands. When XOR radio is switched to 5 GHz band, the Wi-Fi 7 APs operate in dual 5 GHz mode. To provide enough RF isolation between both 5 GHz radios, both the 5 GHz radios operate in band-locked mode and radio frequency.

Limitations

Limitations for XOR radio support.

These are the limitations for XOR radio support:

- RF measurement is disabled when a static channel is configured on slot 1. As a result, the dual-band radio slot 0 operates only with 5 GHz radios and not in the monitor mode. When slot 1 radio is disabled, RF measurement will not run, and the dual band radio slot 0 will be only on 2.4 GHz radio.
- Only one of the 5 GHz radios can operate in the UNII band (100 to 144), due to an AP limitation to maintain the power budget within the regulatory limit.

Configure default XOR radio support (CLI)

Configure Default XOR Radio Support using commands.

Complete this task to configure the basic settings for XOR radio support.

Before you begin

> **Note**
>
> The default radio points to the XOR radio hosted on slot 0.

Procedure

1. Enable the priviledged EXEC mode.

   Example

   ```
   Device> enable
   ```

2. Configure the 802.11 dual-band external antenna gain on the AP.

Example

```
Device# ap name ap-name  dot11 dual-band antenna ext-ant-gain
antenna_gain_value
```

Enter external antenna gain value in multiple of .5 dBi units (that is, an integer value 4 means 4 x 0.5 = 2 dBi of gain).

3. Shut down the default dual-band radio.

Example

```
Device# ap name ap-name  dot11 dual-band shutdown
```

Use the **no** form of the command to enable the radio.

4. Switch to the client-serving mode of the AP.

Example

```
Device# ap name ap-name  dot11 dual-band role manual client-serving
```

5. Switch to the 2.4 GHz radio band.

Example

```
Device# ap name ap-name  dot11 dual-band band 24ghz
```

The basic configuration of the XOR radio is completed.

Example

```
Device> enable
Device# ap name ap-name  dot11 dual-band antenna ext-ant-gain 2
Device# ap name ap-name  dot11 dual-band shutdown
Device# Device# ap name ap-name  dot11 dual-band role manual client-serving
Device# Device# ap name ap-name  dot11 dual-band band 24ghz
```

What to do next

Continue with the configuration steps listed in the "Configure advanced settings" to complete the configuration.

Configure advanced settings (CLI)

Complete this task to configure the advanced settings for XOR radio support, such as, transmit power, channel, channel width, CleanAir settings and so on.

Procedure

---

1. Configure the transmit power for the radio on the AP.

Example

```
Device# ap name ap-name  dot11 dual-band txpower {transmit_power_level | auto}
```

> **Note**
>
> When an FRA-capable radio (slot 0 on 9120 AP[for instance]) is set to Auto, you cannot configure static channel and Txpower on this radio.
>
> If you want to configure static channel and Txpower on this radio, you will need to change the radio role to Manual Client-Serving mode.
>
> This note is not applicable for Cisco Catalyst Wireless 9166 AP (CW9166).

2. Configure the channel number for the dual band.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band channel 40
   ```

   The range of the *channel-number* is from 1 to 173.

3. Optional: Enable the auto channel assignment for the dual-band.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band channel auto
   ```

4. Configure the channel width for the dual band.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band channel width {20 MHz | 40 MHz | 80
    MHz | 160 MHz}
   ```

5. Enable the Cisco CleanAir management on the dual-band radio.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band cleanair
   ```

6. Configure a band for the Cisco CleanAir feature.

   Example

   ```
   Device# ap name ap-name [no] dot11 dual-band cleanair band 5 GHz
   ```

   Selects a band for the Cisco CleanAir feature.

   Use the **no** form of this command to disable the Cisco CleanAir feature.

7. Configure the 802.11n dual-band antenna and antenna ports selection for a specific AP.

   Example

   ```
   Device# ap name ap-name  dot11 dual-band dot11n antenna {A | B | C | D}
   ```

Here, A, B, C, and D are the antenna ports.

The configuration of the additional features for dual-band (XOR) radios are completed.

Example

```
Device# ap name ap-name  dot11 dual-band txpower auto
Device# ap name ap-name  dot11 dual-band channel 40
Device# ap name ap-name  dot11 dual-band channel width 40 MHz
Device# ap name ap-name  dot11 dual-band cleanair
Device# ap name ap-name [no] dot11 dual-band cleanair band 5 GHz
Device# ap name ap-name  dot11 dual-band dot11n antenna A
```

Configure XOR radio support for the specified slot number (GUI)

Configure XOR radio support for the specified slot number using GUI.

Complete this task to configure XOR radio for the specified slot number.

Procedure

---

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. In the **Dual-Band Radios** section, select the AP for which you want to configure dual-band radios.

   The AP name, MAC address, CleanAir capability and slot information for the AP are displayed. If the Hyperlocation method is HALO, it displays the antenna PID and antenna design specifics.
3. Click **Configure**.
4. In the **General** tab, set the **Admin Status** as required.
5. Set the **CleanAir Admin Status** field to Enable or Disable.
6. Click **Update & Apply to Device**.

The XOR radio support for the specified slot number has been configured.

Configure XOR radio support for the specified slot number (CLI)

Configure XOR Radio Support for the Specified Slot Number using commands.

Complete this task to configure XOR radio for a specified slot number.

Procedure

---

1. Enable the priviledged EXEC mode.

   Example

   ```
   Device> enable
   ```
2. Configure dual-band antenna for the XOR radio hosted on slot 0 for a specific AP.

Example

```
Device# ap name ap-name dot11 dual-band slot 0 antenna ext-ant-gain
external_antenna_gain_value
```

Here, *external_antenna_gain_value* is the external antenna gain value in multiples of .5 dBi unit. The range is from 0 to 40.

> **Note**
>
> - For APs supporting self-identifying antennas (SIA), the gain depends on the antenna, and not on the AP model. The gain is learned by the AP and there is no need for controller configuration.
> - For APs that do not support SIA, the APs send the antenna gain in the configuration payload, where the default antenna gain depends on the AP model.

3. Configure the current band for XOR radio hosted on slot 0 for a specific AP.

Example

```
Device# ap name ap-name dot11 dual-band slot 0 band {24ghz | 5ghz}
```

4. Configures dual-band channel for the XOR radio hosted on slot 0 for a specific AP.

Example

```
Device# ap name ap-name dot11 dual-band slot 0 channel {channel-number | auto
 | width [160 | 20 | 40 | 80]}
```

The range of the *channel_number* is from 1 to 165.

5. Enable the Cisco CleanAir feature for dual-band radios hosted on slot 0 for a specific AP.

Example

```
Device# ap name ap-name dot11 dual-band slot 0 cleanair band {24Ghz | 5ghz}
```

6. Configure 802.11n dual-band parameters hosted on slot 0 for a specific AP.

Example

```
Device# ap name ap-name dot11 dual-band slot 0 dot11n antenna {A | B | C |
D}
```

Here are the antenna configurations.

A : Enables antenna port A.

B : Enables antenna port B.

C : Enables antenna port C.

D : Enables antenna port D.

7. Configure dual-band role for the XOR radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name ap-name dot11 dual-band slot 0 role {auto | manual
   [client-serving | monitor]}
   ```

   Here are the dual-band roles.

   - **auto** : refers to the automatic radio role selection.
   - **manual** : refers to the manual radio role selection.

   Note

   When Flexible Radio Assignment (FRA) is disabled, XOR radios whose roles were changed by Radio Resource Management (RRM) might not automatically revert to their default radio role. To restore an individual AP to the default client-serving role, use the these commands:

   ```
   Device# ap name ap-name dot11 dual-band slot 0 shutdown
   ```

   ```
   Device# ap name ap-name dot11 dual-band slot 0 role manual client-serving
   ```

   ```
   Device# ap name ap-name dot11 dual-band slot 0 band 24ghz
   ```

   ```
   Device# ap name ap-name no dot11 dual-band slot 0 shutdown
   ```

8. Disable dual-band radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name ap-name [no] dot11 dual-band slot 0 shutdown
   ```

   Use the **no** form of this command to enable the dual-band radio.

9. Configure dual-band transmit power for XOR radio hosted on slot 0 for a specific AP.

   Example

   ```
   Device# ap name ap-name dot11 dual-band slot 0 txpower {tx-power-level | auto}
   ```

   Here,

   - *tx_power_level* : is the transmit power level in dBm. The range is from 1 to 8.

- **auto** : enables auto-RF.

The XOR radio for the specified slot number is configured.

Example

```
Device> enable
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 antenna
ext-ant-gain 2
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 band 24ghz

                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 channel 3
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 cleanair
band 24Ghz
                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 dot11n
antenna A
              Device# ap name AP-SIDD-A06 [no] dot11 dual-band slot 0 shutdown

                Device# ap name AP-SIDD-A06 dot11 dual-band slot 0 txpower 2
```

Verify XOR radio support

To verify the XOR radio support parameters, use these **show** commands:

Verify auto-RF information

To display the auto-RF information for a Cisco AP, use the **show ap name** *ap-name* **auto-rf dot11 dual-band** command.

```
Device# show ap name cisco-ap-name auto-rf dot11 dual-band
```

Verify the list of BSSIDs

To display the list of BSSIDs for a Cisco AP, use the **show ap name** *ap-name* **wlan dot11 dual-band** command.

52 Cisco Receiver Start of Packet

Topics:

- Receiver start of packet detection threshold

Receiver start of packet detection threshold

Explains what receiver start of packet detection thresholds are and how they impact wireless network performance and client distribution.

A receiver start of packet (Rx SOP) detection threshold is a wireless radio configuration setting that

- determines the Wi-Fi signal level (measured in dBm) at which an APs radio demodulates and decodes a received packet
- affects radio sensitivity and the effective size of the receiver cell, and
- influences how clients are distributed across APs within a wireless network.

Rx SOP is used to address clients with weak RF links, sticky clients, and to support client load balancing across APs. Rx SOP helps optimize network performance in high-density deployments, such as stadiums and auditoriums. In these environments, APs need to prioritize the nearest and strongest clients.

Restrictions for Rx SOP

Rx SOP Restrictions.

- Rx SOP configuration does not apply to the third radio module pluggable on Cisco Aironet 3600 Series APs.
- Rx SOP configurations are supported only in Local, FlexConnect, Bridge, and Flex+Bridge modes.
- Rx SOP configurations are not supported in the FlexConnect+PPPoE, FlexConnect+PPPoE-wIPS, and FlexConnect+OEAP submodes.

Permitted range for the Rx SOP threshold

This table shows the allowed Rx SOP (Receiver Start of Packet) threshold settings.

- It is for different radio bands and threshold levels.

Table 67: Rx SOP Threshold

Radio Band	Threshold High	Threshold Medium	Threshold Low
2.4 GHz	-79 dBm	-82 dBm	-85 dBm
5 GHz	-76 dBm	-78 dBm	-80 dBm

Known behavior

Regardless of the radio mode, the controller sets the radio with the specified RX-SOP value. The AP then decides whether to use this value.

- For the XOR radio (Slot 0), when the AP is in monitor mode, the RX-SOP value sent to the AP depends on the frequency band used before switching to monitor mode. If the radio operated in the 2.4 GHz band, the RX-SOP parameters are selected from the 2.4 GHz RF profile or the default RF profile. If it operated in the 5 GHz band, the RX-SOP parameters are chosen from the 5 GHz RF profile or the default RF profile configured for the AP.

Configure Rx SOP (CLI)

Configure Rx SOP using commands.

Adjust the Rx SOP threshold for high-density Wi-Fi environments.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   DeviceDevice# configure terminal
   ```

2. Configure the 802.11bg or 802.11a radio Rx SOP threshold.

   Example

   ```
   DeviceDevice(config)# ap dot11 {24ghz | 5ghz} rx-sop threshold {auto | custom
    | high | low | medium}
   ```

3. Return to the privileged EXEC mode.

   Example

   ```
   Device# end
   ```

4. Display the 802.11bg or 802.11a high-density parameters.

   Example

   ```
   DeviceDevice# show ap dot11 {24ghz | 5ghz} high-density
   ```

5. Display the summary of all the connected Cisco APs.

   Example

   ```
   DeviceDevice# show ap summary
   ```

Customize RF profile (CLI)

Customize RF profile using commands.

Customize 802.11 RF profile parameters.

Procedure

1. Enter the global configuration mode.

Example

```
DeviceDevice# configure terminal
```

2. Configure the 802.11bg, 802.11a or 802.11 6-GHz high-density parameters.

Example

```
DeviceDevice(config-rf-profile)# high-density rx-sop threshold {auto | custom
 | high | low | medium}
```

3. Display the summary of all the connected Cisco APs.

Example

```
DeviceDevice# show ap summary
```

4. Return to the privileged EXEC mode.

Example

```
Device# end
```

53 Client Limit

Topics:

- Client limits

Client limits

Describes client connection constraints, specifying maximum supported clients per platform, recommended usage thresholds, and best practices for maintaining reliable client operations.

A client limit is a wireless network feature that

- enforces a maximum number of client devices that can connect to an AP, and
- allows you to configure the client cap per AP radio or per WLAN.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 68: Feature history for client limits

Feature Name	Release Information	Feature Description
Client Limit and Delete	Cisco IOS XE Dublin 17.8.x	From this release, client limiting is supported per AP, per radio, and per AP-radio per WLAN. Client limiting is supported on the Cisco Catalyst 9136 Series APs in FlexConnect mode. The following commands are introduced: • association-limit • high-density clients count
Client limits	Cisco IOS XE Gibraltar 16.10.x	This feature enforces a maximum number of client devices per AP, supporting per-AP, per-radio, and per-WLAN configurations.

Limitations

Client limit limitations.

- All APs except the Cisco Catalyst 9136 Series support only 200 clients per radio. If you configure more than 200 clients for these APs, the AP radios can still support only 200 clients based on the AP capability value.
- Client limiting is supported on the Cisco Catalyst 9136 Series APs in FlexConnect mode.

Configure client limit per WLAN (GUI)

Configure client limit per WLAN using GUI.

Restrict the number of client devices that can connect to a specific WLAN.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.

2. Click a WLAN from the list of WLANs.
3. Click the **Advanced** tab.
4. Under the **Max Client Connections** settings, enter the client limit for **Per WLAN**, **Per AP Per WLAN**, and **Per AP Radio Per WLAN**.
5. Click **Update & Apply to Device**.

Configure client limit per WLAN (CLI)

Configure client limit per WLAN using commands.

Set the maximum number of clients permitted to associate with a specific WLAN, AP, or AP radio using CLI commands.

Procedure

1. Enter the privileged EXEC and the global configuration mode.

   Example

   ```
   Device# enable
   Device# configure terminal
   ```

2. Specify the WLAN name.

   Example

   ```
   Device(config)# wlan <wlan-profile-name>
   ```

   WLAN needs to be shut down in order to change the client limit parameters.

3. Configure the maximum number of clients that is associated to the given WLAN.

   Example

   ```
   Device(config-wlan)# client association limit <maximum-client-per-WLAN>
   ```

   Note

   Depending on the primary AP in the Cisco Embedded Wireless Controller network, the maximum number of clients supported varies.

4. Configure the maximum number of clients that is associated to an AP in the WLAN.

   Example

   ```
   Device(config-wlan)# client association limit ap <max-clients-per-AP-per-WLAN>
   ```

   The valid range is between zero and 1200 clients. The default value is zero.

> **Note**
>
> A Cisco Catalyst 9136 Series AP can support a maximum of 1200 clients.

5. Configure the maximum limit of clients that is associated to an AP radio in the WLAN.

   Example

   ```
   Device(config-wlan)# client association limit radio
   <maximum-clients-per-AP-radio-per-WLAN>
   ```

   The valid range is between zero to 500 clients. The default value is 200.

   > **Note**
   >
   > A Cisco Catalyst 9136 Series AP radio can support a maximum of 400 clients.

6. Return to the privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

7. Verify the current configuration of the WLAN and the corresponding client association limits.

   Example

   ```
   Device# show wlan id wlan-id 2
   ```

Configure client limit per AP (GUI)

Configure client limit per AP using GUI.

Set a limit on the number of clients that can associate with each AP.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. Click **Add**.
3. In the **General** tab, enter the name and description of the corresponding AP join profile.
4. Click the **Client** tab.
5. In the **Maximum Client Limit** field, enter the maximum client associations per AP. The valid values are between zero and 1200. The default value is zero.

6. Click **Apply to Device**.

Configure client limit per AP (CLI)

Configure client limit per AP using commands.

Set a limit on the maximum number of client devices allowed to associate with a single AP.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile-name
   ```

3. Configure the maximum client connections per AP.

   Example

   ```
   Device(config-ap-profile)# association-limit <max-client-connections>
   ```

   The default value is zero.

   > **Note**
   >
   > A Cisco Catalyst 9136 Series AP can support a maximum of 1200 clients.

4. Return to the privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

Configure client limit per radio (GUI)

Configure client limit per radio using GUI.

Set a maximum number of client connections per AP radio for an RF profile.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF/Radio**.
2. In the **RF** tab, click the required RF profile name from the displayed list of RF profiles.

   The **Edit RF Profile** page is displayed.
3. Click the **Advanced** tab.
4. Under the **High Density Parameters** section, in the **Max Clients** field, enter the maximum number of client connections per AP radio. The valid range is between zero and 500. The default value is 200 client connections.
5. Click **Update (and) Apply to Device**.

Configure client limit per radio (CLI)

Configure client limit per radio using commands.

Limit the number of client devices that can associate with each AP radio.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Configure the RF profile and enter the RF profile configuration mode.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rf-profile rf-profile-name
   ```
3. Configure the maximum number of client connections per AP radio.

   Example

   ```
   Device(config-rf-profile)# high-density clients count
   <maximum-client-connections>
   ```

   The valid range is between zero and 400. The default value is 200 client connections.

   > **Note**
   >
   > A Cisco Catalyst 9136 Series AP radio can support a maximum of 400 clients.

4. Return to the privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode. **end**

Example

```
Device(config)# end
```

Verify client limit

Verify client limit.

To verify client limit in local mode, run the following command:

```
Device# show wireless stats client delete reasons | sec Max
                Maximum client limit reached on AP                  : 0
                Maximum client limit reached on AP per wlan         : 0
                Maximum client limit reached on AP radio per wlan   : 0
                Maximum client limit reached on AP radio            : 0
```

To verify client limit in the FlexConnect central authentication mode, run the following command:

```
Device# show wireless stats client delete reasons | sec max
                AP limiting maximum client per AP               : 0
                AP limiting maximum client per AP radio per wlan : 0
                AP limiting maximum client per AP radio          : 0
```

54 IP Theft

Topics:

- IP theft
- SAE client exclusion visibility

IP theft

Introduces the fundamental principles of IP theft, including the types of intellectual property at risk, common methods of theft, and the impact on organizations.

An IP Theft feature is a wireless controller security mechanism that

- detects duplicate IP address usage among connected clients
- assigns precedence to clients based on a defined preference order, and
- blocks or excludes clients attempting to use IP addresses already assigned to others.

The IP Theft feature is enabled by default on the controller. If a wireless client tries to use an IP address assigned to a wired client, the controller marks it as a theft attempt.

Lower the idle timeout to prevent false IP theft

Lower the idle timeout for devices that may switch WLANs while out of range to prevent incorrect IP theft events.

When a device moves between WLANs while out of range of APs, it may retain the same IPv6 link-local address but present a different MAC address. Because the controller cannot immediately detect the move, it may interpret the reuse of the same IP address as an IP theft event.

To avoid this, lower the idle timeout value so the controller can promptly remove stale client entries from the original WLAN. This helps ensure that address reuse by legitimate roaming devices is not mistakenly treated as a theft attempt.

IP preference levels for IP theft detection

The controller uses IP preference levels to resolve conflicts when multiple clients claim the same IP address. These levels determine which client has priority based on the method used to learn the IP address. Preference order for IPv4 clients:

- DHCPv4
- ARP
- Data packets

Preference order for IPv6 clients:

- DHCPv6
- NDP (Neighbor Discovery Protocol)
- Data packets

Additional rules:

- Static wired clients always receive higher preference than dynamically assigned clients.
- Wired clients are prioritized over wireless clients when IP conflicts occur.

Configure IP theft (GUI)

Configure IP theft using GUI.

Prevent unauthorized use or reuse of IP addresses by configuring IP theft protections.

Procedure

1. Choose **Configuration** > **Security** > **Wireless Protection Policies** > **Client Exclusion Policies**.
2. Check the **IP Theft or IP Reuse** check box.
3. Click **Apply**.

Configure IP theft (CLI)

Configure IP theft using commands.

Enable IP theft detection and configure client exclusion policies using CLI.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the client exclusion policy.

   Example

   ```
   Device(config)# wireless wps client-exclusion ip-theft
   ```

Configure the IP theft exclusion timer (CLI)

Configure the IP theft exclusion timer using commands.

Set the exclusion timer to temporarily block IP addresses suspected of theft on a WLAN.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy profile-policy default-policy-profile
   ```

3. Specify the timeout, in seconds.

Example

```
Device(config-wireless-policy)# exclusionlist timeout time-in-seconds
```

The valid range is from zero-2147483647. Enter zero for no timeout.

Add static entries for wired hosts (CLI)

Add static entries for wired hosts.

Configure static wired bindings for devices on a VLAN to control IP address and interface assignment.

> **Note**
>
> If you configure wired bindings and SVI IP addresses on the device, the device uses those instead of DHCP.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the IPv4 or IPv6 static entry.

   Example

   ```
   Device(config)# device-tracking binding vlan vlan-id 20 ipv4-address 20.20.20.5
    interface gigabitEthernet ge-intf-num 1 hardware-or-mac-address 0000.1111.2222
   ```

   Example

   ```
   Device(config)# device-tracking binding vlan vlan-id 20 ipv6-address
   2200:20:20::6 interface gigabitEthernet ge-intf-num 1 hardware-or-mac-address
    0000.444.3333
   ```

   Use the first option to configure an IPv4 static entry or the second option to create an IPv6 static entry.

Verify IP theft configuration

Verify IP theft configuration.

Use the command to check if the IP theft feature is enabled or not:

```
Device# show wireless wps summary

Client Exclusion Policy
```

```
  Excessive 802.11-association failures   : Enabled
  Excessive 802.11-authentication failures: Enabled
  Excessive 802.1x-authentication         : Enabled
  IP-theft                                : Enabled
  Excessive Web authentication failure    : Enabled
  Cids Shun failure                       : Enabled
  Misconfiguration failure                : Enabled
  Failed Qos Policy                       : Enabled
  Failed Epm                              : Enabled
```

Use the commands to view additional details about the IP theft feature:

```
Device# show wireless client summary

Number of Local Clients: 1

MAC Address    AP Name                WLAN State              Protocol Method
     Role
-------------------------------------------------------------------------------
000b.bbb1.0001 SimAP-1                2    Run                11a      None
     Local

Number of Excluded Clients: 1

MAC Address    AP Name                WLAN State              Protocol Method

-------------------------------------------------------------------------------
10da.4320.cce9 charlie2               2    Excluded           11ac     None
```

```
Device# show wireless device-tracking database ip

IP                              VLAN  STATE       DISCOVERY   MAC
  ------------------------------------------------------------------------
  20.20.20.2                     20    Reachable   Local       001e.14cc.cbff

  20.20.20.6                     20    Reachable   IPv4 DHCP   000b.bbb1.0001
```

```
Device# show wireless exclusionlist

Excluded Clients

MAC Address       Description          Exclusion Reason               Time
Remaining
-------------------------------------------------------------------------------
10da.4320.cce9                         IP address theft                  59
```

> **Note**
>
> Client exclusion timer deletes the entry from exclusion list with a granularity of 10 seconds. The entry is checked to retain or delete after every 10 seconds. There are chances that the running timer value for excluded clients might display negative values upto 10 seconds.

> **Note**
>
> When you enable client exclusion, the system adds the client to the exclusion list. This feature does not prevent the client from being deleted.

```
Device# show wireless exclusionlist client mac 12da.4820.cce9 detail

Client State : Excluded
Client MAC Address : 12da.4820.cce9
Client IPv4 Address: 20.20.20.6
Client IPv6 Address: N/A
Client Username: N/A
Exclusion Reason : IP address theft
Authentication Method : None
Protocol: 802.11ac
AP MAC Address : 58ac.780e.08f0
AP Name: charlie2
AP slot : 1
Wireless LAN Id : 2
Wireless LAN Name: mhe-ewlc
VLAN Id : 20
```

SAE client exclusion visibility

Introduces SAE client exclusion visibility, enabling controllers to access client exclusion list data in Flex central authentication deployments with SAE-enabled WLANs, and reducing the need for manual log collection for enhanced operational efficiency.

SAE client exclusion visibility is a feature that

- addresses a limitation in Flex central authentication deployments with SAE-enabled WLANs
- makes client exclusion list data available on the controller, and
- reduces manual log collection for customers.

In FlexConnect central authentication deployments with SAE enabled WLANs, the controller does not have access to client exclusion list data because SAE authentication messages are processed at the AP, and authentication failures are not reported to the controller. As a result, you must collect logs manually, which increases operational costs. This feature provides a solution to this issue.

This feature addresses only Flex central authentication, WPA3 + SAE/FT-SAE/SAE-EXT-KEY/FT-SAE- EXT-KEY WLANs.

Feature history for SAE client exclusion visibility
Table 69: Feature history

Feature Name	Release Information	Feature Description
SAE client exclusion visibility	Cisco IOS XE 17.18.2	The SAE authentication client exclusion feature overcomes the limitations of Flex central authentication deployments with SAE-enabled WLANs. The feature reports client authentication failures and exclusion data to the controller. It generates syslog messages for these events and allows administrators to configure the duration that a client remains excluded. It centralizes troubleshooting information that was previously only available on individual APs, reducing the need for manual log collection.

Configure client exclusion timeout (GUI)

To set the duration for which a client remains excluded from the system using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. In the **General** tab, specify the name and description for the policy profile.
3. Click the **Advanced** tab.
4. In the **WLAN Timeout** section, enter a value for the **Client Exclusion Timeout** field, in seconds. This option is enabled by default. The valid range is 0 to 2,147,483,647 seconds. The default is 60 seconds.
5. Click **Apply to Device**.

The system enables and configures the SAE authentication client exclusion feature for the selected wireless policy profile.

What to do next

Monitor excluded clients.

Monitor client exclusion

Monitor excluded clients.

Procedure

1. Choose **Monitoring** > **Wireless** > **Clients**.
2. Click the **Excluded Clients** tab.
   The excluded clients and their details are listed on this page. Additionally, you can add an excluded client by clicking the **Add** button and entering the MAC address and the description.

Configure client exclusion timeout (CLI)

Set the duration for which a client remains excluded from the system.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a policy profile for the WLAN.

   Example

   ```
   Device# wireless profile policy policy-profile-name
   ```

   Replace `policy-profile-name` with the name of your wireless policy profile. For example, `flex-profile-policy`.

3. Disable central DHCP for locally switched clients.

   Example

   ```
   Device(config-wireless-policy)# no central dhcp
   ```

4. Disable central switching.

   Example

   ```
   Device(config-wireless-policy)# no central switching
   ```

5. Enable or disable exclusion listing on the WLAN by setting the exclusion time for the client. This feature is enabled by default.

   Example

   ```
   Device(config-wireless-policy)# [no] exclusionlist timeout seconds
   ```

   The timeout is set in seconds. The default is 60 seconds. The valid range is from 0 to 2,147,483,647 seconds. If you set the value to 0, the client remains excluded indefinitely.

   Use the **no** form of this command to disable the feature.

The specified exclusion timeout is applied to the WLAN policy profile. This duration determines how long the excluded clients remain unable to connect.

```
Device# configure terminal
Device(config)# wireless profile policy default-policy-profile
Device(config-wireless-policy)# no central dhcp
Device(config-wireless-policy)# no central switching
Device(config-wireless-policy)# [no] exclusionlist timeout 60
```

Verify client exclusion data

Verify client exclusion data using these commands.

Verify the wireless policy profile information, including the exclusion status and timeout for clients, using this command:

```
Device# show wireless profile policy detailed flex-profile-policy
WLAN Switching Policy
  Flex Central Switching            : DISABLED
  Flex Central Authentication       : ENABLED
  Flex Central DHCP                 : DISABLED
  Flex NAT PAT                      : DISABLED

Exclusionlist Params
  Exclusionlist                     : ENABLED
  Exclusion Timeout                 : 60
```

View a summary of wireless clients, including information about excluded clients, using this command:

```
Device# show wireless client summary
Number of Clients: 0
Number of Excluded Clients: 1
MAC Address AP Name Type ID State Protocol Method
 ------------------------------------------------------------------------------------------

90XX.84XX.63XX WLAN 3 Excluded N/A None
```

Verify the list of excluded wireless clients using this command:

```
Device# show wireless exclusionlist
Number of Excluded Clients : 1
MAC Address Description Exclusion Reason Time Remaining
------------------------------------------------------------------------------------------------

90XX.84XX.63XX          SAE authentication failure 49
```

55 Unscheduled Automatic Power Save Delivery

Topics:

- Unscheduled automatic power save delivery overview

Unscheduled automatic power save delivery overview

Introduces the fundamentals of power save delivery, then guides users through configuring Power Save (U-APSD) using CLI to optimize wireless device energy efficiency.

Unscheduled automatic power save delivery (U-APSD) is a power-saving mechanism that

- is defined as a QoS facility in IEEE 802.11e standards
- delivers multiple downlink packets in response to a single uplink trigger, and
- reduces wireless latency and extends mobile device battery life

Additional information

- U-APSD activates automatically when WMM (Wi-Fi Multimedia) is enabled.
- U-APSD eliminates the need for polling individual packets at the access point.

Possible sequence for U-APSD operation:

A client device sends a single uplink trigger packet. In response, the access point delivers multiple queued downlink packets based on that single trigger.

View unscheduled automatic power save delivery

View unscheduled automatic power save delivery using commands.

Show detailed information of a client by MAC address.

```
Device# show wireless client mac-address 2B:5B:B3:18:56:E9 detail
Output Policy State : Unknown
Output Policy Source : Unknown
WMM Support : Enabled
U-APSD Support : Enabled
U-APSD value : 15
APSD ACs    : BK(T/D), BE, VI(T/D), VO(T/D)
Power Save : OFF
Current Rate :

--------------------------
BK : Background
BE : Best Effort
VI : Video
VO : Voice.

T: UAPSD Trigger Enabled
D: UAPSD Delivery Enabled
T/D : UAPSD Trigger and Delivery Enabled
```

56 Target Wake Time

Topics:

- Target wake time

Target wake time

Target Wake Time details.

Target wakeup time (TWT) is a power-saving mechanism that

- allows an AP to schedule specific times for devices or stations (STAs) to wake up and communicate
- enables those devices to remain in a power-saving state for longer periods, and
- reduces overall network congestion.

Feature history

Table 70: Feature history for target wake time

Feature name	Release information	Feature description
Target wake time	Cisco IOS XE 17.2.1	Target wake time (TWT) allows an AP to manage activity in the Wi-Fi network, in order to minimize medium contention between Stations (STAs), and to reduce the required amount of time that an STA in the power-save mode needs to be awake. This is achieved by allocating STAs to operate at non-overlapping times, and/or frequencies, and concentrate the frame exchanges in predefined service periods.

Delivery traffic indication map

Unscheduled automatic power-save delivery

With 802.11e, the new power-saving mechanism was introduced that helps voice-capable Wi-Fi devices, as voice packets are transmitted at short time intervals, typically 20 ms/sec.

The existing Wi-Fi client power-saving mechanisms have been in use since 802.11b, where the client devices sleep between AP beacons or multiple beacons, waking up only when they have data to transmit (they can transmit at any time, as AP does not sleep).

A Delivery Traffic Indication Map (DTIM) is a bit-map that

- is contained within beacons
- indicates an AP has buffered downlink traffic, and
- specifies which particular clients are awaiting transmission.

If a client has a DTIM bit set, it can retrieve data from the AP by sending a Power-Save Poll (PS-Poll) frame to the AP. This power-save scheme is effective but only allows clients to doze for a small beacon interval. Clients still need to wake up several times per second to read DTIM from the beacon frame of the AP.

Unscheduled automatic power-save delivery (U-APSD) allows a power-save client to sleep at intervals within a beacon period. AP buffers the downlink traffic until the client wakes up and requests its delivery.

> **Note**
>
> By default Target Wake Time (TWT) is disabled on the controller. To enable TWT, run the **ap dot11 {24ghz | 5ghz| 6ghz} dot11ax twt-broadcast** command.

Optimize power-saving with target wake time

TWT capable STA can either negotiate an individual TWT agreement with TWT-scheduling AP, or it can elect to be part or member of Broadcast TWT agreement existing on the AP. An STA does not need to be aware that a TWT service period can be used to exchange frames with other STAs.

Any frame exchanges that occur during a TWT service period can use any Physical Protocol Data Unit (PPDU) format that is supported by both the AP and STA involved in that TWT agreement. This includes formats such as High Efficiency Multi-User PPDU (HE MU PPDU) and High Efficiency Trigger-Based PPDU (HE TB PPDU).

The TWT agreement types are:

- **Individual TWT**: An individual TWT session is established through negotiation between the AP and a single STA. This arrangement provides a dedicated service period for both downlink and uplink communication between the AP and that specific STA. The expected traffic is highly concentrated within the negotiated service period, achieving up to 99% accuracy. Each service period begins at a defined offset from the Target Beacon Transmission Time (TBTT), lasts for the agreed-upon duration, and then repeats at regular intervals.
- **Solicited TWT**: In a solicited TWT agreement, the STA initiates the TWT session by sending a request to the AP.
- **Unsolicited TWT** : In an unsolicited TWT agreement, the AP initiates the TWT agreement with the STA. The AP sends a TWT response specifying the service period, which the STA then accepts.
- **Broadcast TWT**: In a broadcast TWT agreement, a High-Efficiency AP invites the STA to join a broadcast TWT operation, which may involve participating in an existing broadcast service period or a newly established service period.

TWT requesting STA communicates the wake scheduling information to its TWT responding AP, which then devises a schedule and delivers the TWT values to the TWT requesting STA when a TWT agreement has been established between them.

Configure target wake time at radio level (CLI)

Describes the steps to configure TWT at the radio level.

Complete this task to configure target wake time at the radio level.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Disable the 802.11a, 802.11b, or 802.11 6-GHz network.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} shutdown
```

3. Configure the 802.11ax parameters.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} dot11ax
```

802.11ax cannot be disabled on the 6-GHz band.

4. Configure 802.11 6-GHz dot11ax target wake-up time.

Example

```
Device(config)# [no] ap dot11 {24ghz | 5ghz | 6ghz} dot11ax target-wakeup-time
```

5. Configure 802.11 6-GHz dot11ax target wake-up time broadcast.

Example

```
Device(config)# [no] ap dot11 {24ghz | 5ghz | 6ghz} dot11ax twt-broadcast
```

> **Note**
>
> By default TWT is disabled on the controller. Enable TWT by running this command.

6. Enable the 802.11a, 802.11b, or 802.11 6-GHz network.

Example

```
Device(config)# no ap dot11 {24ghz | 5ghz | 6ghz} shutdown
```

7. Display and verify the 802.11ax network configuration details, which includes information about Target Wakeup Time and Target Wakeup Broadcast.

Example

```
Device(config)# show ap dot11 {24ghz | 5ghz | 6ghz} network
```

The target wakeup time at the radio level is configured.

```
Device# configure terminal
Device(config)# ap dot11 24ghz shutdown
Device(config)# ap dot11 24ghz dot11ax
Device(config)# [no] ap dot11 24ghz dot11ax target-wakeup-time
Device(config)# [no] ap dot11 24ghz dot11ax twt-broadcast
Device(config)# no ap dot11 24ghz shutdown
```

Enable target wakeup time on WLAN (CLI)

Describes the CLI steps to enable target wakeup time on WLAN.

Complete this task to enable target wakeup time on the WLAN.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration submode.

   Example

   ```
   Device(config)# wlan wlan-profile-name
   ```

   The *wlan-profile-name* is the profile name of the configured WLAN.

3. Disable the WLAN network.

   Example

   ```
   Device(conf-wlan)# shutdown
   ```

4. Configure target wake time mode on WLAN.

   Example

   ```
   Device(conf-wlan)# dot11ax target-waketime
   ```

5. Configure the TWT broadcast support on WLAN.

   Example

   ```
   Device(conf-wlan)# dot11ax twt-broadcast-support
   ```

6. Enable WLAN.

   Example

   ```
   Device(conf-wlan)# no shutdown
   ```

7. Display the details of the configured WLAN, including Target Wakeup Time and Target Wakeup Broadcast Time.

   Example

   ```
   Device# show wlan {all | id | name | summary}
   ```

The target wakeup time is enabled on the WLAN.

```
Device# configure terminal
Device(config)# wlan wlan-profile-name
Device(conf-wlan)# shutdown
Device(conf-wlan)# dot11ax target-waketime
Device(conf-wlan)# dot11ax twt-broadcast-support
Device(conf-wlan)# no shutdown
```

Disable target wakeup time on WLAN (CLI)

Describes the CLI steps to disable Target Wakeup Time on the WLAN.

Complete this task to disable target wakeup time on WLAN.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration submode.

   Example

   ```
   Device(config)# wlan wlan-profile-name
   ```

   The *wlan-profile-name* is the profile name of the configured WLAN.

3. Disable the WLAN.

   Example

   ```
   Device(conf-wlan)# shutdown
   ```

4. Disable the target wake time mode on WLAN.

   Example

   ```
   Device(conf-wlan)# no dot11ax target-waketime
   ```

5. Disable the TWT broadcast support on WLAN.

   Example

   ```
   Device(conf-wlan)# no dot11ax twt-broadcast-support
   ```

6. Enable WLAN.

Example

```
Device(conf-wlan)# no shutdown
```

The target wakeup time is disabled on the WLAN.

```
Device# configure terminal
Device(config)# wlan wlan-profile-name
Device(conf-wlan)# shutdown
Device(conf-wlan)# no dot11ax target-waketime
Device(conf-wlan)# no dot11ax twt-broadcast-support
Device(conf-wlan)# no shutdown
```

Configure target wake time (GUI)

Configure Target Wake Time using GUI.

Complete this task to configure the target wakeup time and the broadcast for target wakeup time.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **Parameters**.

   The parameters page is displayed where you can configure global parameters for 5 GHz band and 2.4 GHz band radios.

2. In the **11ax Parameters** section, check the **Target Wakeup Time** check box and the **Target Wakeup Time Broadcast** check box to configure target wakeup time and broadcast target wakeup time.

The target wakeup time and the target wakeup time broadcast are configured.

Verify target wakeup time

Verify Target Wakeup Time.

To verify Target Wakeup Time and Target Wakeup Time Broadcast, use this command

```
Device#show ap dot11 24ghz network
.
.
.
802.11ax                                : Enabled
Target Wakeup Time                      : Enabled
Target Wakeup Time Broadcast            : Enabled
.
.
.
```

57 Enabling USB Port on Access Points

Topics:

- USB port as power source for access points

The document explains how to enable and configure the USB port on Cisco Catalyst 9800 Series Access Points using both GUI and CLI methods. It details steps to enable or disable the USB module, override USB status from AP profiles, and notes specific considerations for certain AP models like the 9105AXW. This configuration allows management of USB ports for power and device connectivity on the APs.

USB port as power source for access points

Outlines the capabilities of USB ports on Access Points, guiding users through profile configuration, USB settings management via CLI and GUI, and monitoring USB configurations for optimal power usage and device integration.

A USB port as power source for access points is a power feature that

- provides up to 2.5W of power to connected USB devices
- automatically shuts down when a device draws more than 2.5W to prevent overload, and
- is available on select Cisco AP models as specified in their datasheets.

USB port power specifications

The USB port enables when the power draw is 2.5W and lower. Refer to the datasheet of your AP to check if the AP has a USB port that can act as a source of power.

Note

Both IW6300 and ESW6300 APs have a USB port that can act as a source of power up to 4.5W for some USB devices.

Note

The controller records the last five power-overdrawn incidents in its logs.

Caution

When unsupported USB device is connected to the Cisco AP, the following message is displayed:

```
The inserted USB module is not a supported device. The behavior of this USB
device and the impact to the Access Point is not guaranteed. If Cisco determines
that a fault or defect can be isolated due to the use of third-party USB modules
installed by a customer or reseller, Cisco may withhold support under warranty
or support program under contract. In the course of providing support for Cisco
networking products, the end user may be required to install Cisco-supported
USB modules in the event Cisco determines that removing third-party parts will
assist Cisco in diagnosing root cause for troubleshooting purposes. Cisco also
reserves the right to charge the customer per then-current time and material
rates for services provided to the customer when Cisco determines, after having
provided such services, that an unsupported device caused the root cause of
the defective product
```

Configure an AP profile (CLI)

Configure an AP Profile using commands.

Create and configure an AP profile to manage access point settings and behavior.

AP profiles allow you to define configuration parameters that can be applied to multiple access points, providing consistent settings and simplified management across your wireless network deployment.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile
   ```

   Example

   ```
   Device(config)# ap profile xyz-ap-profile
   ```

   Note

   When you delete a named profile, the APs associated with that profile will not revert to the default profile.

3. Enable USB for each AP profile.

   Example

   ```
   Device(config-ap-profile)# usb-enable
   ```

   Note

   By default, the USB port on the AP is disabled.

   Use the **no usb-enable** command to disable USB for each AP profile.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-profile)# end
   ```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The AP profile is now configured and can be applied to access points to manage their USB settings and other profile-specific configurations.

Configure USB settings for an access point (CLI)

Configure USB settings for an AP using commands.

Enable and manage USB functionality on access points to support connected USB devices or modules.

USB ports on access points can be used for various purposes including storage, cellular modules, or other peripheral devices. This configuration allows administrators to control USB port functionality at the individual AP level.

Procedure

---

1. Enter privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Enable the USB port on the AP.

   Example

   ```
   Device# ap name ap-name usb-module
   ```

   Example

   ```
   Device# ap name AP44d3.xy45.69a1 usb-module
   ```

   Use the **ap name** *ap-name* **no usb-module** command to disable the USB port on the AP.

   > **Note**
   >
   > If you are using Cisco Catalyst 9105AXW AP and if you enable the USB port (.3at PoE-in), it is not possible to enable the USB PoE-out at the same time.

3. Override USB status of the AP profile and consider the local AP configuration.

   Example

   ```
   Device# ap name ap-name usb-module override
   ```

   Example

   ```
   Device# ap name AP44d3.xy45.69a1 usb-module override
   ```

Use the **ap name** *ap-name* **no usb-module override** command to override USB status of the AP and consider the AP profile configuration.

> **Note**
>
> You can configure the USB status for an AP only if you enable USB override for it.

The USB settings are now configured for the specified access point, allowing control over USB port functionality and override behavior.

Configure USB settings for an access point (GUI)

Configure USB Settings for an AP using GUI.

Before you begin

Follow these steps to configure USB settings for an access point using the GUI:

Procedure

1. Choose **Configuration > Wireless > Access Points**.
2. In the **Access Points** window, click the name of the AP.
3. In the **Edit AP** window, click the **Interfaces** tab.
4. In the **USB Settings** section, configure the **USB Module State** as either of the following:
   - **ENABLED**: Enables the USB port on the AP
   - **DISABLED**: Disables the USB port on the AP

   > **Note**
   >
   > If you are using Cisco Catalyst 9105AXW AP and if you enable the USB port (.3at PoE-in), it is not possible to enable the USB PoE-out at the same time.

5. Configure **USB Override** as either of the following:
   - **ENABLED**: Overrides USB status of the AP profile and considers the local AP configuration
   - **DISABLED**: Overrides USB status of the AP and considers the AP profile configuration

   > **Note**
   >
   > You can configure the USB status for an AP only if you enable USB override for it.

6. Click **Apply & Update to Device**.

USB configuration monitoring for access points (CLI)

Lists CLI commands to monitor USB configurations for access points.

Monitor USB configurations for access points using command-line interface commands to view inventory details, module summaries, configuration details, and status information.

- To view the inventory details of APs, use this command:

  show ap name *ap-name* **inventory**

  The following is a sample output:

```
Device# show ap name AP500F.8059.1620  inventory
NAME: AP2800    , DESCR: Cisco Aironet 2800 Series (IEEE 802.11ac) Access
Point
PID: AIR-AP2802I-D-K9  ,  VID: 01,  SN: XXX1111Y2ZZZZ2800
NAME: SanDisk    , DESCR: Cruzer Blade
PID: SanDisk  ,  SN: XXXX1110010, MaxPower: 224
```

- To view the summary of an AP module, use this command:

  show ap module summary

  This is a sample output:

```
Device# show ap module summary
AP Name               External Module       External Module PID   External
Module Description
-------------------------------------------------------------------------------------
AP500F.1111.2222      Enable                SanDisk               Cruzer
Blade
```

- To view the USB configuration details for each AP, use this command:

  show ap name *ap-name* **config general**

  This is a sample output:

```
Device# show ap name AP500F.111.2222 config general

.
.
.
USB Module Type.................................. USB Module
USB Module Status................................ Disabled
USB Module Operational State..................... Enabled
USB Override ……..... Enabled
```

- To view status of the USB module, use this command:

  show ap profile name *xyz* **detailed**

  This is a sample output:

```
Device# show ap profile name xyz detailed
USB Module               : ENABLED
```

58 Dynamic Frequency Selection

Topics:

- Dynamic frequency selection
- Zero wait dynamic frequency selection

The shorter description for the chapter on Dynamic Frequency Selection (DFS) explains how Cisco Catalyst 9800 Series Wireless Controllers detect radar signals on 5 GHz radios and automatically change channels to avoid interference, ensuring regulatory compliance and network performance. It also covers configuration steps for mesh profiles to enable radar detection and coordinated channel changes.

Dynamic frequency selection

Introduces Dynamic Frequency Selection concepts, details Channel Availability Check history and operation, and provides instructions for configuring DFS using GUI and CLI methods with verification procedures.

Dynamic Frequency Selection (DFS) is a process that

- detects radar signals on DFS-enabled 5.0-GHz (802.11a/h) radios
- automatically sets the frequency to avoid interference with radar signals, and
- ensures radios configured for use in a regulatory domain do not interfere with radar systems.

Flex DFS behavior

In normal DFS, when a radar signal is detected on any of the channels in the 40-MHz or 80-MHz bandwidth, the whole channel is blocked. With Flex DFS, if the radar signals are not detected on the secondary channel, the AP is moved to a secondary channel with a reduction in the bandwidth, usually, by half.

Feature history for channel availability check (CAC)

Lists Channel Availability Check feature history.

This table provides release and related information for features explained in this module. These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Table 71: Feature history for channel availability check (CAC)

Release	Feature	Feature Information
Cisco IOS XE Bengaluru 17.5.1	Channel Availability Check (CAC)	When a DFS channel is selected for an AP radio, the AP radio scans the channel to check for any radar signals before transmitting any frames in the DFS frequency. This process is called Channel Availability Check (CAC).

Channel availability check

Explains the process and requirements for channel availability check on DFS channels.

A channel availability check is a DFS channel scanning process that

- scans DFS channels for radar signals before transmitting any frames in the DFS frequency
- executes before you set a DFS channel for the radio, and
- marks channels as non-available for 30 minutes if radar is detected.

Channel availability check behavior

When a DFS channel is selected for an AP radio, the AP radio scans the channel to check for any radar signals before transmitting any frames in the DFS frequency. This process is called Channel Availability Check (CAC).

> **Note**
>
> CAC is executed before you set a DFS channel for the radio.

If the AP detects that a radar is using a specific DFS channel, the AP marks the channel as non-available and excludes it from the list of available channels. This state lasts for 30 minutes after which the AP checks again to see, if the channel can be used for Wi-Fi transmissions.

> **Note**
>
> - The CAC performed during a boot process takes anywhere between 1 and 10 minutes depending on the country. This is the reason as to why the DFS channels are not available immediately when an AP reboots.
> - APs in the ETSI domain scan channels which are not supported by the controller, as the hardware has the ability to scan.

DFS verification

Lists commands to verify DFS configuration and status.

Use the following commands to verify the DFS configuration:

To display the 802.11h configuration, use the following command:

```
Device# show wireless dot11h
```

To display the auto-rF information for 802.11h configuration, use the following command:

```
Device# show ap auto-rf dot11 5ghz
```

To display the auto-rF information for a Cisco AP, use the following command:

```
Device# show ap name ap1 auto-rf dot11 5gh
```

To display the channel details for a Cisco AP, use the following command:

```
Device# show ap dot11 5ghz summary
AP Name Mac Address Slot Admin State Oper State Width Txpwr Channel
---------------------------------------------------------------------------------
pnp-ap 04eb.409e.b560 1 Enabled Up 40 *8/8 (3 dBm) (52,56)
BLDG1-9130-RACK-1568 04eb.409f.11a0 1 Disabled Down 40 4/8 (15 dBm) (100,104)#
```

> **Note**
>
> In the show command, # is added right next to the channel whenever CAC is running on an AP radio.

Zero wait dynamic frequency selection

Describes a wireless feature that eliminates service outages when access points move to DFS channels by pre-monitoring potential channels.

Zero wait dynamic frequency selection is a wireless feature that

- eliminates the 60-second to 10-minute service outage when access points move to Dynamic Frequency Selection (DFS) channels
- enables access points to pre-monitor and perform Channel Availability Check (CAC) on potential DFS channels before switching, and
- is supported only in U.S. and Europe regulatory domains.

Channel availability check process

APs monitor and perform Channel Availability Check (CAC) on a potential channel for 60 seconds when AP moves to Dynamic Frequency Selection (DFS) channels. Further, the AP ensures that there is no radar operating in the same frequency range before advertising beacons and serving clients. When the AP moves to a DFS, there is a service outage for a minute. This outage can be higher and extend up to 10 minutes.

Configure zero wait dynamic frequency selection globally

Configure Zero Wait Dynamic Frequency Selection globally using the GUI to allow APs to change to DFS without service outage.

Zero Wait DFS allows access points to change to Dynamic Frequency Selection channels without experiencing service interruption. This configuration applies globally to all access points in your network.

Follow these steps to configure Zero Wait Dynamic Frequency Selection globally:

Procedure

1. Choose **Configuration > Radio Configurations > RRM**.
2. In the **RRM** page, click the **5 GHz Band** tab.
3. Click the **DCA** tab.
4. Select the **Zero Wait DFS** check box to allow the AP to change to DFS without a service outage.
5. Click **Apply**.

Zero Wait Dynamic Frequency Selection is now enabled globally for all access points in your network.

Configure zero wait dynamic frequency selection globally (CLI)

Configure Zero Wait Dynamic Frequency Selection globally using commands.

Enable Zero Wait Dynamic Frequency Selection to improve channel availability and reduce channel change delays on 5-GHz radios.

Zero Wait Dynamic Frequency Selection optimizes radar detection and channel switching processes by allowing access points to pre-scan DFS channels. This reduces service interruption when radar is detected on the current operating channel.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable the Zero Wait Dynamic Frequency Selection feature.

   Example

   ```
   Device(config)# ap dot11 5ghz rrm channel zero-wait-dfs
   ```

   By default, the feature is disabled.

   Use the **no** form of this command to disable the feature.

   > **Note**
   >
   > The Zero Wait Dynamic Frequency Selection feature is only available on a 5-GHz radio.

Zero Wait Dynamic Frequency Selection is now enabled globally, allowing 5-GHz radios to perform background DFS scanning for improved channel switching performance.

Enable zero wait dynamic frequency selection on a RF profile (GUI)

Configure Zero Wait Dynamic Frequency Selection on a RF Profile using GUI.

Enable Zero Wait Dynamic Frequency Selection to allow the AP to change to DFS without a service outage.

Before you begin

Follow these steps to enable Zero Wait Dynamic Frequency Selection on a RF Profile:

Procedure

1. Choose **Configuration > Tags & Profiles > RF/Radio**.
2. In the **RF** tab, click **Add**.

   The **Add RF Profile** page is displayed.

3. Enter the name for the RF profile.
4. From the **Radio Band** drop-down, choose the **5 GHz** band.
5. Click the **RRM** tab.

6. Click the **DCA** tab.
7. Select the **Zero Wait DFS** check box to allow the AP to change to DFS without a service outage.
8. Click **Apply to Device**.

Enable zero wait dynamic frequency selection on a RF profile (CLI)

Configure zero wait dynamic frequency selection on a RF profile using commands.

Enable the Zero Wait Dynamic Frequency Selection feature to reduce DFS channel availability delays on 5 GHz RF profiles.

Zero Wait DFS allows access points to immediately use DFS channels by pre-scanning them, eliminating the typical 60-second wait time required for DFS channel availability checks. This configuration is applied at the RF profile level to improve wireless performance and channel utilization.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a radio frequency (RF) profile and enter RF profile configuration mode.

   Example

   ```
   Device(config)# ap dot11 5ghz rf-profile profile-name
   ```

   Example

   ```
   Device(config)# ap dot11 5ghz rf-profile test-dfs
   ```

3. Enable the Zero Wait Dynamic Frequency Selection feature for the RF profile.

   Example

   ```
   Device(config-rf-profile)# channel zero-wait-dfs
   ```

   Use the **no** form of this command to disable the feature.

Zero Wait DFS is now enabled on the RF profile, allowing access points using this profile to immediately utilize DFS channels without the standard waiting period.

Zero wait dynamic frequency selection configuration verification

Lists commands to verify Zero Wait Dynamic Frequency Selection configuration.

Use the following commands to verify the DFS configuration.

To display the Zero Wait DFS configuration on an AP, use the following command:

```
Device# show ap name ap1 config slot 1 | inc Zero

  Zero Wait DFS Parameters
    Zero Wait DFS Capable                        : Yes
    CAC Domain                                   : None
```

To display the global configuration related to the Zero Wait Dynamic Frequency Selection feature, use the following command:

```
Device# show ap dot11 5ghz channel | inc Zero

  Zero Wait DFS Parameters
    Zero Wait DFS Capable                        : Yes
    CAC Domain                                   : None
```

To display the RF profile configuration related to the Zero Wait Dynamic Frequency Selection feature, use the following command:

```
Device# show ap rf-profile name test detail | sec Zero

Description                          :
RF Profile Name                      : test
Band                                 : 5 GHz
Transmit Power Threshold v1          : -70 dBm
Min Transmit Power                   : -10 dBm
Max Transmit Power                   : 30 dBm
.
.
.
Guard Interval                       : default
Zero Wait DFS                        : Enabled
```

59 Cisco Access Points with Tri-Radio

Topics:

- Cisco access points with tri-radio

The document describes the Tri-Radio feature for Cisco Access Points, detailing how three radios operate to optimize performance in high-density environments. It explains radio roles, dual radio mode configurations, and provides guidelines and procedures for managing Tri-Radio settings on Cisco Catalyst 9800 Series Wireless Controllers.

Cisco access points with tri-radio

Describes Cisco APs with tri-radio capabilities and their operational modes for high density environments.

A Cisco access point with tri-radio is a wireless device that

- operates three radios designed for high density environments
- runs one dedicated 2.4-GHz 4x4 mode radio and one 5-GHz 8x8 mode radio by default, and
- enables the 8x8 radio to split into two independent 5-GHz 4x4 radios when dual radio mode is enabled.

Tri-radio operational modes and configuration

Access Points with three radios are designed for high density environments. The APs by default run one dedicated 2.4-GHz 4x4 mode radio and one 5-GHz 8x8 mode radio. In the default mode, the radios are managed by the Flexible Radio Assignment (FRA), and the Dual Radio Mode is in the disabled state indicating that the radios have either been assigned as client serving 8x8 radio or have not yet been evaluated by FRA.

When you enable the dual radio mode setting, the 8x8 radio is split to two independent 5-GHz 4x4 radios. In this mode, slot 1 and slot 2 are active independent 4x4 radio interfaces. They can serve different user groups with different assigned channels.

> **Note**
>
> To disable the dual radio mode, you must first disable the admin status of the subordinate radio. Otherwise, a warning message is displayed.

In Cisco IOS XE Dublin 17.13.1, the **ap tri-radio** command cannot be configured, since the Tri-radio settings are enabled by default, and cannot be disabled.

A tri-radio AP has upto two configurable 5-GHz radios. The following table describes the radio role and its deployment benefits:

Table 72: 5-GHz radio operational modes and criteria

Radio Role		Driving Factors
Radio 1	**Radio 2**	
8x8 Client-Serving	None	• Preferred operation: 160 MHz or 80 + 80 MHz • Higher MU-MIMO stations • Required higher number of Spatial Streams (SS)
4x4 Client-Serving	4x4 Client-Serving	• Preferred operation: 80 MHz or below • High Capacity in low or medium density • Directional antenna units (Coverage Slicing)

Radio Role		Driving Factors
Radio 1	**Radio 2**	
4x4 Client-Serving	4x4 Monitor	• Preferred operation: 80 MHz or below • Lower MU-MIMO stations • Better channel reuse in high density • Monitoring application requires 4x4 Rx

The following table lists the different radio modes and roles supported by the AP:

Table 73: Tri-radio AP radio configuration

Setup	Radio Mode	Maximum Radio Capability	Dual Role Mode
1	2.4-GHz + 5-GHz	2.4-GHz, 4 antennas, 4SS, and 20 MHz 5-GHz, 8 antennas, 4SS, and 160 MHz	Disabled
2	5-GHz + 5-GHz	2.4-GHz, 4 antennas, 4SS, and 20 MHz 5-GHz, 4 antennas, 4SS, and 80 MHz 5-GHz, 4 antennas, 4SS, and 80 MHz	Enabled

In the Cisco IOS XE 17.2.1 Release, FRA manages the role assignment for each radio independently. You can set the radio mode as automatic or manual, and select either Client-Serving role or Monitor role as the radio role. Based on the dual radio mode configuration, the role selection is available for one or for both interfaces.

Guidelines and restrictions for tri-radio access points

Outlines configuration guidelines and hardware restrictions for tri-radio access point deployment and management.

- Dual radio mode is set to **Auto** by default. FRA manages the dual radio mode in **Auto** mode.
- The tri-radio support for AP with external antenna is as follows:
  - RP-TNC antenna is supported in Cisco Catalyst 9130AX Series APs.
  - The C-ANT9101, C-ANT9102, and C-ANT9103 antennas on Cisco Catalyst 9130AX Series APs support 2 radios (2.4-GHz (4x4) and 5-GHz (8x8)). This antennas does not support two 5-GHz (4x4) radios due to hardware limitation.
- From Cisco IOS XE Cupertino 17.7.x, the Tri-Radio feature is supported in Cisco Catalyst 9124 Series APs.

Configure tri-radio for AP (GUI)

Configure Tri-Radio for AP using GUI.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **Network**.

   The **Network** > **5 GHz Radios** page is displayed.

2. In the **General** tab, the **Tri-Radio Mode** check box is enabled by default. Check the check box to enable this feature.

   > **Note**
   >
   > Enabling the check box will allow Tri-Radio Mode for all APs capable of this feature. If you disable this later, the affected APs will revert to their default radio configuration.

3. Click **Apply**

Configure the tri-radio (CLI)

Configure the tri-radio using commands.

Enable tri-radio functionality on supporting access points to optimize wireless performance through automatic dual radio role management.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure all supporting tri-radio AP's dual radio role in auto mode.

   Example

   ```
   Device(config)# [no] ap tri-radio
   ```

   Use the **[no]** form of the command to disable the feature.

The tri-radio feature is configured on all supporting access points, enabling automatic dual radio role management for optimized wireless performance.

Configure 5-GHz dual radio mode for AP (GUI)

Configure 5-GHz dual radio mode for AP using GUI.

Before you begin

Follow these steps to configure 5-GHz dual radio mode for AP using GUI:

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. On the **Access Points** page, click the **5 GHz Radios** section and select a Cisco 9130 Series AP from the list.

   The **Edit Radios 5 GHz Band** window is displayed.
3. In the **Edit Radios 5-GHz Band** > **Configure** > **General** tab, under **Dual Radio Mode**, select one from the following radio button options
   - Auto: Permits FRA to decide the mode for this AP.
   - Enabled: Enables Dual Radio mode for this AP.
   - Disabled: Disables Dual Radio mode for this AP.
4. Click **Update & Apply to Device**.

Configure the dual radio mode and enable slots (CLI)

Configure the dual radio mode and enable slots using commands.

Configure dual radio mode on Cisco APs to enable flexible radio operation and optimize wireless coverage.

Dual radio mode allows Cisco APs to operate with enhanced radio flexibility, supporting both dual and tri-radio configurations. This is useful for environments requiring optimized wireless performance and coverage management.

Procedure

1. Optional: Disable the 802.11a radio on Cisco AP.

   Example

   ```
   Device# ap name ap-name dot11 5ghz slot slot-number shutdown
   ```

   Example

   ```
   Device# ap name ap-name dot11 5ghz slot 1 shutdown
   ```

   Disables the 802.11a radio on Cisco AP.
2. Configure the 802.11a dual and tri-radio on the AP.

   Example

   ```
   Device# ap name ap-name dot11 5ghz slot 1 dual-radio mode mode-option
   ```

   Example

   ```
   Device# ap name ap-name dot11 5ghz slot 1 dual-radio mode enable
   ```

Configures the 802.11a dual and tri-radio on the AP. Enable auto to allow RRM to switch the AP between dual radio or tri radio mode based on the channel width configuration. In auto mode, the slot 2 state is managed by the RRM. Use the disable keyword to disable the dual-radio.

> **Note**
>
> When the AP is set to auto mode, the dual radio mode is disabled by default.

3. Enable the 802.11a radio on Cisco AP.

Example

```
Device# ap name ap-name no dot11 5ghz slot slot-number shutdown
```

Example

```
Device# ap name ap-name no dot11 5ghz slot 1 shutdown
```

Enables the 802.11a radio on Cisco AP.

The dual radio mode is configured and the specified slots are enabled, allowing the AP to operate with the configured radio settings.

Set radio roles for slots (CLI)

Configure radio roles for slots using commands.

Before you begin

Follow these steps to set radio roles for slots using CLI commands:

> **Note**
>
> Radio role assignment is supported only for access points operating in local mode. It is not supported for access points operating in bridge mode.

Procedure

---

1. **enable**

Example

```
Device> enable
```

Enables privileged EXEC mode.

Enter your password, if prompted.

2. **ap name** *ap-name* **dot11 24ghz 5ghz | 6ghz slot** *<slot ID>* **radio role auto manual monitor client-serving**

Example

```
Device# ap name ap-name dot11 5ghz slot 2 radio role manual monitor
```

Sets the radio role manual to either client serving or monitor.

Configure the tri-radio dual radio role (CLI)

Configure the tri-radio dual radio role using commands.

Set up tri-radio dual radio role configuration to optimize wireless coverage and monitoring capabilities.

Tri-radio access points support independent radio role configuration for each radio, allowing you to designate radios for client-serving or monitoring functions to optimize network performance and security.

Procedure

1. Configure the 802.11a radio role independently for each supporting AP's radio.

   Example

   ```
   Device# ap name ap-name dot11 5ghz slot slot-number radio role role-mode
   function
   ```

   Example

   ```
   Device# ap name 9130axtrial dot11 5ghz slot 1 radio role manual monitor
   ```

   The channel and the Tx power values can be configured when the radio role is set to manual mode.

2. Configure the 802.11b radio role independently for the supporting AP's radio.

   Example

   ```
   Device# ap name ap-name dot11 24ghz slot 0 radio role role-mode function
   ```

   Example

   ```
   Device# ap name 9130axtrial dot11 24ghz slot 0 radio role manual client-serving
   ```

The tri-radio dual radio roles are configured, with each radio optimized for its designated function of either client-serving or monitoring.

Tri-radio configuration verification on the controller

Lists commands to verify tri-radio configuration on the controller.

To verify that the dual radio mode is enabled, use the following **show** command:

```
Device# show ap name APXXXX.4XXX.04XX config slot 1 | inc Dual
 Dual Radio Capable                              : True
 Dual Radio Mode                                 : Enabled
 Dual Radio Operation mode                       : Auto
```

To verify the tri-radio status, use the following **show** command:

```
Device# show ap triradio status
Tri-Radio Status : Enabled
```

To verify that the radio role is set, use the following **show** command:

```
show ap name ap-name config slot <slot_number> | i Radio
Radio Type                                     : 802.11ax - 5 GHz
Radio Subband                                  : All
Radio Role                                     : Auto
Radio Mode                                     : Local
Radio SubType                                  : Main
```

60 Cisco Catalyst Center Assurance Wi-Fi 6 Dashboard

Topics:

- Cisco Catalyst Center assurance Wi-Fi 6 dashboard

The Cisco Catalyst Center Assurance Wi-Fi 6 Dashboard provides a visual overview of your wireless network's Wi-Fi 6 readiness and performance. It features dashlets that display client distribution by capability, AP readiness, protocol distribution, airtime efficiency, and wireless latency comparisons between Wi-Fi 6 and non-Wi-Fi 6 networks, helping to assess and optimize network efficiency and client experience. Configuration and verification procedures are available via CLI for debugging purposes, though management is recommended through the Cisco Catalyst Center.

Cisco Catalyst Center assurance Wi-Fi 6 dashboard

Explains the Cisco Catalyst Center Assurance Wi-Fi 6 Dashboard that provides a visual representation of your wireless network.

The Cisco Catalyst Center Assurance Wi-Fi 6 Dashboard is a visual interface that

- provides a comprehensive view of your wireless network performance and capabilities
- displays various dashlets showing Wi-Fi 6 readiness and network efficiency comparisons, and
- enables monitoring of Wi-Fi 6 networks compared to non-Wi-Fi 6 networks.

Dashboard components

> **Note**
>
> We recommend you manage this feature using the Cisco Catalyst Center UI. The procedures are to be executed with for debugging purposes only.

The dashboard contains the following dashlets:

- **Client Distribution by Capability**: This dashlet shows all the clients associated and their capability in the wireless network. The inner circle shows the wireless protocol capabilities of all the different clients in the network. Capability here is the ability of wireless clients to associate with Wi-Fi 6 APs or non-Wi-fi 6 APs. The outer arc segment shows how many 802.11ax capable clients are joined to a Wi-Fi 6 network as well as how many of them are not.
- **Wi-Fi 6 Network Readiness**: This dashlet shows all the APs in the network. The inner circle shows the APs which are Wi-Fi 6 APs and non Wi-Fi 6 APs. The outer arc segment shows the number of Wi-Fi 6 enabled AP in the network.
- **AP Distribution by Protocol**: This dashlet shows the protocols enabled on your APs in real time.
- **Wireless Airtime Efficiency**: This dashlet compares and displays the Airtime Efficiency between your Wi-Fi 6 network and Non-Wi-Fi 6 network for each of the access categories (voice, video, best effort, background).

  The spectrum is efficiently utilized if the AP's radios can send more traffic (successful bytes transmitted to the client) in less airtime (microseconds) than other networks under similar RF conditions.
- **Wireless Latency by Client Count**: This Dashlet compares the Wireless Latency between your Wi-Fi 6 and Non-Wi-Fi 6 Network for each of the access categories (voice, video, best effort, background).

  Wireless latency is measured by the time (microseconds) it takes for a packet to be successfully transmitted from an AP to the client. Hence, AP radios with a higher client count generally have higher latency than compared to those with a lower client count under similar RF conditions.

  > **Note**
  >
  > **Client count** in this dashlet refers to the clients that are actively sending traffic for a given Access Category and are not just associated clients.

For more information, see the **Monitor Wi-Fi 6 Readiness** section in the Cisco DNA Assurance User Guide.

Configure Cisco Catalyst Center assurance Wi-Fi 6 dashboard parameters (CLI)

Configure Cisco Catalyst Center Assurance Wi-Fi 6 Dashboard Parameters using commands.

Enable and configure traffic distribution statistics collection for Wi-Fi 6 dashboard monitoring in Cisco Catalyst Center Assurance.

The traffic distribution feature provides detailed analytics about wireless network performance, including signal strength distribution, airtime efficiency by access category and traffic type, and latency metrics. This data is essential for Wi-Fi 6 network optimization and troubleshooting through the Cisco Catalyst Center dashboard.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable configuration for all the APs that are associated with the specified AP profile name.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

   Example

   ```
   Device(config)# ap profile pp-1
   ```

3. Enable traffic distribution feature with the specified AP profile.

   Example

   ```
   Device(config-ap-profile)# statistics traffic-distribution
   ```

4. Configure the interval at which the AP sends the traffic distribution statistics.

   Example

   ```
   Device(config-ap-profile)# statistics traffic-distribution interval
   interval-secs
   ```

   Example

   ```
   Device(config-ap-profile)# statistics traffic-distribution interval 300
   ```

   Default value is 300 seconds. Valid range is between 30 and 3600 seconds.

> **Note**
>
> Execute this command only with the assistance from Cisco Technical Assistance Center (TAC) support engineer.

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-profile)# exit
   ```

6. Display traffic distribution data by signal strength, if received from the AP in the latest statistics update interval.

   Example

   ```
   Device# show wireless stats ap name ap-name traffic-distribution slot slot-id
    packet-count signal {average | good | poor} [last-received]
   ```

   Example

   ```
   Device# show wireless stats ap name ff123a traffic-distribution slot 1
   packet-count signal good
   ```

   Use **last-received** keyword to view the statistics received in any statistics update interval from the AP.

7. Display the Airtime efficiency data based on access category, if received from the AP in the latest statistics update interval.

   Example

   ```
   Device# show wireless stats ap name ap-name traffic-distribution slot slot-id
    airtime access-category {background | best-effort | video | voice}
   [last-received]
   ```

   Example

   ```
   Device# show wireless stats ap name ff123a traffic-distribution slot 1 airtime
    access-category best-effort
   ```

   Use **last-received** keyword to view the statistics received in any statistics update interval from the AP.

8. Display the Airtime efficiency data based on traffic type, if received from the AP in the latest statistics update interval.

   Example

   ```
   Device# show wireless stats ap name ap-name traffic-distribution slot slot-id
    airtime traffic-type {legacy | mu | ofdma | su} [last-received]
   ```

   Example

   ```
   Device# show wireless stats ap name ff123a traffic-distribution slot 1
   traffic-type ofdma
   ```

Use **last-received** keyword to view the statistics received in any statistics update interval from the AP.

9. Display wireless latency data based on access category, if received from the AP in the latest statistics update interval.

Example

```
Device# show wireless stats ap name ap-name traffic-distribution slot slot-id
 latency access-category {background | best-effort | video | voice}
[last-received]
```

Example

```
Device# show wireless stats ap name ff123a traffic-distribution slot 1 latency
 access-category best-effort
```

Use **last-received** keyword to view the statistics received in any statistics update interval from the AP.

Traffic distribution statistics are now enabled and configured for the specified AP profile, allowing Cisco Catalyst Center Assurance to collect and display Wi-Fi 6 dashboard analytics including signal strength distribution, airtime efficiency, and latency metrics.

AP DFS counter verification (CLI)

Lists commands to verify AP DFS counters using CLI commands.

Verify AP DFS counters using CLI commands for selected radio bands and specific access points.

DFS counter verification commands

- To verify the DFS counter for the selected radio band:

  show ap auto-rf dot11 24ghz | 5ghz | dual-band

```
Device#show ap auto-rf dot11 dual-band
```

- To verify the DFS counter for the selected radio band of a specific AP:

  show ap name *ap-name* **auto-rf dot11 24ghz | dual-band**

```
Device#show ap name ff32a auto-rf dot11 dual-band
```

- To verify the DFS counter for the selected 5-GHz slot of a specific AP:

  show ap name *ap-name* **auto-rf dot11 5ghz slot** *slot-id*

```
Device#show ap name ff32a auto-rf dot11 5ghz slot 1
```

Wi-Fi 6 access point parameter verification

Lists commands to verify Wi-Fi 6 access point parameters in the AP console and Cisco Catalyst 9800 Wireless Controller console.

Enter these commands in the AP console.

- To verify the traffic distribution statistics configuration, use the following command: **show ap traffic distribution configuration**

- To verify the exported data from the AP to the controller, use the following command: **show interfaces dot11Radio** *slot-id* **traffic distribution** {**cumulative** | **instantaneous** | **periodic**} **database**
- To verify Access Point DFS counters, use the following command: **show interfaces dot11radio** *slot-id***dfs**
- To debug the traffic distribution statistics, use the following command: {**no**} **debug traffic wireless distribution dump** {**periodic** | **aggregated**}
- To clear the traffic distribution dump, use the following command: **clear traffic distribution dump**

61 Antenna Disconnection Detection

Topics:

- Antenna disconnection detection

Antenna disconnection detection

Introduces antenna disconnection detection, detailing recommendations and limitations, operational mechanisms, configuration for CLI and GUI, SNMP trap detection processes, and verification procedures for confirming antenna status and setup integrity.

The Antenna Disconnection Detection feature monitors the signal strength across antennas on a receiver.

- Having multiple antennas on both transmitters and receivers enhances performance and reliability of APs
- Multiple antennas offer better reception by choosing the stronger signal or combining signals
- Detecting impaired antennas or physical damage is crucial for AP reliability

Whenever the signal strength delta goes beyond a set limit for a certain period, the antenna is likely experiencing issues.

For every detection time period that you configure, the AP sends an Inter-Access Point Protocol (IAPP) message that carries the antenna condition. This message is sent only once when the issue is detected and is displayed in the controller trap messages, SNMP traps, and controller debug logs.

Feature History

Release	Feature	Feature Information
Cisco IOS XE Bengaluru 17.4.1	Antenna Disconnection Detection	This feature detects the signal strength delta across the antennas on the receiver. If the delta is more than the defined limit for a specific duration, the corresponding antenna is considered to have issues.

Supported APs

The feature is supported only on these APs

- Cisco Catalyst 9120AX Series Access Points
- Cisco Catalyst 9130AX Series Access Points
- Cisco Aironet 2800e Access Points
- Cisco Aironet 3800e Access Points

Configuration workflow

1. Configure APs.
2. Configure an AP profile.
3. Enable the feature in AP profile.
4. Configure feature parameters.
5. Verify the configuration.

Limitations

Limitations of antenna disconnection detection for specific APs.

This document outlines limitations for specific APs.

- The SNMP trap is not supported on the Cisco Embedded Wireless Controller.
- The IAPP message is sent only when there is a change in the error condition.

How antenna disconnection detection works

Summary

The antenna disconnection detection process monitors signal strength across antennas and identifies malfunctioning antennas based on relative and absolute RSSI thresholds. It improves access point reliability by flagging weak or failed antennas early.

The key components involved in the process are

- Access point (AP): Compares RSSI values of all antennas and monitors for signal anomalies.
- RSSI thresholds: Include both relative (RSSI failure threshold) and absolute (weak RSSI) values.
- Detection timer: Sets the evaluation duration before confirming a disconnection.
- IAPP messaging: Sends a one-time alert when a disconnection is detected.

Workflow

The process involves these stages

1. Signal monitoring: The AP measures the received signal strength (RSSI) of all antennas.
2. Relative comparison: Each antenna's RSSI is compared to the antenna with the highest RSSI. If the delta exceeds the RSSI failure threshold, that antenna is marked as broken.
3. Absolute check: If all antennas have RSSI values below the weak RSSI threshold, they are all marked as malfunctioning, regardless of delta comparison.
4. Failure fallback: If neither condition is satisfied (e.g., threshold not met), the system defaults to reporting all antennas as malfunctioning.
5. Alert transmission: When a failure condition is confirmed, an IAPP message is generated once to notify monitoring systems via controller logs, SNMP traps, or debug logs.

Result

This process ensures reliable antenna monitoring by combining relative and absolute RSSI evaluations, resulting in faster detection of signal degradation and hardware faults.

Configure antenna disconnection detection

Configuring antenna disconnection detection using commands.

Enable antenna disconnection detection via CLI to monitor and alert on failing antennas.

This task applies to compatible Cisco access points. It uses RSSI thresholds and detection timers to identify and alert on signal anomalies that may indicate hardware failure.

Before you begin

Identify the AP profile to configure.

Procedure

1. Enter global configuration mode

Example

```
Device# configure terminal
```

Enters global configuration mode.

2. Configure an AP profile and enter AP profile configuration mode

Example

```
Device(config)# ap profile xyz-ap-profile
```

Configures an AP profile and enters AP profile configuration mode.

3. Enable antenna disconnection detection

Example

```
Device(config-ap-profile)# antenna monitoring
```

Enables antenna disconnection detection.

To disable antenna disconnection detection, use the **no antenna monitoring** command.

4. Configure RSSI failure threshold value

Example

```
Device(config-ap-profile)# antenna monitoring rssi-failure-threshold 20
```

Configures RSSI failure threshold value, in dB. Valid values range from 10 to 90, with a default of 40.

5. Configures weak RSSI value

Example

```
Device(config-ap-profile)# antenna monitoring weak-rssi -90
```

Configures weak RSSI value, in dBm. Valid values range from -90 to -10, with a default of 60.

6. Configure the antenna disconnection detection time

Example

```
Device(config-ap-profile)# antenna monitoring detection-time 20
```

Configures the antenna disconnection detection time, in minutes. Valid values range from 9 to 180, with a default of 120.

7. Return to privileged EXEC mode

Example

```
Device(config-ap-profile)# end
```

Saves the configuration and returns to privileged EXEC mode.

The network device actively monitors the antenna disconnection status and detect malfunctioning antennas based on configured RSSI thresholds. This enables timely alerts and corrective actions to maintain optimal network performance.

Configure antenna disconnection detection using GUI

Configure antenna disconnection detection using GUI.

Enable and configure antenna disconnection detection using the GUI.

Procedure

---

1. Choose **Configuration > Tags & Profiles > AP Join**.
2. In the **AP Join Profile** window, click the **General** tab.
3. Check the **Antenna Monitoring** check box to enable antenna monitoring.
4. In the **RSSI Fail Threshold(dB)** field, enter a value, in dB. The values range from 10 to 90 dB, with a default of 40.
5. In the **Weak RSSI(dBm)** field, enter a value, in dBm. Valid values range from -90 to -10 dBm, with a default of 60 dBm.
6. In the **Detection Time(min)** field, enter the antenna disconnection detection time, in minutes. Values range from 9 to 180 min, with a default of 120 min.
7. Click **Update & Apply to Device**.

The system applies the antenna disconnection detection settings to the device according to your specifications.

Detect broken antenna using SNMP trap (CLI)

Detect broken antenna using SNMP trap using commands.

Configure SNMP traps for antenna failures to enable rapid detection and response to hardware issues in APs.

Procedure

---

1. Enter global configuration mode

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. Enable all SNMP notification types

   Example

   ```
   Device(config)# snmp-server enable traps
   ```

   Enables all the SNMP notification types that are available on the system.

3. Configure SNMP trap for broken antenna detection

Example

```
Device(config)# trapflags ap broken-antenna
```

Enables an SNMP trap, which sends notifications when an antenna fails in any Cisco AP.

4. Return to privileged EXEC mode

Example

```
Device(config)# end
```

Returns to privileged EXEC mode.

SNMP trap notifications is sent when an antenna failure is detected, facilitating prompt maintenance actions.

Detect broken antenna using SNMP trap (GUI)

Detect broken antenna using SNMP trap using GUI.

Detect and report issues with antenna performance automatically using SNMP traps.

Procedure

---

1. Choose **Administration > Management > SNMP**.
2. Click the **Wireless Traps** tab.
3. Set the **Access Point** status as **Enabled**, if not done already.
4. Check the **Broken Antenna** check box to enable the trap.
5. Click **Apply**.

The system monitors antenna statuses and generates SNMP traps for any malfunctions, promptly notifying administrators to facilitate quick resolution and improve network reliability.

Verify antenna disconnection detection (Reference)

To verify the antenna disconnection detection feature configuration on an AP and AP profile.

You must ensure that the antenna disconnection detection feature is configured accurately on both individual access points and their profiles. This involves verifying specific settings like RSSI thresholds and detection times to maintain optimal network performance.

To verify the antenna disconnection detection feature configuration on an AP, use the following command:

```
Device# show ap name 3800-AP config general

Cisco AP Name: 3800-AP
================================================

Cisco AP Identifier                              : f4db.e632.df40
Country Code                                     : Multiple Countries :
US,IN,CN,CU
Regulatory Domain Allowed by Country             : 802.11bg:-ACE
802.11a:-ABCDHN
AP Country Code                                  : CN  - China
```

```
AP Regulatory Domain
  Slot 0                                          : -E
  Slot 1                                          : -C
MAC Address                                       : f4db.e62f.165a
IP Address Configuration                          : DHCP
IP Address                                        : 9.9.33.3
IP Netmask                                        : 255.255.255.0
Gateway IP Address                                : 9.9.33.1
Fallback IP Address Being Used                    :
Domain                                            :
Name Server                                       :
CAPWAP Path MTU                                   : 1485
Capwap Active Window Size                         : 1

.
.
.

AP broken antenna detection                                 : Enabled
RSSI threshold                                              : 40
Weak RSSI                                                   : -80
Detection Time                                              : 120

.
.
.
```

To verify the antenna disconnection detection feature configuration on an AP profile, use the following command:

```
Device# show ap profile name rf-profile-24g detailed

AP Profile Name: rf-profile-24g
.
.
.
AP broken antenna detection:
  Status                                                    : ENABLED
  RSSI threshold                                            : 40
  Weak RSSI                                                 : -80
  Detection Time                                            : 120
```

Verify antenna disconnection detection (GUI)

Verify antenna disconnection detection using GUI.

User can verify whether the antenna disconnection detection is supported by the access points, ensuring the system's monitoring capabilities are accurate and reliable.

Procedure

1. Choose **Monitoring > Wireless > AP Statistics**.
2. Click an AP name or anywhere on the row corresponding to an AP to activate the **General** window.
3. Click the **360 View** tab.

   The 360 View tab is the default selection. The **Antenna Monitoring** field indicates whether the AP supports monitoring or not.

Verify that the Antenna Monitoring field indicates support status. If supported, monitoring is accurate and reliable.

62 Neighbor Discovery Protocol Mode on Access Points

Topics:

- NDP mode
- Configuration Tasks

NDP mode

Introduces Neighbor Discovery Protocol Mode, explaining its purpose, operational principles, and significance in access point deployments within wireless networks.

Neighbor Discovery Protocol (NDP) modes are wireless protocol configurations in AP that

- determine whether NDP packets are transmitted **on the serving radio's configured channel (On-Channel)** or using the **legacy Off-Channel** method where the serving radio re-tunes to other channels to transmit NDP.
- allow selection through a controller setting (**Auto** or **Off-Channel**) to match specific deployment requirements.
- support mixed (brownfield) deployments where APs with and without scanning radios coexist, automatically choosing the appropriate mode for optimal performance.

Feature history

Feature Name	Release	Description
Neighbor Discovery Protocol Mode on Cisco Catalyst 9124AX Outdoor Access Points	Cisco IOS XE 17.5.1	From this release, supported APs can operate on both on-channel and off-channel NDP modes. The feature is configured per slot: • **ap dot11 {24ghz | 5ghz} rrm ndp-mode {auto | off-channel}** • **ndp-mode {auto | off-channel}**

Supported radios

- Cisco Aironet 3800 Series (Wi-Fi 5 with scanning radio; 2.4 GHz or 5 GHz only)
- Cisco Catalyst 9120 Series Access Points
- Cisco Catalyst 9130 Series Access Points
- Cisco Catalyst 9124 Series Access Points
- All Wi-Fi 6 and newer Catalyst APs

NDP transmission modes

In outdoor deployments – and in all modern Cisco Wi-Fi 6 and newer APs – NDP packets are transmitted based on the configured Neighbor Discovery Protocol (NDP) mode, which can be set through the controller to suit deployment requirements.

- **On-Channel**- The serving (client-facing) radio remains on its configured channel and continuously streams NDP packets (about every 20 ms) only on that channel. A *full-time scanning radio* (implemented through RF ASIC or equivalent) handles all receive operations across all channels.
- **Off-Channel (Legacy)** - Used when a scanning radio is not present. The serving radio periodically re-tunes off its serving channel to transmit NDP packets on other channels, cycling through all regulatory domain channels in about three minutes. This was the standard behavior in older APs before scanning radios became available.
- Auto Mode - **Default** setting. The controller automatically selects the transmission method:
  - If **all APs have scanning radios**: uses On-Channel mode for optimal performance.

- If **any AP lacks a scanning radio**: uses Off-Channel mode for compatibility.

You can verify which mode is in use with a packet capture on any serving channel:

- If **On-Channel** is in use, NDP packets appears on that channel at approximately 20 ms intervals as a continuous stream.
- If **Off-Channel** is in use, NDP packets appears on each channel only once during a full scan cycle (about every 3 minutes for all channels combined).

Operational guidelines

- In greenfield Wi-Fi 6+ deployments, which includes all modern APs, On-Channel mode provides the best performance and neighbor acquisition times.
- In brownfield (mixed) environments, Auto setting ensures the correct mode is chosen and maintains compatibility with legacy devices.
- 6 GHz deployments always operate in **On-Channel** mode (no legacy APs exist for this band).

> **Note**
>
> The performance gain by using **On-Channel** NDP is 25% more than the performance over legacy AP's in 5 and 6 GHz. Legacy APs are Cisco APs that do not have a full-time scanning radio (RF ASIC or equivalent) and do not support 6 GHz.

Configuration Tasks

Details configuration and monitoring procedures for NDP Mode, including step-by-step instructions for GUI and CLI setup, RF profile integration, monitoring radio statistics, and verification to ensure correct NDP operation on access points.

Configure RRM NDP mode (GUI)

Configure RRM Neighbor Discovery Mode using GUI.

Set the Radio Resource Management (RRM) neighbor discovery mode for APs. This optimizes wireless network performance.

Use this task to select how access points detect neighbors—either automatically or using off-channel scanning—through the GUI.

Before you begin

Confirm that the radio you want to configure (2.4 GHz or 5 GHz) is enabled.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **RRM**.
2. In the **Radio Resource Management** window, click either the **5 GHz Band** or the **2.4 GHz Band** tab.
3. In the **General** tab, under the **Noise/Interference/Rogue/CleanAir/SI Monitoring Channels** section, click the **RRM Neighbor Discovery Mode** toggle button to configure a mode.
4. Click **Apply**.

The RRM neighbor discovery mode is updated for the selected band.

Configure the NDP mode (CLI)

Configure the NDP Mode using commands.

Set the neighbor discovery protocol (NDP) mode for 802.11 radios using CLI.

Specify whether supported APs transmit NDP packets in auto or off-channel mode for 802.11a/b/g/n/ac radios.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the operating mode for 802.11a neighbor discovery using **ap dot11 { 24ghz | 5ghz } rrm ndp-mode {auto | off-channel}** command.

   Example
   For Brownfield (mixed) deployments:

   ```
   Device(config)# ap dot11 24ghz rrm ndp-mode auto
   ```

   ```
   Device(config)# ap dot11 5ghz rrm ndp-mode auto
   ```

   Example
   Greenfield (all APs with RF ASIC conventional radios)

   ```
   Device(config)# ap dot11 24ghz rrm ndp-mode off-channel
   ```

You have configured the NDP mode.

Configure the NDP Type (CLI)

Configure the Neighbor Discovery Protocol Type using commands.

Set which type of Neighbor Discovery Protocol (NDP) messages your AP broadcasts on the selected radio band: 2.4 GHz, 5 GHz, or 6 GHz.

Use this procedure to specify whether NDP messages are **protected** or **transparent** when configuring Radio Resource Management (RRM) parameters for an AP through the CLI.

Before you begin

- Determine which radio (2.4 GHz, 5 GHz, or 6 GHz) you intend to configure.
- Determine if you want your NDP messages to be **protected** or **transparent**.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the NDP type for 802.11a (2.4 GHz), 802.11b (5 GHz), or 802.11 (6 GHz) neighbor discovery using the **ap dot11 { 24ghz | 5ghz | 6ghz } rrm ndp-type { protected | transparent }** command.

   Example

   ```
   Device(config)# ap dot11 6ghz rrm ndp-type protected
   ```

   The two types are protected and transparent.

The AP uses the designated NDP type for neighbor discovery on the specified radio band.

Configure NDP mode in the RF profile (GUI)

Enable or modify Neighbor Discovery Protocol (NDP) mode settings for a specific RF profile to optimize wireless network performance.

NDP mode affects how devices discover neighboring APs within a wireless network RF profile. You can set NDP to **Auto** or **Off-Channel** as needed.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF**.
2. Click **Add**.
   The **Add RF Profile** window is displayed.
3. Click the **General** tab.
4. Click the **NDP Mode** toggle button to select the NDP mode as **AUTO** or as **OFF-CHANNEL**.
5. Click **Apply to Device**.

The chosen RF profile is updated with the selected NDP mode setting, which is now active in your wireless environment.

Configure NDP mode in the RF profile (CLI)

Configure Neighbor Discovery Protocol Mode in the RF Profile using commands.

Set the operating mode for Neighbor Discovery Protocol (NDP) in a specified RF profile using the CLI.

Use this task to configure the NDP operating mode—either **auto** or **off-channel**—for a specific RF profile (2.4 GHz, 5 GHz, or 6 GHz) on your wireless device.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enter the RF profile configuration using the **ap dot11 { 24ghz | 5ghz| 6ghz } rf-profile** *rf-profile-name* command.

Example

```
Device(config)# ap dot11 24ghz rf-profile rf-profile-name
```

3. Configure the operating mode for neighbor discovery.

Example

```
Device(config-rf-profile)# ndp-mode mode
```

The RF profile is updated to use the specified NDP mode on the selected radio band.

Monitor radio statistics and NDP using GUI

View and analyze radio statistics, including NDP capability and mode for APs.

Use this task when you need to check and evaluate the wireless radio status and NDP settings for access points in your network environment.

Note

You can verify which NDP transmission mode is in use with a packet capture on the serving channel:

- If the AP is using **On-Channel** mode, NDP frames will appear on the serving channel at approximately 20 ms intervals as a continuous stream.
- If the AP is using **Off-Channel (legacy)** mode, NDP frames will appear on each channel only once during a full scan cycle (about every 3 minutes for all channels combined). In this case, you may not see NDP frames on the serving channel for long periods while the serving radio is tuned elsewhere.

.

Procedure

1. Choose **Monitoring** > **Wireless** > **Radio Statistics**.
2. Click either **5 GHz Radios**, **2.4 GHz Radios**, or **Dual-Band Radios** tab.
   The corresponding radio band window displays the list of configured APs.
3. To view the general attributes of an AP, click the corresponding AP to display the **General** tab. The following information is displayed:
   - **AP Name**: Displays the assigned identifier for the AP, which is unique within the network. The AP name can be ASCII characters from 32 to 126, excluding leading and trailing spaces.
   - **IP Address**: Displays the IP address assigned to the AP in dotted-decimal format.

- **AP Mode**: Displays the configured AP mode. The supported modes are:
  - **Local**: It is the default mode, and it offers a basic service set (BSS) on a specific channel. When the AP does not transmit wireless client frame, it scans other channels to measure noise interference, discover rogue devices, and check for matches against Intrusion Detection System (IDS) events.
  - **Monitor**: An AP in monitor mode does not transmit. It is a dedicated sensor that checks IDS events, detects rogue APs, and determines the position of wireless stations.
  - **Sniffer**: You can configure an AP as a network sniffer using the controller. This mode captures and forwards all packets on a particular channel to a remote machine running packet analyzer software. These packets contain information on time stamps, signal strength, packet sizes, and so on. Sniffers allow you to monitor and record network activity and detect problems.
  - **Bridge**: The AP becomes a dedicated point-to-point or point-to-multipoint bridge. Two APs in bridge mode can connect two remote sites. Multiple APs can also form an indoor or outdoor mesh. Note that you cannot connect to the bridge with clients.
  - **Clear**: Returns the AP back to client-serving mode depending on the remote site tag configuration.
- **MAC Address**: Displays the registered MAC address on the controller.
- **Number of Slots** : Displays the number of slots supported by the AP.
- **Radio Type**: Displays the radio band configured on the controller. By default, both 802.11b/g/n (2.4-GHz) and 802.11a/n/ac (5-GHz) bands are enabled.
- **Slot ID**: Displays the slot on which radio is installed.
- **Sub band Type**: Displays the configured radio sub-band.
- **NDP Capability**: Displays the supported Neighbor Discovery Protocol (NDP) capability.
- **NDP Mode**: Displays the configured NDP mode. If the NDP mode that is configured is AUTO, the controller selects ON-Channel as the NDP mode. If the NDP-mode that is configured is OFF-Channel, the controller selects OFF-Channel as the NDP mode.

Verify NDP mode

Verify Neighbor Discovery Protocol Mode.

To verify the NDP mode, run these commands:

```
Device# show ap rf-profile name test-24g
Description                      : test
RF Profile Name                  : test-24g
Band                             : 2.4 GHz
Transmit Power Threshold v1      : -70 dBm
Min Transmit Power               : -10 dBm
Max Transmit Power               : 30 dBm
.
.
.
NDP mode                         : Auto
.
```

```
.
.
```

```
Device# show ap rf-profile name test-5g detail
Description                       : Test
RF Profile Name                   : test-5g
Band                              : 5 GHz
Transmit Power Threshold v1       : -70 dBm
Min Transmit Power                : -10 dBm
Max Transmit Power                : 30 dBm
.
.
.
NDP mode                          : Off-channel
.
.
.
```

```
Device# show ap name ap-name config dot11 24ghz
Cisco AP Identifier                              : 3cxx.0exx.36xx
Cisco AP Name                                    : Cisco-9105AXW-AP
Country Code                                     : Multiple Countries:
US,MK,J4,IN
Regulatory Domain Allowed by Country             : 802.11bg:-AEJPQU
802.11a:-ABDEIJNPQU
AP Country Code                                  : US  - United States
AP Regulatory Domain                             : -A
MAC Address                                      : 5cxx.0dxx.e0xx
IP Address Configuration                         : DHCP
.
.
.
NDP mode                                         :  Off-channel
.
.
.
```

```
Device# show ap name ap-name config dot11 5ghz
Cisco AP Identifier                              : 3cxx.0exx.36xx
Cisco AP Name                                    : Cisco-9105AXW-AP
Country Code                                     : Multiple Countries:
US,MK,J4,IN
Regulatory Domain Allowed by Country             : 802.11bg:-AEJPQU
802.11a:-ABDEIJNPQU
AP Country Code                                  : US  - United States
AP Regulatory Domain                             : -B
MAC Address                                      : 5cxx.0dxx.e0xx
IP Address Configuration                         : DHCP
IP Address                                       : Disabled
.
.
.
NDP mode                                         :  On-channel
.
.
.
```

```
Device# show ap dot11 24ghz monitor
Default 802.11b AP monitoring
  802.11b Monitor Mode                : Enabled
```

```
  802.11b Monitor Channels             : Country channels
  802.11b RRM Neighbor Discover Type   : Transparent
  802.11b AP Coverage Interval         : 180 seconds
  802.11b AP Load Interval             : 60 seconds
  802.11b AP Measurement Interval      : 180 seconds
  802.11b AP Reporting Interval        : 180 seconds
  802.11b NDP RSSI Normalization       : Enabled
  802.11b Neighbor Timeout factor      : 20
  802.11b NDP mode                     : Auto
```

```
Device# show ap dot11 5ghz monitor
Default 802.11a AP monitoring
  802.11a Monitor Mode                 : Enabled
  802.11a Monitor Channels             : Country channels
  802.11a RRM Neighbor Discover Type   : Transparent
  802.11a AP Coverage Interval         : 180 seconds
  802.11a AP Load Interval             : 60 seconds
  802.11a AP Measurement Interval      : 180 seconds
  802.11a AP Reporting Interval        : 180 seconds
  802.11a NDP RSSI Normalization       : Enabled
  802.11a Neighbor Timeout factor      : 20
  802.11a NDP mode                     : Auto
```

63 6-GHz Band Operations

Topics:

- 6-GHz Band Support
- 6-GHz client steering

The 6-GHz band operations section introduces features and considerations unique to wireless communication on the 6-GHz radio band, enhancing connectivity and performance in modern Wi-Fi networks.

These topics describe the features that are specific to 6-GHz band radio.

6-GHz Band Support

The 6-GHz band operations section introduces features and considerations unique to wireless communication on the 6-GHz radio band, enhancing connectivity and performance in modern Wi-Fi networks.

Feature history

This table provides release and related information for the feature explained in this module.

Table 74: Feature history for 6 GHz band operations

Feature name	Release information	Feature description
6 GHz radio band support in Cisco Catalyst 9136I APs	Cisco IOS XE Cupertino 17.7.1	Cisco Catalyst 1936I Access Points support the 6 GHz radio band. This topic describes the 6 GHz band operations.
6 GHz client steering support in Cisco Catalyst 9136I APs	Cisco IOS XE Cupertino 17.7.1	6 GHz client steering occurs when the controller receives a periodic client statistics report from a 2.4 GHz band or the 5 GHz band. The client-steering configuration enabled under WLAN applies only to 6 GHz capable clients. Client steering is triggered if a client in the report is 6 GHz capable, redirecting the client to the 6 GHz band.

Configure preferred scanning channels in the RF profile (GUI)

Configure preferred scanning channels in the RF profile using GUI.

Complete this task to configure the preferred scanning channels in the RF profile in the WebUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF/Radio**.
2. In the **RF** tab, click **Add**.
   The **Add RF Profile** page is displayed.
3. Click the **RRM** tab.
4. Click the **DCA** tab.
5. In the **Dynamic Channel Assignment** section, select the required channels in **DCA Channels** section.
6. In the **PSC Bias** field, click the toggle button to enable the preferred scanning channel bias for DCA.
7. Click **Apply to Device**.

The preferred scanning channels in the RF profile are configured.

Configure preferred scanning channels in RF profile (CLI)

Configure Preferred Scanning Channels in the RF Profile using commands.

Complete this task to configure preferred scanning channels in the RF profile.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an RF profile and enter RF profile configuration mode.

   Example

   ```
   Device(config)# ap dot11 6ghz rf-profile rf-profile-name
   ```

3. Configure the RF Profile DCA settings and enable the bias for the preferred scanning channel for DCA.

   Example

   ```
   Device(config-rf-profile)# channel psc
   ```

The preferred scanning channels in the RF profile are configured.

Example

```
Device# configure terminal
Device(config)# ap dot11 6ghz rf-profile rf-profile-name
Device(config-rf-profile)# channel psc
```

Configure broadcast probe response in RF profile (GUI)

Configure Broadcast Probe Response in RF Profile using GUI.

Complete this task to configure broadcast probe response in the RF profile.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF/Radio**.
2. In the **RF** tab, click **Add**.
   The **Add RF Profile** page is displayed.
3. Click **802.11ax**.
4. In the **6 GHz Discovery Frames** section, click **Broadcast Probe Response**.
5. In the **Broadcast Probe Response Interval** field, enter the response time interval in milli-seconds (ms). The range is 5 ms to 25 ms. The default value is 20 ms.
6. Click **Apply to Device**.

The broadcast probe response in the RF profile is configured.

Configure broadcast probe response in RF profile (CLI)

Configure Broadcast Probe Response in RF Profile using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an RF profile and enter the RF profile configuration mode.

   Example

   ```
   Device(config)# ap dot11 6ghz rf-profile rf-profile-name
   ```

3. Configure the broadcast probe response.

   Example

   ```
   Device(config-rf-profile)# dot11ax bcast-probe-response
   ```

4. Configure the broadcast probe response interval.

   Example

   ```
   Device(config-rf-profile)# dot11ax bcast-probe-response time-interval
   time-interval
   ```

The RF profile configuration includes the broadcast probe response.

Example

```
Device# configure terminal
Device(config)# ap dot11 6ghz rf-profile rf-profile-name
Device(config-rf-profile)# dot11ax bcast-probe-response
Device(config-rf-profile)# dot11ax bcast-probe-response time-interval 20
```

Configure FILS discovery frames in RF profile (GUI)

Configure FILS Discovery Frames in the RF Profile using GUI.

Complete this task to configure FILS discovery frames in the RF profile.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF/Radio**.
2. In the **RF** tab, click **Add**.
   The **Add RF Profile** page is displayed.
3. Click **802.11ax**.

4. In the **6 GHz Discovery Frames** section, click **FILS Discovery**.

> **Note**
>
> To prevent the transmission of discovery FILS frames when the discovery frames are set to **None** in the RF profile, ensure that you disable FILS discovery frames by either switching to the 5 GHz or the 2.4 GHz bands on the AP or by selecting the Broadcast Probe Response option.

5. Click **Apply to Device**.

The FILS discovery frames are configured in the RF profile.

Configure FILS discovery frames in RF profile (CLI)

Configure FILS Discovery Frames in the RF Profile using commands.

Complete this task to configure FILS discovery frames in the RF profile.

Procedure

---

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure an RF profile and enter the RF profile configuration mode.

Example

```
Device(config)# ap dot11 6ghz rf-profile rf-profile-name
```

3. Configure the 802.11ax FILS discovery.

Example

```
Device(config-rf-profile)# dot11ax fils-discovery
```

> **Note**
>
> To prevent the transmission of discovery FILS frames when the discovery frames are set to **none** in the RF profile, disable FILS discovery frames by switching to the 5 GHz or 2.4 GHz bands on the AP, or by changing to Broadcast Probe Response.

The configuration of FILS discovery frames in the RF profile has been completed.

Example

```
Device# configure terminal
Device(config)# ap dot11 6ghz rf-profile rf-profile-name
Device(config-rf-profile)# dot11ax fils-discovery
```

Configure multi BSSID profile (GUI)

Configure Multi BSSID Profile using GUI.

Complete this task by configuring the multi BSSID profile.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Multi BSSID**.
2. Click **Add**.
   The **Add Multi BSSID Profile** page is displayed.
3. Enter the name and the description of the BSSID profile.
4. Enter the following 802.11ax parameters:
   a) **Downlink OFDMA**
   b) **Uplink OFDMA**
   c) **Downlink MU-MIMO**
   d) **Uplink MU-MIMO**
   e) **Target Waketime**
   f) **TWT Broadcast Support**
5. Click **Apply to Device**.

The multi BSSID profile is configured.

Configure multi BSSID profile (CLI)

Configure Multi BSSID Profile using commands.

Complete this task to configure the multi BSSID profile for 6 GHz band radio.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the multi BSSID profile and enter the multi BSSID profile configuration.

   Example

   ```
   Device (config)# wireless profile multi-bssid multi-bssid-profile-name
   ```

3. Configure the 802.11ax parameters.

Example

```
Device (config-wireless-multi-bssid-profile)# dot11ax {downlink-mumimo |
downlink-ofdma | target-waketime  | twt-broadcast | uplink-mumimo |
uplink-ofdma}
```

The multi BSSID profile for 6 Ghz band radio is configured.

Example

```
Device# configure terminal
Device (config)# wireless profile multi-bssid multi-bssid-profile-name
Device (config-wireless-multi-bssid-profile)# dot11ax downlink-mumimo
```

Configure multi BSSID in RF profile (GUI)

Configure Multi-BSSID in the RF Profile using GUI.

Complete this task to configure multi BSSID in the RF profile.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **RF/Radio**.
2. In the **RF** tab, click **Add**.
   The **Add RF Profile** page is displayed.
3. Click the **802.11ax** tab.
4. In the **Multi BSSID Profile** field, choose the profile from the drop-down list.
5. Click **Apply to Device**.

Multi BSSID is configured in the RF profile.

Configure multi BSSID in RF profile (CLI)

Configure Multi BSSID in the RF Profile using commands.

Complete this task to configure multi BSSID in the RF profile.

Procedure

---

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an RF profile and enter RF profile configuration mode.

   Example

   ```
   Device(config)# ap dot11 6ghz rf-profile rf-profile-name
   ```

3. Configure 802.11ax multi BSSID profile name in the RF profile configuration mode.

Example

```
Device(config-rf-profile)# dot11ax multi-bssid-profile multi-bssid-profile-name
```

The multi BSSID configuration in the RF profile is completed.

Example

```
Device# configure terminal
Device(config)# ap dot11 6ghz rf-profile rf-profile-name
Device(config-rf-profile)# dot11ax multi-bssid-profile multi-bssid-profile-name
```

Configure dynamic channel assignment freeze (CLI)

Configure Dynamic Channel Assignment Freeze using commands.

Complete this task to disable DCA for 6 GHz after the 6 GHz radios are stabilized with the right channels.

Before you begin
Ensure that Dynamic Channel Assignment (DCA) for 6 GHz is enabled. Wait for the 6 GHz radios to get stabilized with the right set of channel assignments.

Procedure

Disable DCA for 6 GHz bands.

Example

```
Device# no ap dot11 6ghz rrm channel dca global auto
```

DCA for 6 GHz radios is disabled.

6-GHz client steering

6-GHz Client Steering details.

6-GHz client steering is a network management feature that

- enables the steering of clients to the 6-GHz band based on periodic client statistics reports from the 2.4-GHz or 5-GHz bands
- is configured under WLAN settings, and
- applies only to clients that are 6-GHz capable.

The 6-GHz band offers additional channels, increased bandwidth, and reduced network congestion compared to the current 2.4-GHz and 5-GHz bands. Consequently, wireless clients with 6-GHz capability connect to the 6-GHz radio to leverage these advantages.

> **Note**
>
> Client steering to 6-GHz will not work if the SSID is used as OWE transition mode.

Configure 6-GHz client steering in global configuration mode (GUI)

Configure 6-GHz Client Steering in the Global Configuration Mode using GUI.

Complete this task to configure 6-GHz client steering in global configuration mode.

Procedure

1. Choose **Configuration** > **Wireless** > **Advanced**.
2. Click the **6 GHz Client Steering** tab. Client steering is configurable per WLAN.
3. In the **6 GHz Transition Minimum Client Count** field, enter a value to set the minimum number of clients for client steering. The default value is three clients. The range is from 0 to 200 clients.
4. In the **6 GHz Transition Minimum Window Size** field, enter a value to set the minimum window size of client steering. The default value is three. The range is from 0 to 200.
5. In the **6 GHz Transition Maximum Utilization Difference** field, enter a value to set the maximum utilization difference for steering. The range is from 0 to 100 percent. The default value is 20.
6. In the **6 GHz Transition Minimum 2.4 GHz RSSI Threshold** field, enter a value to set the minimum value for client steering 2.4-GHz RSSI threshold.
7. In the **6 GHz Transition Minimum 5 GHz RSSI Threshold** field, enter a value to set the minimum value for client steering 5-GHz RSSI threshold.
8. Click **Apply**.

The 6-GHz client steering is configured in the global configuration mode.

Configure 6-GHz client steering in global configuration mode (CLI)

Configure 6-GHz Client Steering in the Global Configuration Mode using commands.

Complete this task to configure 6-GHz client steering in the global configuration mode.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Set the minimum number of clients for client steering.

   Example

   ```
   Device(config)# client-steering client-count min-num-clients
   ```

The range is from 0 to 200.

3. Set the minimum window size of client steering.

Example

```
Device(config)# client-steering window-size window-size
```

The range is from 0 to 200.

4. Set the maximum channel utilization difference (2.4-GHz or 5-GHz to 6-GHz) for steering.

Example

```
Device(config)# wireless client client-steering util-threshold threshold
```

The range is from 0 to 100 percent.

5. Set the minimum value for the 2.4-GHz RSSI threshold for client steering.

Example

```
Device(config)# wireless client client-steering min-rssi-24ghz -70
```

6. Set the minimum value for the 5-GHz RSSI threshold for client steering.

Example

```
Device(config)# wireless client client-steering min-rssi-5ghz -75
```

The 6-GHz client steering is configured in the global configuration mode.

Example

```
Device# configure terminal
Device(config)# client-steering client-count 3
Device(config)# client-steering window-size 5
Device(config)# wireless client client-steering util-threshold 25
Device(config)# wireless client client-steering min-rssi-24ghz -70
Device(config)# wireless client client-steering min-rssi-5ghz -75
```

Configure 6-GHz client steering on WLAN (GUI)

Configure 6-GHz Client Steering on the WLAN using GUI.

Complete this task to configure 6 GHz client steering on WLAN.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
   The **Add WLAN** page is displayed.
3. Click **Advanced**.
4. Check the **6 GHz Client Steering** check box to enable client steering on the WLAN.

5. Click **Apply to Device**.

The 6 GHz client steering on WLAN is configured.

Configure 6-GHz client steering on WLAN (CLI)

Configure 6-GHz Client Steering on the WLAN using commands.

Complete this task to configure 6 GHz client steering on WLAN.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration submode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id ssid-name
   ```

3. Configure 6-GHz client steering on the WLAN.

   Example

   ```
   Device(config-wlan)# client-steering
   ```

The 6-GHz client steering on WLAN is configured.

Example

```
Device# configure terminal
Device(config)# wlan wlan-name 18 ssid-name
Device(config-wlan)# client-steering
```

Verify 6-GHz client steering

Verify 6-GHz Client Steering.

To verify client steering, run the following commands:

```
Device# show wlan wlan-id
                WLAN Profile Name     : wlan1
                ================================================
                Identifier                                    : 1
                Description                                   :
                Network Name (SSID)                           : ssid-demo
                Status                                        : Disabled
                Broadcast SSID                                : Enabled
```

```
                .
                .
                .
                6Ghz Client Steering                              : Enabled
                .
                .
                .
```

```
Device# show wireless client steering
                Client Steering Configuration Information
                Macro to micro transition threshold                 : -55 dBm

                Micro to Macro transition threshold                 : -65 dBm

                Micro-Macro transition minimum client count          : 3
                Micro-Macro transition client balancing window      : 3
                Probe suppression mode                             : Disabled

                Probe suppression transition aggressiveness         : 3
                Probe suppression hysteresis                        : -6 dB
                6Ghz transition minimum client count                : 3
                6Ghz transition minimum window size                 : 3
                6Ghz transition maximum channel util difference     : 20%
                6Ghz transition minimum 2.4Ghz RSSI threshold       : -60 dBm

                6Ghz transition minimum 5Ghz RSSI threshold         : -65 dBm


                WLAN Configuration Information

                WLAN Profile Name                     11k Neighbor Report
   11v BSS Transition

--------------------------------------------------------------------------------

                12   test1                            Enabled
   Enabled
                8    test                             Enabled
   Enabled
```

Part VII

Network Management

Topics:

- AP Packet Capture
- DHCP Option82
- RADIUS Authentication and Accounting
- Cisco StadiumVision
- Persistent SSID Broadcast
- Network Monitoring
- Creating a Lobby Ambassador Account
- Lobby Ambassador Account
- Guest User Accounts
- Link Local Bridging
- Web Admin Settings
- Web UI Configuration Command Accounting in TACACS Server
- Embedded Packet Capture
- Layer 3 Access
- IPv6 Layer 3 Access

64 AP Packet Capture

Topics:

- AP client packet capture

AP client packet capture

Introduces the AP client packet capture feature, outlining its use for wireless client troubleshooting by capturing and filtering packets on APs, uploading captures to an FTP server, and supporting network engineers in analyzing over-the-air traffic to diagnose connectivity or performance issues.

An AP client packet capture is a diagnostic feature that

- enables packet capture on APs for the purpose of wireless client troubleshooting
- filters and captures packets on the channel and radio where the AP is operating, and
- uploads the captured packets to a file on an FTP server for later analysis (for example, in Wireshark).

Feature history

Table 75: Feature history for AP client packet capture

Feature name	Release information	Feature description
AP client packet capture	Cisco IOS XE 16.7.1	The AP client packet capture feature enables packet capture on APs, filters and captures packets on the channels and radios, and uploads the captured packets to a file.

This feature allows network engineers to analyze over-the-air traffic as observed by the AP, helping identify connectivity or performance issues experienced by wireless clients. For instance, when debugging why a client device cannot connect to a Wi-Fi network, engineers can use AP client packet capture to record and examine all packets exchanged between the client and the AP.

Limitations of AP client packet capture

The limitations of AP client packet capture include:

- You can perform packet capture for only one client at a time per site.
- You can start packet capture on a specific AP or a set of APs using static mode.

  When you start packet capture in auto mode, the system selects nearby APs to capture packets for a specific client. In auto mode, you cannot start or stop packet capture on an individual AP. Use the **stop all** command to stop packet capture when it is started in auto mode.
- After the Single Sign-On (SSO) process is complete, the packet capture stops after a switchover.

Enable packet capture (GUI)

Enable Packet Capture using GUI.

Enable packet capture on selected network interfaces.

Procedure

1. Choose **Troubleshooting** > **Packet Capture**.
2. Click **Add**. Enter the name that identifies the capture point.

3. From the **Filter** drop-down list, choose if you want to capture IPv4 or IPv6 packets. For IPv4, provide the source and destination addresses and the subnet mask. For IPv6, enter the source and destination addresses and the prefix length.
4. Check the **Monitor Control Plane** check box to configure the control plane as an attachment point.
5. Check the **DHCP** check box to enable **Inner Filter Protocol**.
6. Enter the **Inner Filter MAC address** using one of the standard MAC address formats: xx:xx:xx:xx:xx:xx, xx-xx-xx-xx-xx-xx, or xxxx.xxxx.xxxx.
7. Type the capture buffer size. The default buffer is 10 MB.
8. Specify the session limit either in seconds or in the number of packets captured.
9. Select the interfaces from which you want to capture packets, in the **Available** list. Click the arrow button to move it to the **Selected** list.
10. Click **Apply to Device**.

Packet capture is active on the chosen interfaces, and captured packet data is available for analysis.

Enable packet capture (CLI)

Enable Packet Capture using commands.

Enable packet capture for a specified client.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Enable packet capture for the specified client on a set of nearby APs.

   Example

   ```
   Device# ap packet-capture start client-mac-address auto
   ```

Packet capture is now enabled on your specified client.

```
Device# enable
Device# ap packet-capture start 0011.0011.0011 auto
```

Create AP packet capture profile and map to an AP join profile (GUI)

Create AP Packet Capture Profile and Map to an AP Join Profile using GUI.

Add a packet capture profile to an AP join profile so you can monitor APs.

Procedure

1. Click **Configuration** > **Tags & Profiles** > **AP Join Profile**.

2. Click **Add** to create a new AP Join Profile and enter the required details.
3. In the **Add AP Join Profile** area, click **AP** > **Packet Capture**.
4. Click the **Plus** icon to create a new Packet Capture profile or select a profile from the dropdown menu.
5. Click **Save**.

The AP join profile now includes a packet capture profile. This change enables packet capture for associated APs.

Create AP packet capture profile and map to an AP join profile

Create AP Packet Capture Profile and Map to an AP Join Profile using commands.

Create and enable an AP packet capture profile, then associate it with a specific AP join profile to facilitate packet capture for targeted APs.

Packet capture profile configurations apply to individual APs. The packet capture profile is assigned to an access point profile, which is then mapped to a site tag.

When starting packet capture, APs apply the packet capture profile configurations associated with their site and AP join profile.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile.

   Example

   ```
   Device(config)# wireless profile ap packet-capture packet-capture-profile-name
   ```

3. Configure an AP packet capture profile.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

4. Enable packet capture in the AP profile.

   Example

   ```
   Device(config-ap-profile)# packet-capture profile-name
   ```

5. Exit the AP profile configuration mode.

   Example

   ```
   Device(config-ap-profile)# end
   ```

6. Check the detailed information for the selected AP packet capture profile.

Example

```
Device# show wireless profile ap packet-capture detailed profile-name
```

The AP packet capture profile is successfully created and mapped to the selected AP join profile.

```
Device# configure terminal
Device(config)# wireless profile ap packet-capture test1
Device(config)# ap profile ap-profile1
Device(config-ap-profile)# packet-capture capture-profile
Device(config-ap-profile)# end
Device# show wireless profile ap packet-capture detailed test1
```

Start or stop packet capture

Start or Stop Packet Capture using commands.

You can start or stop wireless packet capture for a specific client using CLI commands.

Procedure

1. Enable packet capture for a specific client.

   Example

   ```
   Device# ap packet-capture stop client-mac-address {auto | static | ap-name}
   ```

2. Disable packet capture for a specific client.

   Example

   ```
   Device# ap packet-capture stop client-mac-address {all | static | ap-name}
   ```

The system starts or stops packet capture for the selected client on the specified AP.

```
Device# ap packet-capture start 0011.0011.0011 auto
Device# ap packet-capture stop 0011.0011.0011 all
```

65 DHCP Option82

Topics:

- DHCP Option 82
- Configure DHCP Option 82 global interface
- Configure DHCP Option 82 through a VLAN interface
- AP DHCP Option 82 support on FlexConnect local switching mode

DHCP Option 82

Outlines the functions and security benefits of DHCP Option 82, describes its suboptions and relay agent information, and details controller configuration options for including network-specific identifiers and enabling address allocation based on topology or client characteristics.

DHCP Option 82 is a relay agent information option that

- carries information known by the relay agent
- increases DHCP security by enabling identification of request sources, and
- allows address allocation based on network topology or client device characteristics.

The controller can be configured to add Option 82 information to DHCP requests from clients before forwarding the requests to a DHCP server. The DHCP server can then be configured to allocate IP addresses to the wireless client based on the information present in DHCP Option 82.

DHCP provides a framework for passing configuration information to hosts on a TCP/IP network. Configuration parameters and other control information are carried in tagged data items stored in the Options field of the DHCP message. These data items are also called options. Option 82 includes information known by the relay agent.

The Relay Agent Information option is a single DHCP option containing one or more suboptions that convey information known by the Relay Agent. Option 82 enables a DHCP Relay Agent to insert circuit-specific information into a forwarded request to a DHCP server. This option uses two suboptions:

- Circuit ID: The Circuit ID suboption includes information that is specific to the circuit the request came in on. This suboption is an identifier that is specific to the relay agent. Thus, the circuit that is described will vary depending on the relay agent.
- Remote ID: The Remote ID suboption includes information on the remote host-end of the circuit. This suboption usually contains information that identifies the relay agent. In a wireless network, this is likely a unique identifier for the wireless AP.

Note

- All valid Remote ID combinations are separated by a colon (:) as the delimiter.
- IP MAC binding is required for DHCP Option 82 to work in some situations.

DHCP Option 82 configuration options

You can configure these DHCP Option 82 options in a controller:

- DHCP Enable
- DHCP Opt82 Enable
- DHCP Opt82 Ascii
- DHCP Opt82 RID
- DHCP Opt Format
- DHCP AP MAC
- DHCP SSID

- DHCP AP ETH MAC
- DHCP AP NAME
- DHCP Site Tag
- DHCP AP Location
- DHCP VLAN ID

The controller includes the SSID in ASCII format and the VLAN ID in hexadecimal format within the remote ID suboption of Option 82 in the outgoing DHCP packets to the server for this configuration.

```
ipv4 dhcp opt82 format ssid
ipv4 dhcp opt82 format vlan-id
```

However, if *ipv4 dhcp opt82 ascii* configuration is also present, the controller adds VLAN-ID and SSID in ASCII format.

For Cisco Catalyst 9800 Series Configuration Best Practices, see the following link: https://www.cisco.com/c/en/us/products/collateral/wireless/catalyst-9800-series-wireless-controllers/guide-c07-743627.html

Configure DHCP Option 82 format

Configure DHCP Option 82 Format using commands.

Configure the wireless controller to insert DHCP Option 82 information into DHCP requests from wireless clients.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable configuration for the specified profile policy.

   Example

   ```
   Device(config)# wireless profile policy policy-name
   ```

3. Shut down the profile policy.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

4. Assign the profile policy to a VLAN.

   Example

   ```
   Device(config-wireless-policy)# vlan vlan-name
   ```

5. Optional: Set the session timeout value in seconds. The range is between 20 to 86,400 seconds.

Example

```
Device(config-wireless-policy)# session-timeout value-btwn-20-86400
```

6. Optional: Set the idle timeout value in seconds. The range is between 15 to 100,000 seconds.

   Example

   ```
   Device(config-wireless-policy)# idle-timeout 15
   ```

7. Enable central switching.

   Example

   ```
   Device(config-wireless-policy)# central switching
   ```

8. Enable DHCP Option 82 for the wireless clients.

   Example

   ```
   Device(config-wireless-policy)# ipv4 dhcp opt82
   ```

9. Optional: Enable ASCII on the DHCP Option 82 feature.

   Example

   ```
   Device(config-wireless-policy)# ipv4 dhcp opt82 ascii
   ```

10. Optional: Add Cisco 2-byte Remote ID (RID) to the DHCP Option 82 feature.

    Example

    ```
    Device(config-wireless-policy)# ipv4 dhcp opt82 rid
    ```

11. Enable DHCP Option 82 on the corresponding AP.

    Example

    ```
    Device(config-wireless-policy)# ipv4 dhcp opt82 format {ap_ethmac | ap_location
     | apmac | apname | policy_tag | ssid | vlan_id}
    ```

    For information on the various options available with the command, see Cisco Catalyst 9800 Series Wireless Controller Command Reference .

12. Enable the profile policy.

    Example

    ```
    Device(config-wireless-policy)# no shutdown
    ```

The wireless profile policy inserts DHCP Option 82 information in the specified format for wireless clients.

```
Device# configure terminal
Device(config)# wireless profile policy pp3
Device(config-wireless-policy)# shutdown
Device(config-wireless-policy)# vlan 72
Device(config-wireless-policy)# session-timeout 300
Device(config-wireless-policy)# idle-timeout 15
Device(config-wireless-policy)# central switching
Device(config-wireless-policy)# ipv4 dhcp opt82
Device(config-wireless-policy)# ipv4 dhcp opt82 ascii
Device(config-wireless-policy)# ipv4 dhcp opt82 rid
Device(config-wireless-policy)# ipv4 dhcp opt82 format apmac
Device(config-wireless-policy)# no shutdown
```

Configure DHCP Option 82 global interface

Outlines global DHCP Option 82 configuration using server override settings and different SVIs, including relay agent information setup, link selection suboption enablement, and use of specified source interfaces for DHCP message relaying.

Configure DHCP Option 82 globally through server override (CLI)

Configure DHCP Option 82 Globally Through Server Override using commands.

Set up DHCP Option 82 settings to override relay agent information in DHCP packets at the server.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Insert the global server override and link selection suboptions.

   Example

   ```
   Device(config)# ip dhcp-relay information option server-override
   ```

The system processes DHCP Option 82 information globally based on server settings.

```
Device# configure terminal
Device(config)# ip dhcp-relay information option server-override
```

Configure DHCP Option 82 through server override (CLI)

Configure DHCP Option 82 Through Server Override using commands.

To enable precise DHCP Option 82 behavior, configure the server override and link-selection suboptions.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Set the server override suboption to either an RFC-specific value or a proprietary value.

   Example

   ```
   Device(config)# ip dhcp compatibility suboption server-override [cisco |
   standard]
   ```

3. Set the link selection suboption to either an RFC-specific value or a proprietary value.

   Example

   ```
   Device(config)# ip dhcp compatibility suboption link-selection [cisco |
   standard]
   ```

DHCP Option 82 operates according to the configured settings.

```
Device# configure terminal
Device(config)# ip dhcp compatibility suboption server-override cisco
Device(config)# ip dhcp compatibility suboption link-selection cisco
```

Configure DHCP Option 82 globally through different SVIs (GUI)

Configure DHCP Option 82 Globally Through Different SVIs using GUI.

Enable DHCP Option 82 to insert relay information for clients across selected SVIs.

Procedure

1. Choose **Configuration > VLAN**.
2. Select a VLAN from the drop-down list.

   The **Edit SVI** window appears.

3. Click the **Advanced** tab.
4. Select an option from the **IPv4 Inbound ACL** drop-down list.
5. Select an option from the **IPv4 Outbound ACL** drop-down list.
6. Select an option from the **IPv6 Inbound ACL** drop-down list.
7. Select an option from the **IPv6 Outbound ACL** drop-down list.
8. Enter an IP address in the **IPv4 Helper Address** field.
9. Set the status to **Enabled** if you want to enable the **Relay Information Option** setting.
10. Enter the **Subscriber ID**.
11. Set the status to **Enabled** if you want to enable the **Server ID Override** setting.
12. Set the status to **Enabled** if you want to enable the **Option Insert** setting.

13. Select an option from the **Source-Interface Vlan** drop-down list.
14. Click **Update & Apply to Device**.

DHCP Option 82 is enabled for the selected SVIs. This allows relay agent information to be included in DHCP requests.

Configure DHCP Option 82 globally through different SVIs (CLI)

Configure DHCP Option 82 Globally Through Different SVIs using commands.

Enable DHCP Option 82 globally so that DHCP relayed messages include option 82 information through the specified SVIs.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Set global source interface for relayed messages.

   Example

   ```
   Device(config)# ip dhcp-relay source-interface vlan vlan-id
   ```

The device globally applies DHCP relay with Option 82 information using the specified SVI interface.

```
Device# configure terminal
Device(config)# ip dhcp-relay source-interface vlan 72
```

Configure DHCP Option 82 through a VLAN interface

Outlines procedures for configuring DHCP Option 82 on VLAN interfaces and SVIs using commands such as option-insert, server-ID-override, and subscriber-ID, including steps for VLAN and SVI setup, suboption configuration, IP assignment, UDP relay settings, and disabling MOP features for DHCP relay control.

Configure DHCP Option 82 through option-insert command (CLI)

Configure DHCP Option 82 Through Option-Insert Command using commands.

Enable DHCP Option 82 insertion on a VLAN interface to provide relay agent information in DHCP packets.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a VLAN ID.

Example

```
Device(config)# interface vlan vlan-id
```

3. Insert relay information in BOOTREQUEST.

Example

```
Device(config-if)# ip dhcp relay information option-insert
```

4. Configure the IP address for the interface.

Example

```
Device(config-if)# ip address ip-address
```

5. Configure the destination address for UDP broadcasts.

Example

```
Device(config-if)# ip helper-address ip-address
```

6. Disable the MOP for an interface.

Example

```
Device(config-if)# no mop enabled
```

7. Disable the task of sending MOP periodic system ID messages.

Example

```
Device(config-apgroup)# no mop sysid
```

DHCP requests sent from this VLAN interface will contain Option 82 relay information to aid in DHCP server assignment and troubleshooting.

```
Device# configure terminal
Device(config)# interface vlan 72
Device(config-if)# ip dhcp relay information option-insert
Device(config-if)# ip address 9.3.72.38 255.255.255.0
Device(config-if)# ip helper-address 9.3.72.1
Device(config-if)# no mop enabled
Device(config-apgroup)# no mop sysid
```

Configure DHCP Option 82 through server-ID-override command (CLI)

Configure DHCP Option 82 through the server-ID-override command.

Enable DHCP Option 82 with server-ID override to control how DHCP relay information is inserted on a Cisco device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the server-id override suboption to use an RFC-specific or Cisco-specific value.

   Example

   ```
   Device(config)# ip dhcp compatibility suboption server-override cisco
   ```

3. Configure the link-selection suboption to use an RFC-specific or Cisco-specific value.

   Example

   ```
   Device(config)# ip dhcp compatibility suboption link-selection cisco
   ```

4. Configure a VLAN ID.

   Example

   ```
   Device(config)# interface vlan vlan-id
   ```

5. Insert the server-id override suboption and link-selection suboption.

   Example

   ```
   Device(config-if)# ip dhcp relay information option server-id-override
   ```

6. Configure the IP address for the interface.

   Example

   ```
   Device(config-if)# ip address ip-address
   ```

7. Configure the destination address for UDP broadcasts that your interfce will use.

   Example

   ```
   Device(config-if)# ip helper-address ip-address
   ```

8. Disable MOP for an interface.

Example

```
Device(config-if)# no mop enabled
```

9. Disable the task of sending MOP periodic system ID messages.

Example

```
Device(config-if)# [no] mop sysid
```

The device uses the specified server-id override and link-selection options in DHCP Option 82 relay messages, which the device forwards through the configured VLAN interface.

```
Device# configure terminal
Device(config)# ip dhcp compatibility suboption server-override cisco
Device(config)# ip dhcp compatibility suboption link-selection cisco
Device(config)# interface vlan 72
Device(config-if)# ip dhcp relay information option server-id-override
Device(config-if)# ip address 9.3.72.38 255.255.255.0
Device(config-if)# ip helper-address 9.3.72.1
Device(config-if)# no mop enabled
Device(config-if)# no mop sysid
```

Configure DHCP Option 82 through a subscriber-ID (CLI)

Configure DHCP Option 82 through a Subscriber-ID using commands.

Configure DHCP Option 82 through a subscriber-ID.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a VLAN ID.

Example

```
Device(config)# interface vlan vlan-id
```

3. Insert the subscriber identifier suboption to include the subscriber ID in DHCP messages.

Example

```
Device(config-if)# ip dhcp relay information option subscriber-id subscriber-id
```

4. Configure the IP address for the interface.

Example

```
Device(config-if)# ip address ip-address
```

5. Configure the destination address for UDP broadcasts.

Example

```
Device(config-if)# ip helper-address ip-address
```

6. Disable MOP for an interface.

Example

```
Device(config-if)# no mop enabled
```

7. Disable the task of sending MOP periodic system ID messages.

Example

```
Device(config-apgroup)# no mop sysid
```

The system now inserts the subscriber-ID suboption (DHCP Option 82) into relayed DHCP packets on the specified VLAN interface.

```
Device# configure terminal
Device(config)# interface vlan 72
Device(config-if)# ip dhcp relay information option subscriber-id test10
Device(config-if)# ip address 9.3.72.38 255.255.255.0
Device(config-if)# ip helper-address 92.0.2.1
Device(config-if)# no mop enabled
Device(config-apgroup)# no mop sysid
```

Configure DHCP Option 82 through server-ID-override and subscriber-ID commands (CLI)

Configure DHCP Option 82 Through server-ID-override and subscriber-ID using commands.

Enable customized handling of DHCP Option 82 information for relayed DHCP requests by setting server and subscriber ID options.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a VLAN ID.

Example

```
Device(config)# interface vlan vlan-id
```

3. Insert server ID override and link selection suboptions.

Example

```
Device(config-if)# ip dhcp relay information option server-id-override
```

4. Insert the subscriber identifier suboption.

Example

```
Device(config-if)# ip dhcp relay information option subscriber-id subscriber-id
```

5. Configure the IP address for the interface.

Example

```
Device(config-if)# ip address ip-address
```

6. Configure the destination address for UDP broadcasts.

Example

```
Device(config-if)# ip helper-address ip-address
```

7. Disable the MOP for an interface.

Example

```
Device(config-if)# no mop enabled
```

8. Disable the task of sending MOP periodic system ID messages.

Example

```
Device(config-apgroup)# no mop sysid
```

The interface relays DHCP requests with Option 82 data, including server ID override and subscriber ID.

```
Device# configure terminal
Device(config)# interface vlan 72
Device(config-if)# ip dhcp relay information option server-id-override
Device(config-if)# ip dhcp relay information option subscriber-id test10
Device(config-if)# ip address 9.3.72.38 255.255.255.0
Device(config-if)# ip helper-address 9.3.72.1
Device(config-if)# no mop enabled
Device(config-apgroup)# no mop sysid
```

Configure DHCP Option 82 through different SVIs (CLI)

Configure DHCP Option 82 Through Different SVIs using commands.

Enable DHCP Option 82 using different switch virtual interfaces (SVIs) for relayed requests.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a VLAN ID.

   Example

   ```
   Device(config)# interface vlan vlan-id
   ```

3. Configure a source interface on the VLAN to relay messages.

   Example

   ```
   Device(config-if)# ip dhcp relay source-interface vlan vlan-id
   ```

4. Configure the IP address for the interface.

   Example

   ```
   Device(config-if)# ip address ip-address
   ```

5. Configure the destination address for UDP broadcasts.

   Example

   ```
   Device(config-if)# ip helper-address ip-address
   ```

6. Disable the Maintenance Operations Protocol (MOP) on the interface.

   Example

   ```
   Device(config-if)# no mop enabled
   ```

7. Disable periodic system ID message transmission for MOP.

   Example

   ```
   Device(config-apgroup)# no mop sysid
   ```

The device relays DHCP messages with Option 82 through the configured VLAN SVIs.

```
Device# configure terminal
Device(config)# interface vlan 72
Device(config-if)# ip dhcp relay source-interface vlan 74
Device(config-if)# ip address 9.3.72.38 255.255.255.0
Device(config-if)# ip helper-address 9.3.72.1
Device(config-if)# no mop enabled
Device(config-apgroup)# no mop sysid
```

AP DHCP Option 82 support on FlexConnect local switching mode

Describes AP DHCP Option 82 Support on FlexConnect Local Switching Mode.

AP DHCP Option 82 support on FlexConnect local switching mode is a DHCP relay feature that

- provides additional information on the physical attachment of the client
- enhances security while using DHCP for network address allocation, and
- enables the AP to act as a DHCP relay agent to prevent DHCP client requests from unreliable sources.

DHCP Option 82 functionality

As a DHCP relay agent, the AP can add DHCP Option 82 information such as AP MAC, AP Name, and SSID to DHCP requests from clients before forwarding the requests to the DHCP server. Afterward, the DHCP servers can allocate IP addresses to wireless clients based on the data contained within DHCP Option 82.

> **Note**
>
> This feature is supported only in FlexConnect Local Switching mode.

Configure AP DHCP Option82 support

Configure AP DHCP Option82 support using commands.

Enable DHCP Option82 support on access points to provide additional network information to DHCP servers.

DHCP Option82 allows access points to insert additional information such as AP MAC address, AP name, or SSID into DHCP requests from clients. This information helps DHCP servers make more informed decisions about IP address assignments and network policies.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable configuration for the specified profile policy.

Example

```
Device(config)# wireless profile policy policy-name
```

Example

```
Device(config)# wireless profile policy policy-ap
```

3. Shut down the profile policy.

Example

```
Device(config-wireless-policy)# shutdown
```

4. Configure the AP MAC address or the AP name format to be added to DHCP requests from clients before forwarding them to the DHCP server.

Example

```
Device(config-wireless-policy)# ipv4 dhcp opt82 format apmac|apname
```

Example

```
Device(config-wireless-policy)# ipv4 dhcp opt82 format apmac
```

You can use either apmac or apname, but not both.

5. Configure the unique SSID format to be added to DHCP requests from clients before forwarding them to the DHCP server.

Example

```
Device(config-wireless-policy)# ipv4 dhcp opt82 format ssid
```

6. Enable the profile policy.

Example

```
Device(config-wireless-policy)# no shutdown
```

DHCP Option82 support is now configured on the access point policy, allowing the AP to insert additional network information into DHCP requests from connected clients.

Verify AP DHCP Option82 support

Provides commands to verify AP DHCP Option82 support configuration and status.

Verify the configured parameters for DHCP Option82 using commands on the AP console and controller to confirm the configuration status and settings.

To verify the configured parameters for DHCP Option82, use this command on the AP console:

Note

You can fetch information about the WLAN and WLAN ID using the **show flexconnect WLAN** command.

```
Device# show flexconnect dhcp option-82 wlan 1

DHCP OPTION 82 CONFIG FOR WLAN:
VAP ID      :0
Format      :Binary
Delimiter   :":"
Circuit-ID  :
Remote-ID   :
    AP Name
    SSID-Name
```

To verify whether DHCP Option82 is enabled on the controller, use this command:

```
Device# show wireless profile policy detailed

Opt82
  SSID                      : ENABLED
  AP_ETHMAC                 : DISABLED
  APNAME                    : ENABLED
```

66 RADIUS Authentication and Accounting

Topics:

- RADIUS realms
- RADIUS accounting of AP events
- RADIUS call station identifier
- RADIUS VSA

RADIUS realms

RADIUS Realm details.

A RADIUS realm is an AAA configuration element that

- enables assigning authentication and accounting requests to specific RADIUS servers based on user domain information
- uses the domain portion of a user's Network Access Identifier (NAI) to select the appropriate RADIUS server, and
- provides realm-based filtering and control for authentication and accounting requests within a WLAN.

Feature history
Table 76: Feature history for RADIUS realms

Feature name	Release information	Feature description
RADIUS realms	Cisco IOS XE 16.9.1	RADIUS realms are configuration elements in AAA (Authentication, Authorization, and Accounting) systems. They use the domain portion of a user's Network Access Identifier (NAI)–such as the part after "@" in an email address–to direct authentication and accounting requests to specific RADIUS servers. Realms help organizations manage user access and resource usage across different groups or domains.

When mobile clients connect to a WLAN, the RADIUS realm is included in the Extensible Authentication Protocol Method for UMTS Authentication and Key Agreement (EAP-AKA) identity response of the authentication request packet. For WLAN, the NAI format for EAP-AKA can be specified as *username@domain.com*. In the NAI format, the realm follows the @ symbol and is specified as *example.com*. If the vendor-specific attribute "test" is added, the NAI format becomes *test@example.com*.

The RADIUS Realm feature can be enabled and disabled on a WLAN. If Realm is enabled on a WLAN, the corresponding user should send the username in the NAI format. The controller sends the authentication request to the AAA server only after it receives a realm in NAI format from the client and verifies that the realm complies with the required standards. Additionally, the controller sends accounting requests to the AAA server based on realm filtering.

Realm support on a WLAN

Each WLAN is configured to support NAI realms. Once the realm is enabled on a specific SSID, the system compares the received realms in the EAP identity response to those configured on the RADIUS server. If a username with the realm is not sent by the client, the WLAN uses the default RADIUS server for authentication. When the client's realm does not match the realms configured on the WLAN, the client is deauthenticated and dropped.

If the RADIUS Realm feature is not enabled on a WLAN, the controller uses the username received in the EAP identity request directly and authenticates the user using the configured RADIUS server. The RADIUS Realm feature is disabled on WLANs by default.

- **Realm match for authentication**: In dot1x with EAP methods (similar to EAP AKA), the username is received as part of an EAP identity response. A realm is derived from the username and is matched with the realms that are already configured in the corresponding RADIUS authentication server. If there is a match, the authentication requests are forwarded to the RADIUS server. If there is a mismatch, the client is deauthenticated.

- **Realm match for accounting**: After receiving a client's username through an access-accept message, the system derives the realm from the username when accounting messages are triggered. The realm is compared with the configured accounting realms on the RADIUS accounting server. When there is a match, the server receives the accounting requests. If there is no match, the system drops the accounting requests.

Enable RADIUS realm

Enable RADIUS Realm using commands.

Enable the RADIUS realm to allow the WLAN to use the RADIUS realm selection for AAA authentication.

Enabling the RADIUS realm is required when configuring wireless AAA policies that use RADIUS for authentication on your Cisco device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a new AAA policy.**wireless aaa policy**

   Example

   ```
   Device(config)# wireless aaa policy aaa-policy
   ```

3. Enable AAA RADIUS realm selection.

   Example

   ```
   Device(config-aaa-policy)# aaa-realm enable
   ```

   Note

   Use the **no aaa-realm enable** or the **default aaa-realm enable** command to disable the RADIUS realm.

The AAA RADIUS realm is now enabled and available for authentication purposes in the specified wireless AAA policy.

```
Device# configure terminal
Device(config)# wireless aaa policy policy-1
Device(config-aaa-policy)# aaa-realm enable
```

Configure realm to match the RADIUS server for authentication and accounting

Configure Realm to Match the RADIUS Server for Authentication and Accounting using commands.

Configure the realm to use your chosen RADIUS server group for authentication and accounting network access requests.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a AAA authentication model.

   Example

   ```
   Device(config)# aaa new-model
   ```

3. Set the authorization method.

   Example

   ```
   Device(config)# aaa authorization network default group radius-server-group
   ```

4. Indicate that 802.1x must use the realm group RADIUS server.

   Example

   ```
   Device(config)# aaa authentication dot1x realm group radius-server-group
   ```

5. Define the authentication method at login.

   Example

   ```
   Device(config)# aaa authentication login realm group radius-server-group
   ```

6. Enable accounting so the system sends a start record notice when a client is authorized and a stop record notice when the session ends.

   Example

   ```
   Device(config)# aaa accounting identity realm start-stop group
   radius-server-group
   ```

Your device now uses the realm and RADIUS server group for authentication, authorization, and accounting.

```
Device# configure terminal
Device(config)# aaa new-model
Device(config)# aaa authorization network default group aaa_group_name
Device(config)# aaa authentication dot1x cisco.com group cisco1
Device(config)# aaa authentication login cisco.com group cisco1
Device(config)# aaa accounting identity cisco.com start-stop group v
```

Configure AAA policy for WLANs

Configure the AAA Policy for a WLAN using commands.

Configure authentication, authorization, and accounting (AAA) policies to control and secure access for WLAN users.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a new AAA policy for wireless.

   Example

   ```
   Device(config)# wireless aaa policy aaa-policy-name
   ```

3. Enable AAA RADIUS server selection by realm.

   Example

   ```
   Device(config-aaa-policy)# aaa-realm enable
   ```

4. Return to global configuration mode.

   Example

   ```
   Device(config-aaa-policy)# exit
   ```

5. Configure a WLAN policy profile.

   Example

   ```
   Device(config)# wireless profile policy wlan-policy-profile
   ```

6. Map the AAA policy.**aaa-policy***aaa-policy*

   Example

   ```
   Device(config-wireless-policy)# aaa-policy aaa-policy
   ```

7. Set the accounting list.**accounting-list***acct-config-realm*

   Example

   ```
   Device(config-wireless-policy)# accounting-list acct-config-realm
   ```

8. Return to global configuration mode.

   Example

   ```
   Device(config-wireless-policy)# exit
   ```

9. Configure a WLAN.**wlan***wlan-namewlan-idssid*

   Example

   ```
   Device(config)# wlan wlan-name wlan-id ssid
   ```

10. Enable the security authentication list for IEEE 802.1x.

    Example

    ```
    Device(config-wlan)# security dot1x authentication-list auth-list-realm
    ```

11. Return to global configuration mode.

    Example

    ```
    Device(config-wlan)# exit
    ```

12. Configuresa policy tag.

    Example

    ```
    Device(config)# wireless tag policy policy-name
    ```

13. Map a policy profile to the WLAN.

    Example

    ```
    Device(config-policy-tag)# wlan wlan-name policy policy-profile
    ```

14. Return to global configuration mode.

    Example

    ```
    Device(config-policy-tag)# exit
    ```

The AAA policy is successfully applied to the target WLAN, enabling centralized authentication and accounting for users connecting to that WLAN.

```
Device# configure terminal
Device(config)# wireless aaa policy aaa-policy-1
Device(config-aaa-policy)# aaa-realm enable
Device(config-aaa-policy)# exit
Device(config)# wireless profile policy wlan-policy-1
Device(config-wireless-policy)# aaa-policy aaa-policy-1
Device(config-wireless-policy)# accounting-list cisco.com
Device(config-wireless-policy)# exit
Device(config)# wlan wlan2 14 wlan-aaa
Device(config-wlan)# security dot1x authentication-list cisco.com
Device(config-wlan)# exit
Device(config)# wireless tag policy tag-policy-1
Device(config-policy-tag)# wlan Abc-wlan policy wlan-policy-a
Device(config-policy-tag)# exit
```

Verify RADIUS-realm configuration

Verify the RADIUS-Realm Configuration.

Run this command to verify the RADIUS-realm configuration.

```
Device# show wireless client mac-address 14bd.61f3.6a24 detail
Client MAC Address : 14bd.61f3.6a24
Client IPv4 Address : 192.0.2.1
Client IPv6 Addresses : fe80::286e:9fe0:7fa6:8f4
Client Username : cisco-mac@cisco.com
AP MAC Address : 4c77.6d79.5a00
AP Name: AP4c77.6d53.20ec
AP slot : 1
Client State : Associated
Policy Profile : name-policy-profile
Flex Profile : N/A
Wireless LAN Id : 3
Wireless LAN Name: ha_realm_WLAN_WPA2_AES_DOT1X
BSSID : 4c77.6d79.5a0f
Connected For : 26 seconds
Protocol : 802.11ac
Channel : 44
Client IIF-ID : 0xa0000001
Association Id : 1
Authentication Algorithm : Open System
Client CCX version : No CCX support
Re-Authentication Timeout : 1800 sec (Remaining time: 1775 sec)
Input Policy Name  : None
Input Policy State : None
Input Policy Source : None
Output Policy Name  : None
Output Policy State : None
Output Policy Source : None
WMM Support : Enabled
U-APSD Support : Enabled
  U-APSD value : 0
  APSD ACs    : BK, BE, VI, VO
Fastlane Support : Disabled
Power Save : OFF
Supported Rates : 9.0,18.0,36.0,48.0,54.0
Mobility:
  Move Count                  : 0
  Mobility Role               : Local
  Mobility Roam Type          : None
  Mobility Complete Timestamp : 06/12/2018 19:52:35 IST
Policy Manager State: Run
NPU Fast Fast Notified : No
Last Policy Manager State : IP Learn Complete
Client Entry Create Time : 25 seconds
Policy Type : WPA2
Encryption Cipher : CCMP (AES)
Authentication Key Management : 802.1x
Encrypted Traffic Analytics : No
Management Frame Protection : No
Protected Management Frame - 802.11w : No
EAP Type : PEAP
VLAN : 113
Multicast VLAN : 0
Access VLAN : 113
Anchor VLAN : 0
WFD capable : No
Managed WFD capable : No
```

```
Cross Connection capable : No
Support Concurrent Operation : No
Session Manager:
  Interface        : capwap_9040000f
  IIF ID           : 0x9040000F
  Authorized       : TRUE
  Session timeout  : 1800
  Common Session ID: 097704090000000DF4607B3B
  Acct Session ID  : 0x00000fa2
  Aaa Server Details
  Server IP        : 192.0.2.2
  Auth Method Status List
       Method : Dot1x
              SM State         : AUTHENTICATED
              SM Bend State    : IDLE
  Local Policies:
       Service Template : wlan_svc_name-policy-profile_local (priority 254)
              Absolute-Timer   : 1800
              VLAN             : 113
  Server Policies:
  Resultant Policies:
              VLAN             : 113
              Absolute-Timer   : 1800
DNS Snooped IPv4 Addresses : None
DNS Snooped IPv6 Addresses : None
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 0
Fast BSS Transition Details :
  Reassociation Timeout : 0
11v BSS Transition : Not implemented
FlexConnect Data Switching : Central
FlexConnect Dhcp Status : Central
FlexConnect Authentication : Central
FlexConnect Central Association : No
.
.
.
Fabric status : Disabled
Client Scan Reports
Assisted Roaming Neighbor List
```

RADIUS accounting of AP events

Introduces accounting of AP events, providing instructions for configuring accounting method lists within AP profiles, verifying AP accounting information, and implementing AAA accounting using named and predefined method lists through CLI-based procedures.

RADIUS accounting of AP events is a network monitoring mechanism that

- Tracks the status transitions of APs within a wireless controller environment
- Records AP join and disjoin events
- Provides historical visibility into AP downtime and uptime through accounting messages sent to a RADIUS server.

Feature History

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 77: Feature history table

Feature Name	Release	Description
Device Ecosystem Data	Cisco IOS XE 17.10.1	This feature sends device analytics data that is present in the RADIUS accounting request to Cisco ISE to profile endpoints The command is introduced: • dot11-tlv-accounting
Chargeable User Identity in RADIUS Accounting	Cisco IOS XE 17.9.1	Chargeable User Identity (CUI) is a unique identifier for a client visiting a network. This attribute can be used as an alternative for the client's username as part of the authentication process. The command is introduced: • dot11-tlv-accounting
Improved Logging in RADIUS Accounting	Cisco IOS XE 17.1.1	Prior to Cisco IOS XE Amsterdam 17.1.1 release, the controller did not send accounting messages for AP join and disjoin events during network issues. From Cisco IOS XE Amsterdam 17.1.1 Release and later, the RADIUS server keeps a record of all APs that were down and have come up.

Configure accounting method-list for an AP profile

Configure Accounting Method-List for an AP Profile using commands.

Define an accounting method list within an access point (AP) profile to enable or disable accounting for AP operations.

Use this task to specify how accounting is managed for an AP profile on your device. This allows tracking of AP events and assists with auditing and troubleshooting

Before you begin

- Identify the AP profile name you want to configure. The default AP profile name is `default-ap-profile`.
- Determine the accounting method list name you wish to apply.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the AP profile. The default AP join profile name is default-ap-profile.

Example

```
Device(config)# ap profile ap-profile-name
```

3. Configure the accounting method list for the AP profile.

Example

```
Device(config-ap-profile)# [no] accounting method-list method-list-name
```

Use the **no** form of this command to disable the accounting method list.

The system associates the specified accounting method list with the AP profile, enabling or disabling accounting

Example

```
Device# configure terminal
Device(config)# ap profile ap-profile-name
Device(config-ap-profile)# [no] accounting method-list method-list-name
```

Verify AP accounting information

Verify the AP Accounting Information.

Verify the AP accounting information, including the MAC address, packets sent, packets received, and the method list.

```
Device#show wireless stats ap accounting
Base MAC       Total packet Send    Total packet Received Methodlist
-------------------------------------------------------------------------------
00b0.e192.0f20     4       3    abc
38ed.18cc.5788     8       8         ML_M
70ea.1ae0.af08     0       0         ML_A
```

View details for a method list configured for an AP profile.

```
Device# show ap profile name Method-list detailed
AP Profile Name               : test-profile
Description                   :
.
.
.
Method-list name              : Method-list
Packet Sequence Jump DELBA    : ENABLED
Lag status                    : DISABLED
.
Client RSSI Statistics
  Reporting                   : ENABLED
  Reporting Interval          : 30 seconds
```

Configure AAA accounting using default method list (CLI)

Use this task to enable and configure AAA command accounting to monitor and record user command activity using the default method list on your device.

Use this task to monitor and record user command activity on devices through AAA accounting features.

Configure AAA accounting to track user commands executed on a controller, leveraging the default accounting method. This supports compliance and security needs.

Before you begin

- Confirm that AAA is enabled on the device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create an accounting method list and enables accounting.

   Example

   ```
   Device(config)# aaa accounting commands 15 default start-stop group group-name
   ```

   - *privilege_level*: AAA accounting level. The valid range is from zero to 15.
   - *group-name*: AAA accounting group that supports only TACACS+ group.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The controller records user command activities according to the configured accounting method.

Configure HTTP command accounting using named method list (CLI)

Use this task to configure HTTP command accounting on your device, allowing you to monitor and record user command activity with AAA features using a named method list.

Set up command accounting to track user actions on network devices via HTTP with a specified AAA method list using commands.

HTTP command accounting provides auditing and compliance by recording commands executed by users. Using a named method list offers flexibility for different accounting requirements.

Before you begin

- Ensure AAA accounting is enabled on your device.
- Have a predefined AAA accounting method list (if not, configure one).

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure HTTP command accounting using the named method list.

   Example

   ```
   Device(config)# ip http accounting commands 1 oneacct
   ```

   - *level*: The privilege value ranges from zero to 15. By default, the command privilege levels available on the controller are:
     - *0*: Includes the **disable**, **enable**, **exit**, **help**, and **logout** commands.
     - *1*: Includes all the user-level commands at the controller prompt (>).
     - *15*: Includes all the enable-level commands at the controller prompt (>).
   - *named-accounting-method-list*: Name of the predefined command accounting method list.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The device records user command activities in accordance with the configured accounting method.

RADIUS call station identifier

Explains how RADIUS call station identifiers are used in network access scenarios to specify the network endpoint a client attempts to access and how this information supports authorization policies.

A RADIUS call station identifier is an attribute in the RADIUS protocol that

- allows a Network Access Server (NAS) to specify information about the endpoint (the called station) that a client is trying to access
- is included in Access-Request packets (as Called-Station-Id), and
- enables network policies by providing context for authorization decisions.

Use cases

- In dial-up scenarios, the attribute can contain the phone number dialed by the user. The NAS captures this attribute in the Access-Request packet using Dialed Number Identification (DNIS) or similar technology.
- For IEEE 802.1X authenticators (wired or wireless network access), the attribute can contain the MAC address of the bridge or AP, formatted as ASCII text.

Role in RADIUS authorization

- The RADIUS server can use this attribute to define which MAC addresses, network segments, or called stations a client is allowed to access.
- The RADIUS server can restrict or allow access based on the network resource or location to which the client is connecting. However, this is possible only in configurations that support preauthentication. In these configurations, a client attempts to authenticate before fully connecting.

The **Called-Station-Id** attribute enables the NAS to specify to the RADIUS server which endpoint or resource the client is requesting. The server uses this information to enforce connection policies.

> **Note**
>
> The **Called-Station-Id** attribute applies only to Access-Request packets, not to Access-Accept or CoA-Request packets.

Feature history for RADIUS call station identifier

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 78: Feature history for RADIUS call station identifier

Release	Feature Information
Cisco IOS XE Bengaluru 17.4.1	The RADIUS called station identifier configuration is enhanced to include more attributes. The newly added options for authentication and accounting are: • policy-tag-name • flex-profile-name • ap-macaddress-ssid-flexprofilename • ap-macaddress-ssid-policytagname • ap-macaddress-ssid-sitetagname • ap-ethmac-ssid-flexprofilename • ap-ethmac-ssid-policytagname • ap-ethmac-ssid-sitetagname For more information on the attributes listed above, see the commands: • **radius-server attribute wireless accounting call-station-id** • **radius-server attribute wireless authentication call-station-id**

Configure a RADIUS call station identifier

Use this task to specify a custom call station (Called-Station-ID) identifier in RADIUS authentication and accounting messages for policy tracking and assignment.

Set a custom call station identifier (**Called-Station-ID**) for RADIUS authentication and accounting messages sent from the device.

Use this task to specify a policy tag name or identifier in RADIUS messages for better tracking or policy assignment.

Before you begin

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a call station identifier sent in the RADIUS authentication messages.

   Example

   ```
   Device(config)# radius-server attribute wireless authentication call-station-id
    policy-tag-name
   ```

3. Configure a call station identifier sent in the RADIUS accounting messages.

   Example

   ```
   Device(config)# radius-server attribute wireless accounting call-station-id
   policy-tag-name
   ```

The specified policy tag name is now included as the **Called-Station-ID** in all RADIUS authentication and accounting messages.

RADIUS VSA

RADIUS VSA details.

A RADIUS VSA is a vendor-specific attribute used in RADIUS implementations that

- allows vendors to communicate specialized information between a network access server and a RADIUS server
- enables support for attributes not defined by standard RADIUS specifications, and
- is included in authentication or accounting packets, or both, according to implementation needs.

The main elements of the VSA are:

- Type: Identifies the attribute type.
- Length: Indicates attribute length.

- String (Data): Contains vendor-specific data, which includes:
  - Vendor ID
  - Vendor type
  - Vendor length
  - Vendor data

Supported modes and attributes

Authentication and accounting requests per WLAN support these VSAs in addition to the existing AAA attributes.

Table 79: Newly supported attributes

Attribute name	Well-known attribute	VSA sub-attribute	Vendor ID
SVR-Zip-Code	26	14	14369
SVR-Device-Type	26	17	14369
SVR-Device-Model-Number	26	18	14369
SVR-Lat-Long	26	19	14369
SVR-Venue-Category	26	20	14369
SVR-Network-Type	26	21	14369
Aggregation-AAA	26	22	14369
BW-Venue-Id	26	7	22472
BW-Venue-TZ	26	8	22472
BW-Class	26	10	22472
BW-Venue-Description	26	11	22472
BW-ISO-Country-Code	26	14	22472
BW-E164-Country-Code	26	15	22472
BW-State-Name	26	16	22472
BW-City-Name	26	17	22472
BW-Area-Code	26	18	22472
BW-User-Group	26	27	22472
BW-Venue-Name	26	29	22472
BW-Operator-Name	26	37	22472

> **Note**
>
> You can use this feature only in FlexConnect central authentication mode with local switching. FlexConnect local authentication mode is not supported. Use this feature for wireless sessions only.

Create an attribute list

Create an Attribute List using commands.

Create a custom AAA attribute list for use in policy mapping.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a AAA attribute list.

   Example

   ```
   Device(config)# aaa attribute list attribute-list
   ```

3. Specify a AAA attribute type.

   Example

   ```
   Device(config-attr-list)# attribute type attribute-type
   Device(config-attr-list)# attribute type BW-City-Name "MUMBAI"
   ```

4. Optional: Specify a AAA attribute type such as the state name.

   Example

   ```
   Device(config-attr-list)# attribute type BW-State-Name "MAHARASHTRA"
   ```

5. Optional: Specify a AAA attribute type such as the venue name.

   Example

   ```
   Device(config-attr-list)#attribute type BW-Venue-Name "WANKHEDE"
   ```

6. Return to Privileged EXEC mode.

Example

```
Device(config-attr-list)# end
```

After you create the AAA attribute list, you can use it in policy mapping.

What to do next

Create a AAA policy and map the attribute list.

Map a AAA policy to WLAN policy profile

Map a AAA Policy to the WLAN Policy Profile using commands.

Associate a AAA policy with a WLAN policy profile. This enforces authentication and authorization rules on wireless clients.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a new wireless policy profile.

   Example

   ```
   Device(config)# wireless profile policy EAP-AKA profile-policy
   ```

3. Create a new AAA policy.

   Example

   ```
   Device(config-wireless-policy)# aaa-policy Verizon-aaa-policy aaa-policy
   ```

4. Return to Privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

You have mapped the AAA policy to the WLAN policy profile.

What to do next

Map the WLAN policy profile to a WLAN.

Map the WLAN policy profile to a WLAN

Map the WLAN Policy Profile to a WLAN using commands.

Associate a previously configured WLAN policy profile to a WLAN on a Cisco device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a new policy tag.

   Example

   ```
   Device(config)# wireless tag policy policy-name
   ```

3. Map the policy profile to a WLAN.

   Example

   ```
   Device(config-policy-tag)# wlan wlan-profile-name policy aaa-policy
   ```

4. Return to Privileged EXEC mode.

   Example

   ```
   Device(config-policy-tag)# end
   ```

The specified WLAN policy profile is successfully mapped to the WLAN.

```
Device# configure terminal
Device(config)# wireless tag policy EAP-AKA
Device(config-policy-tag)# wlan EAP-AKA policy EAP-AKA
Device(config-policy-tag)# end
```

67 Cisco StadiumVision

Topics:

- Cisco StadiumVision

Introduces Cisco StadiumVision, outlining its key features and guiding users through wireless controller parameter configuration via GUI and CLI, along with reference procedures for verification and successful deployment.

Cisco StadiumVision

Outlines Cisco StadiumVision functionality, providing procedures to configure wireless controller parameters using both GUI and CLI methods, and includes reference steps for verifying configurations.

Cisco StadiumVision is a high-definition IPTV solution that:

- provides advanced digital content management and delivery,
- transforms the appearance and atmosphere of venues, and
- is built on the Cisco Connected Stadium solution.

This technology enables replay of critical moments of a game on Wi-Fi-capable devices.

To enable Cisco StadiumVision on the controller , you need to configure these parameters:

1. On the Wireless Controller :
   - Multicast Data Rate
   - RX Sensitivity SOP
   - Multicast Buffer
2. CAPWAP
3. AP Radio Driver and Firmware:
   - Multicast Data Rate
   - RX Sensitivity SOP
   - Multicast Buffer

Configure Cisco StadiumVision parameters (GUI)

Configure Cisco StadiumVision using GUI.

This task describes how to configure parameters for Cisco StadiumVision using the GUI.

Procedure

1. Choose **Configuration > Wireless > Advanced**.
2. Click the **High Density** tab.
3. Set the **Multicast Data Rate** for 5 GHz radio or 2.4 GHz radio using the respective drop-down lists.
4. Click **Apply** to save the settings.

Upon completion, the wireless parameters for StadiumVision will be configured successfully.

Configure Cisco StadiumVision parameters (CLI)

Configure parameters for Cisco StadiumVision using commands.

This task provides instructions for configuring various parameters in Cisco StadiumVision, including WLAN, enhanced multicast buffer size, and radio receive sensitivity SOP thresholds.

- Multicast buffer and data rate configurations are supported for all AP models.

Procedure

1. Configure WLAN

   Example

   ```
   Device(config)# wlan wlan1 10
   ```

   Configures a WLAN.

2. Configure enhanced multicast buffer size

   Example

   ```
   Device(config-wlan)# multicast buffer 45
   ```

   Configures enhanced multicast buffer size between 30 (default) and 60 on a WLAN.

   > **Note**
   >
   > You can enable only two out of the possible 512 WLANs configured on the controller and embedded wireless controller for enhanced multicast buffers.

3. Configure the radio receive sensitivity SOP threshold

   Example

   ```
   Device(config)# ap dot11 [5ghz| 24ghz] rx-sop threshold custom -70
   ```

   Configures the radio receive sensitivity SOP threshold between -60 to -85 dB, which can also be configured as predefined auto, low, high, medium values specific to 5ghz or 24ghz bands.

   By default, the configuration is disabled and its value is set to *auto*. If the RxSOP value of *auto* (0) is pushed, then the AP considers the value burnt-in during manufacturing.

Upon completion of these configurations, the Cisco StadiumVision system will operate with the new settings applied, enhancing performance and compatibility based on the configured parameters.

Verify StadiumVision configurations

Verify StadiumVision configurations.

- **show ap rf-profile name** *rf-name* **detail**
- **show ap dot11 5ghz** *high-density*

Rx SOP

```
Device#show ap rf-profile name Typical_Client_Density_rf_5gh detail | i SOP
Rx SOP Threshold                : auto
```

Multicast Buffer

```
Device#show wlan id 1 | sec Buffer
Multicast Buffer                              : Enabled
Multicast Buffer Size                         : 45

Device#

Device#sh wlan name vwlc-OpenAuth | inc Buffer
Multicast Buffer                              : Enabled
Multicast Buffer Size                         : 45
Device#
```

Multicast Data Rate

```
Device#sh ap dot11 24ghz high-density
AP Name                          Mac Address            Slot         Rxsop

Threshold Type Value (dbm)       Multicast Data Rate(Mbps)
-------------------------------------------------------------------------------
-----------------------------------------------
test-1800-AP                      aaaa.bbbb.cccc            0         auto

        0                         54
AP4001.7AB2.BEB6                  aaab.bbbb.cccc            2         auto

        0                         54
AP70DF.2FA2.72EE                  aaac.bbbb.cccc            0         auto

        0                         0

Device#show ap dot11 5ghz high-density
AP Name                          Mac Address            Slot         Rxsop

Threshold Type Value (dbm)       Multicast Data Rate(Mbps)
-------------------------------------------------------------------------------
-----------------------------------------------
Saji-1800-AP                      aaab.bbbb.cccc            1         auto

        0                         12
Saji-2802I-AP                     aaab.bbbb.cccc            0         custom

         -82                       12
Saji-2802I-AP                     aaac.bbbb.cccc            1         custom

         -82                       12
AP4001.7AB2.BEB6                  aaad.bbbb.cccc            0         custom

         -82                       12
AP4001.7AB2.BEB6                  aaae.bbbb.cccc            1         custom

         -82                       0
AP500F.8086.8B56                  aaaf.bbbb.cccc            0         custom

         -82                       12
AP500F.8086.8B56                  aaag.bbb.cccc            1         custom

         -82                       12
AP70DF.2FA2.72EE                  aaah.bbbb.cccc            1         auto

        0                         0
```

```
Device#
Device(config)#ap dot11 5ghz rf-profile test_5ghz_rf
Device(config-rf-profile)#high-density multicast data-rate RATE_18M


Device#  show ap rf-profile name test_5ghz_rf detail | inc Multicast
Multicast Data Rate               : 18 Mbps
Device#
```

68 Persistent SSID Broadcast

Topics:

- Persistent SSID broadcast

This chapter explains the persistent SSID broadcast feature for Cisco Aironet 1542 RAPs and MAPs, providing command steps to enable or disable it and commands to verify AP configuration and behavior when the WAN link is down.

Persistent SSID broadcast

Explains how persistent SSID broadcasts allow mesh access points to continue broadcasting the SSID even if WAN connectivity is lost, supporting network troubleshooting and management in Cisco Aironet 1542 APs operating in Flex+Bridge mode.

A persistent SSID broadcast is a wireless network feature that

- enables Root APs (RAPs) and Mesh APs (MAPs) to continue broadcasting the SSID when the WAN link is down
- helps operators isolate issues between the backhaul network and the access wireless network, and
- applies specifically to Cisco Aironet 1542 APs operating in FlexConnect plus Bridge mode.

RAPs have a wired connection to the controller while MAPs have wireless connection to the controller .

This requirement helps isolate responsibility, whether the fault lies with the backhaul or the access wireless network. Different operators may own each part of the network.

RAPs and MAPs broadcast the SSID while in standalone mode if the default gateway is reachable.

Also refer Mesh Deployment Guide for Cisco Catalyst 9800 Series Wireless Controllers.

Configure persistent SSID broadcast (CLI)

Use this procedure to configure or disable persistent SSID broadcast on an access point profile using commands.

Enable or disable persistent SSID broadcast so that associated APs re-join and maintain SSID visibility as configured using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the AP profile.

   Example

   ```
   Device(config)# ap profile ap-profile-name
   ```

3. Configure the SSID broadcast mode.

   Example

   ```
   Device(config-ap-profile)# ssid broadcast persistent
   ```

   ```
   Device(config-ap-profile)# [no] ssid broadcast persistent
   ```

   The **persistent** keyword enables a persistent SSID broadcast, where the associated APs re-join. Use the **no** form of the command to disable the feature.

> **Note**
>
> Enabling or disabling this feature causes the AP to re-join.

Verify persistent SSID broadcast

This section lists the command and relevant output fields used to confirm the persistent SSID broadcast status on Cisco APs.

To view the configuration of all Cisco APs, use this **show** command:

```
Device#show ap config general
Cisco AP Name   : AP4C77.6DF2.D598
=================================================
Office Extend Mode                                : Disabled
Persistent SSID Broadcast                         : Enabled
Remote AP Debug                                   : Disabled
```

69 Network Monitoring

Topics:

- Network monitoring

Network monitoring

Introduces network monitoring systems, highlighting real-time data collection through APIs, use of NETCONF/YANG for automation and programmability, and mechanisms for synchronous status updates and asynchronous notifications, including integration with third-party applications and key operational considerations in Cisco environments.

A network monitoring system is a network management solution that

- collects and exposes real-time operational data from network devices through APIs
- uses protocols such as NETCONF/YANG, and other technologies to automate data transfer and enable programmability, and
- provides synchronous status updates and asynchronous notifications for alarms, events, and third-party applications.

Network monitoring systems can expose APIs or push data to third-party applications. In Cisco environments, data transfer commonly occurs through the NETCONF/YANG protocol, supporting automated and programmable network operations.

- The two types of information provided are:
  - Status information received synchronously: NETCONF is the management interface for status information received synchronously. It allows publishing the operational state of the device, including the controller.
  - Alarm and event information sent asynchronously: NETCONF/YANG push is used for alarm and event information. It provides the mechanism for sending subscribed NETCONF notifications.
- Contact API or Developer Support for NETCONF/YANG features at this link:

  https://developer.cisco.com/site/support/#

> **Note**
>
> When using NETCONF, you may see a user named \"yang_mgmt_infra"\ appearing in accounting records for TACACS or RADIUS. This user is internal. If you pull data from Cisco IOS, you should expect to see this user.

Status information received synchronously - configuration examples

Configuration example for Status Information, Alarm and Event Information.

You can use the NETCONF/YANG interface to fulfill customer requests.

To enable Status Information, Alarm Information, and Event Information, configure the NETCONF server on the controller using this command.

netconf-yang

> **Note**
>
> The Cisco Catalyst 9800 wireless controller supports only RSA keys for the trustpoint used by the ncsshd process. EC keys are not supported; using them causes the ncsshd process to crash and prevents you from using NETCONF.

The **netconf-yang** command enables notifications, configuration access, and operation access (OAM) through NETCONF/YANG. For details about the NETCONF/YANG, refer to the *NETCONF Protocol* chapter of the Programmability Configuration Guide at https://www.cisco.com/c/en/us/support/ios-nx-os-software/ios-xe-17/products-installation-and-configuration-guides-list.html

For the Status Information Received Synchronously type, NETCONF exports these items:

- Name of the village
- APs in each village
- Status of each AP
- Number of clients currently connected and logged on in each village and each AP

All the data for the items listed above is already available as the controller operational data exported through NETCONF. The examples below explain where the data items listed are available.

This command is used in the controller:

```
wireless tag site village_name_1
```

The site tags can be retrieved by NETCONF using the **get-config** operation.

Example output for **Name of the Village**:

```
<site-cfg-data xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-site-cfg">
[...]
<site-tag-configs>
  <site-tag-config>
    <site-tag-name>village_name_1</site-tag-name>
    <description>custom user site tag for a village</description>
  </site-tag-config>
[...]
</site-tag-configs>
```

The operational data for the controller lists all connected (joined) APs and their corresponding site tags. Each output entry includes detailed information about the access point and its site tag. The relevant fields and controller show commands are presented here:

Example output of **AP per Village**:

```
<data>
    <access-point-oper-data
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-access-point-oper">
    [...]
      <radio-oper-data>
        <wtp-mac>00:1b:0c:00:02:00</wtp-mac>   #show ap dot11 {24ghz|5ghz}
summary "MAC Address"
        <radio-slot-id>0</radio-slot-id>       #show ap dot11 {24ghz|5ghz}
summary "Slot"
        <ap-mac>00:1b:0c:00:02:00</ap-mac>
        <slot-id>0</slot-id>
        <radio-type>1</radio-type>             # 1 - 2.4GHz, 2 - 5GHz
```

```
        <admin-state>enabled</admin-state>     #show ap dot11 {24ghz|5ghz}
summary "Admin State"
        <oper-state>radio-up</oper-state>      #show ap dot11 {24ghz|5ghz}
summary "Oper State"
    [...]
[...]
      <capwap-data>
        <wtp-mac>00:1b:0c:00:02:00</wtp-mac>                #show ap summary
 "Radio MAC"
        <ap-operation-state>registered</ap-operation-state> #show ap summary
 "State"
        <ip-addr>10.102.140.10</ip-addr>                    #show ap summary
 "IP Address"
        [...]
        <admin-state>1</admin-state>              #show ap status   "Status",
 1 - Enabled, 2 - Disabled
       <location>default-location </location>    #show ap summary  "Location"

        <country-code>CH </country-code>
        <name>AP_A-1</name>                        #show ap summary  "AP Name"

 [...]
        <tag-info>
          [...]
          <site-tag>
           <site-tag-name>village_name_1</site-tag-name> #show ap name AP_A-1
 config general "Site Tag Name"
            [...]
          </site-tag>
[...]
```

The operational data of the controller contains information about all connected wireless clients. This includes detailed device-related information, such as MAC address, IP address, state, and access point name.

Example output of the **Number of clients currently online and logged in each village and each AP**:

```
<data>
    <client-oper-data
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-client-oper">
      <common-oper-data>
        <client-mac>00:00:1a:04:00:02</client-mac>     #show wireless client
summary  "MAC Address"
        <ap-name>AP_A-1</ap-name>                      #show wireless client
summary  "AP Name"
        [...]
        <co-state>client-status-run</co-state>         #show wireless client
summary  "State"
```

Alarm and Event Information Received Asynchronously - Configuration Examples

Configuration examples for Alarm and Event Information Received Asynchronously.

The push functionality for the alarm and event information is fulfilled with on-change notifications through NETCONF dynamic subscriptions, with XML encoding.

Example output of **AP Up/Down Events - Subscription**

```
Request:

<?xml version="1.0" encoding="UTF-8"?>
<rpc message-id="urn:uuid:b0c581c9-ff5a-4352-9e64-7f2ce1ec603a"
```

```
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
 <establish-subscription
xmlns="urn:ietf:params:xml:ns:yang:ietf-event-notifications"
   xmlns:yp="urn:ietf:params:xml:ns:yang:ietf-yang-push">
  <stream>yp:yang-push</stream>

<yp:xpath-filter>/access-point-oper-data/capwap-data/ap-operation-state</yp:xpath-filter>

  <yp:dampening-period>0</yp:dampening-period>
 </establish-subscription>
</rpc>

Reply:

<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
message-id="urn:uuid:673b42b2-e988-4e20-a6c3-0679c08e6114"><subscription-result
 xmlns='urn:ietf:params:xml:ns:yang:ietf-event-notifications'
xmlns:notif-bis="urn:ietf:params:xml:ns:yang:ietf-event-notifications">notif-bis:ok</subscription-result>
<subscription-id
xmlns='urn:ietf:params:xml:ns:yang:ietf-event-notifications'>2147483652</subscription-id>
</rpc-reply>
-->>
(Default Callback)
Event time      : 2018-03-09 15:08:21.880000+00:00
Subscription Id : 2147483651
Type            : 2
Data            :
<datastore-changes-xml xmlns="urn:ietf:params:xml:ns:yang:ietf-yang-push">
  <yang-patch xmlns="urn:ietf:params:xml:ns:yang:ietf-yang-patch">
    <patch-id>null</patch-id>
    <edit>
      <edit-id>edit1</edit-id>
      <operation>merge</operation>
      <target>/access-point-oper-data/capwap-data</target>
      <value>
        <capwap-data
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-access-point-oper">
          <ap-operation-state>registered</ap-operation-state>
          <wtp-mac>00ab11006600</wtp-mac>
        </capwap-data>
      </value>
    </edit>
  </yang-patch>
</datastore-changes-xml>
<<--
```

70 Creating a Lobby Ambassador Account

Topics:

- Lobby ambassador account

Lobby ambassador account

Explains the function and capabilities of lobby ambassador accounts for managing guest user access in Cisco network environments.

A lobby ambassador account is a user account that

- allows designated personnel (lobby ambassadors) to create and delete guest users on the network
- allows lobby ambassadors to assign guest user parameters such as password, life span, and role profile, and
- restricts lobby ambassadors from creating WLAN or web authentication policies.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 80: Feature history for lobby ambassador account

Feature Name	Release Information	Feature Description
Lobby ambassador account	Cisco IOS XE Dublin 17.2.x	From Cisco IOS XE Cupertino 17.2.x onwards, lobby administrators can add or delete a client from the allowed list to manage the association with a WLAN or SSID.

As a global administrator, you create a lobby ambassador user to manage guest accounts.

Configure the RADIUS server with the Cisco-AV-pair privilege level set greater than zero to enable lobby ambassador functionality.

Note

An administrator can create lobby ambassador accounts remotely using a RADIUS or TACACS server, or locally.

Only the administrator can create WLAN policies, web authentication policies, and AAA attribute lists. Lobby ambassadors use these attribute lists to map guest users to role profiles, including Quality-of-Service profiles.

After upgrading to Cisco Catalyst 9800 Controller software release 17.2.x, clear the browser cache to correctly view the lobby admin GUI.

Create a lobby ambassador user account

Create a lobby ambassador user account using the GUI and CLI.

You can configure administrator or lobby ambassador usernames and passwords to prevent unauthorized users from reconfiguring the controller and viewing configuration information.

Create a user account (GUI)

Create a user account using the GUI.

Add a new user account to the system.

Procedure

1. Choose **Administration** > **User Administration**.
2. Click **Add**.
3. In the **User Name** field, enter a user name for the new account.
4. Select the policy that you want to associate with the user from the **Policy** drop-down list.
5. Select the privilege level that you want to associate with the user by clicking the user privilege icon from the **Privilege** drop-down list. The options are:
   - **Go to Basic Mode**
   - **Go to Advanced Mode**

   Go to Basic Mode: This privilege level defines the commands that users enter using the CLI after they have logged into the device. Privilege one allows access in user EXEC mode and privilege 15 allows access in Privileged EXEC mode.

   Go to Advanced Mode:

   Admin: Users with Privilege 15 can execute all the **show**, **config**, and **exec** commands on the device. These users will have access to all GUI sections.

   Read Only: Users with Privileges one to 14 are considered read-only users. The default privilege is one if a user is created using the GUI. These users have access only to the Dashboard and the Monitoring sections.

   No Access: Users with Privilege zero can log in to the device through Telnet or SSH to access the CLI, but cannot access the GUI.

   Lobby Admin: Users who can create only guest user accounts. Lobby ambassadors can create and delete guest users and set parameters such as:
   - Password.
   - Lifetime of the guest user.
   - Guest role profiles (Quality-of-Service) profiles that should be applied on a guest using the AAA attribute list.
6. In the **Password** field, enter a password for the new account.
7. In the **Confirm Password** field, enter the same password again to confirm again.
8. Click **Apply to Device**.

Log in using the lobby account

Use commands to log in using the lobby account.

Execute the following commands before logging in using the lobby credentials:

- **aaa new-model**
- **aaa authorization exec default local**
- **ip http authentication aaa**

Log out from the administrator account. Then, log in using the lobby credentials. After logging in, the **Guest User** page appears.

Create a lobby ambassador account (CLI)

Create a lobby ambassador account using commands.

Add a new lobby ambassador (guest network administrator) to the system using CLI commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a user account.

   Example

   ```
   Device(config)# user-name user-name example-user
   ```

3. Specify the account type as lobby admin.

   Example

   ```
   Device(config-user-name)# type lobby-admin
   ```

4. Create a password for the lobby administrator account.

   Example

   ```
   Device(config-user-name)# password 0 password example-password
   ```

5. Create an attribute list for lobby admin access.

   Example

   ```
   Device(config-user-name)# aaa attribute list user-name example-user
   ```

6. Create an attribute type for lobby admin access.

   Example

   ```
   Device(config-user-name)# attribute type wlan-profile-name wlan_wl_mab
   ```

7. Return to the global configuration mode.

   Example

   ```
   Device(config-user-name)# exit
   ```

71 Lobby Ambassador Account

Topics:

- Lobby ambassador account

Lobby ambassador account

Explains the function and capabilities of lobby ambassador accounts for managing guest user access in Cisco network environments.

A lobby ambassador account is a user account that

- allows designated personnel (lobby ambassadors) to create and delete guest users on the network
- allows lobby ambassadors to assign guest user parameters such as password, life span, and role profile, and
- restricts lobby ambassadors from creating WLAN or web authentication policies.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 81: Feature history for lobby ambassador account

Feature Name	Release Information	Feature Description
Lobby ambassador account	Cisco IOS XE Dublin 17.2.x	From Cisco IOS XE Cupertino 17.2.x onwards, lobby administrators can add or delete a client from the allowed list to manage the association with a WLAN or SSID.

As a global administrator, you create a lobby ambassador user to manage guest accounts.

Configure the RADIUS server with the Cisco-AV-pair privilege level set greater than zero to enable lobby ambassador functionality.

Note

An administrator can create lobby ambassador accounts remotely using a RADIUS or TACACS server, or locally.

Only the administrator can create WLAN policies, web authentication policies, and AAA attribute lists. Lobby ambassadors use these attribute lists to map guest users to role profiles, including Quality-of-Service profiles.

After upgrading to Cisco Catalyst 9800 Controller software release 17.2.x, clear the browser cache to correctly view the lobby admin GUI.

Create a lobby ambassador user account

Create a lobby ambassador user account using the GUI and CLI.

You can configure administrator or lobby ambassador usernames and passwords to prevent unauthorized users from reconfiguring the controller and viewing configuration information.

Create a user account (GUI)

Create a user account using the GUI.

Add a new user account to the system.

Procedure

1. Choose **Administration** > **User Administration**.
2. Click **Add**.
3. In the **User Name** field, enter a user name for the new account.
4. Select the policy that you want to associate with the user from the **Policy** drop-down list.
5. Select the privilege level that you want to associate with the user by clicking the user privilege icon from the **Privilege** drop-down list. The options are:
   - **Go to Basic Mode**
   - **Go to Advanced Mode**

   Go to Basic Mode: This privilege level defines the commands that users enter using the CLI after they have logged into the device. Privilege one allows access in user EXEC mode and privilege 15 allows access in Privileged EXEC mode.

   Go to Advanced Mode:

   Admin: Users with Privilege 15 can execute all the **show**, **config**, and **exec** commands on the device. These users will have access to all GUI sections.

   Read Only: Users with Privileges one to 14 are considered read-only users. The default privilege is one if a user is created using the GUI. These users have access only to the Dashboard and the Monitoring sections.

   No Access: Users with Privilege zero can log in to the device through Telnet or SSH to access the CLI, but cannot access the GUI.

   Lobby Admin: Users who can create only guest user accounts. Lobby ambassadors can create and delete guest users and set parameters such as:
   - Password.
   - Lifetime of the guest user.
   - Guest role profiles (Quality-of-Service) profiles that should be applied on a guest using the AAA attribute list.
6. In the **Password** field, enter a password for the new account.
7. In the **Confirm Password** field, enter the same password again to confirm again.
8. Click **Apply to Device**.

Log in using the lobby account (CLI)

Execute the following commands before logging in using the lobby credentials:

Procedure

1. Enable NEW access control commands and functions.

   Example

   ```
   aaa new-model
   ```
2. Enter the local database.

Example

```
aaa authorization exec default local
```

3. Set HTTP server authentication method and use AAA access control methods.

Example

```
ip http authentication aaa
```

Log out from the administrator account. Then, log in using the lobby credentials. After logging in, the **Guest User** page appears.

Create a lobby ambassador account (CLI)

Create a lobby ambassador account using commands.

Add a new lobby ambassador (guest network administrator) to the system using CLI commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Create a user account.

Example

```
Device(config)# user-name user-name example-user
```

3. Specify the account type as lobby admin.

Example

```
Device(config-user-name)# type lobby-admin
```

4. Create a password for the lobby administrator account.

Example

```
Device(config-user-name)# password 0 password example-password
```

5. Create an attribute list for lobby admin access.

Example

```
Device(config-user-name)# aaa attribute list user-name example-user
```

6. Create an attribute type for lobby admin access.

   Example

   ```
   Device(config-user-name)# attribute type wlan-profile-name wlan_wl_mab
   ```

7. Return to the global configuration mode.

   Example

   ```
   Device(config-user-name)# exit
   ```

Configure WLAN (GUI)

Configure a WLAN using the GUI.

Set up a new WLAN on your wireless controller using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. In the **WLANs** window, click the name of the **WLAN** or click **Add** to create a new one. The **Add/Edit WLAN** window is displayed.
3. In the **Add/Edit WLAN** window, click the **General** tab to configure these parameters.
   - In the **Profile Name** field, enter or edit the name of the profile.
   - In the **SSID** field, enter or edit the SSID name.

     The SSID name is alphanumeric and can be up to 32 characters in length.
   - In the **WLAN ID** field, enter or edit the ID number. The valid range is between one and 512.
   - Select the **802.11** radio band from the **Radio Policy** drop-down list.
   - Using the **Broadcast SSID** toggle button, change the status to either **Enabled** or **Disabled**.
   - Using the **Status** toggle button, change the status to either **Enabled** or **Disabled**.
4. Click the **Security** tab, and then select the **Layer 2** tab to configure these parameters:
   - Select **None** from the **Layer 2 Security Mode** drop-down list. This setting disables Layer 2 security.
   - Enter the **Reassociation Timeout** value, in seconds. This value specifies the duration before a fast transition reassociation times out.
   - Check the **Over the DS** check box to enable Fast Transition over a distributed system.
   - Choose **OWE**. Opportunistic Wireless Encryption (OWE) provides data confidentiality with encryption over the air between an AP radio and a wireless client. OWE Transition Mode ensures backwards compatibility.
   - Choose Fast Transition (802.11r), the IEEE standard for fast roaming. This standard allows the initial handshake with a new AP to occur before the client roams to the target AP. This method is known as Fast Transition.
   - Check the check box to enable MAC filtering in the WLAN.
   - Check the **Lobby Admin Access** check box to enable Lobby Admin access.

5. Click **Save & Apply to Device**.

Client allowed list

Client allowed list details.

A client allowed list is a WLAN security feature that

- enables creation of an allowed list for clients on a particular WLAN or SSID-based MAC address, and
- is supported only with MAC addresses that do not include delimiters.

If you create a new client MAC address as an allowed list user with an invalid WLAN profile name, map the client MAC address to the WLAN profile carefully.

If you are a client in a university or hotel, you may need network access for a limited time. Many guests bring multiple devices. Protect networks from misuse or unauthorized access and allow legitimate clients to connect.

There are two types of administrator roles:

- Global Administrator: Creates a lobby admin user on the controller and enables each lobby administrator to access the WLAN.
- Lobby Administrator: Adds or deletes clients from the allowed list using the GUI to manage association to a WLAN or SSID. Existing lobby administrators can also configure the allowed list.

Restrictions for client allowed list

Client allowed list restrictions.

A lobby admin can add clients to the allowed list only through the GUI, not through the CLI.

Creating a client allowed list (GUI)

Create a client allowed list using GUI.

As a lobby administrator, you can use several methods to create an allowed list for valid WLAN users.

Adding single MAC address to allowed list

Add single MAC address to allowed list using the GUI.

Import a specific MAC address into your WLAN whitelist efficiently using the admin portal.

Procedure

1. Log in to the Lobby Admin portal.
2. Click **Whitelist Users**.
3. Select **WLAN** from the drop-down list.
4. Click **Add New Whitelist User**.
5. Click **By MAC Address** radio button.
6. Enter the **MAC address** and **Description**.
7. Click **Apply to Device**.

Adding bulk MAC address to allowed list

Add bulk MAC address to allowed list using the GUI.

Import multiple MAC addresses into your WLAN whitelist efficiently using the admin portal.

Procedure

1. Log in to the Lobby Admin portal.
2. Click **Whitelist Users**.
3. Select the WLAN from the drop-down list.
4. Click **Add New Whitelist User**.
5. Click **Bulk Import** radio button.
6. Select the CSV file that lists the clients in MAC address, description format.
7. Click **Apply to Device**.

Manage guest users (GUI)

Manage guest users using the GUI.

Add and manage guest user access through the controller GUI.

Procedure

1. Log in to the Lobby Admin portal using the lobby admin credentials.
2. Click **Whitelist Users**.
3. Select the corresponding **WLAN** from the **WLAN** drop-down list.
4. Select **Onboarding** from the **WLAN Mode** to enable clients to access the network.
5. Click **Apply**.
6. Select a MAC address from the **Connected/Not Whitelisted** in the Whitelist window. Once the clients join the controller, the MAC addresses are listed in the **Connected/Not Whitelisted**. In the Onboarding mode, MAC filtering in the selected WLAN is disabled. In such a scenario, change the mode using **Secure** mode.
7. Select **Secure** to automatically add the clients that are connected to the allowed list. In the secure mode, MAC filtering in the selected WLAN is enabled.
8. Click **Apply to Device**.

   The clients are listed in the **Connected/Whitelisted**.

View a client allowed list

View a client allowed list using the GUI.

See which clients are connected or whitelisted on a WLAN.

Procedure

1. Log in to the Lobby Admin portal.
2. Click **Whitelist Users**.

3. Select the corresponding **WLAN** from the **WLAN** drop-down list.

   The window lists the following information:

   - **Connected/Whitelisted**: Lists the clients that are connected and added to the allowed list by the Lobby admin.
   - **Connected/Not Whitelisted**: Lists the clients that are connected, but not added to the allowed list by the Lobby admin.
   - **Not Connected/Whitelisted**: Lists the clients that are not connected, but added to the allowed list.

72 Guest User Accounts

Topics:

- Create guest user accounts

This chapter explains how administrators and lobby ambassadors create, configure, verify, and assign temporary guest user accounts for WLAN access using GUI and commands, including lifetime, simultaneous-login limits, and WLAN-specific username binding.

Create guest user accounts

Explains guest user accounts, which provide temporary, restricted access to wireless networks for visitors through accounts configured by administrators or designated lobby ambassadors.

A guest user account is a user account type that

- grants temporary access to a WLAN or SSID
- is created by a network administrator or lobby ambassador, and
- automatically expires after a configured duration.

The controller can provide guest user access on WLANs. You must create guest user accounts to enable this access. The lobby ambassador has limited configuration privileges and can access only the web pages used to manage guest user accounts.

The lobby ambassador can specify how long guest user accounts remain active.

You can associate a user name with a WLAN profile name to restrict guest users to a specific WLAN.

Prerequisites for guest users

- Guest users are created by administrator or lobby ambassador.
- Guest user should be role-based.
- Guest users should be able to connect to the network and access the internet.
- Guest users configured locally or remotely (through RADIUS or TACACS) do not have access to the device through Telnet, SSH, or WebUI.

Create a guest user account (GUI)

Learn how to create a guest user account using the web interface, including specifying credentials, configuring session parameters, and saving the account for device access.

Enable guest access by adding a new user account in the system through the GUI.

Procedure

1. Choose **Configuration** > **Security** > **Guest User**.
2. On the **Guest User** page, click **Add**.
3. Enter a user name, password, and description for the new account. Check the **Generate password** check box to automatically generate a password.
4. Enter the number of simultaneous user logins. The valid values range between 0 to 64.

   Enter zero for unlimited users.
5. In the **Lifetime** section, choose the number of years, months, days, hours, and minutes.
6. Click **Save & Apply to Device**.

Create a guest user account (CLI)

Learn how to create a guest user account on the device using commands.

Enable temporary or guest access to your network by creating a guest user account with specific attributes and access limitations using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a guest user account.

   Example

   ```
   Device(config)# user-name guest-user-name
   ```

3. Specify the account type as guest user account.

   Example

   ```
   Device(config-user-name)# type network-user description description guest-user
    max-login-limit number of simultaneous logins lifetime year yy month mm day
    day hour hour minute minute second second
   ```

   > **Note**
   >
   > User configuration values must be equal to or lower than the global values, since global values take precedence.
   >
   > The command **max-user-login** configured at the global level, overrides the session limit configured at the local user level. The value range is from 0 to 8.

4. Create a password for the guest user account.

   Example

   ```
   Device(config-user-name)# password 0 password
   ```

   If FIPS mode is enabled, configure the guest user's password with an encryption level greater than six by entering the command **algorithm-type sha256 secret <Password>**. If you set a lower encryption level, the guest user will not be able to authenticate.

5. Create a AAA attribute list to apply QoS profiles on the guest user account.

   Example

   ```
   Device(config-user-name)# aaa attribute list aaa-attribute-list-name
   ```

6. Return to the global configuration mode.

Example

```
Device(config-user-name)# exit
```

Note

If the lobby admin is local, enter this command:

```
aaa authentication login default local
```

If the lobby admin is a remote user, enter these commands:

```
aaa authentication login default group radius/tacacs
```

```
aaa remote username <remote-lobby-admin-name>
```

In case of local or remote lobby, enter this command to map the authorization policies:

```
aaa authorization exec default local
```

Verify guest user account

This section lists the commands and output fields for verifying guest user accounts using the CLI.

```
Device# show aaa local guest_user all

User-Name           : new4
Type                :  GUEST USER
Password            : *
Is_passwd_encrypted : No
Attribute-List      : Not-Configured
Viewname            : Not-Configured
Lobby Admin Name    : NEW_LOBBY_ADMIN
Max Login Limit     : 0
Description         : guest
Start-Time          : 07:56:39 IST Jan 25 2019
Lifetime            : 1 years 0 months 0 days 0 hours 0 mins 0 secs
Expiry-Time         : 07:56:39 IST Jan 20 2020 Remaining Lifetime  : 0 years
11 months 29 days 22 hours 52 mins 49 secs
```

To verify a specific guest user account, use this command:

```
Device# show aaa local guest_user new_guest3

User-Name           : new_guest3
Type                :  GUEST USER
Password            : *
Is_passwd_encrypted : No
```

```
Attribute-List      : Not-Configured
Viewname            : Not-Configured
Lobby Admin Name    : INVALID_ADMIN
Max Login Limit     : 9
Description         : new
Start-Time          : 04:39:01 IST Feb 4 2019
Lifetime            : 1 years 0 months 0 days 0 hours 0 mins 0 secs
Expiry-Time         : 04:39:01 IST Jan 30 2020
Remaining Lifetime  : 0 years 11 months 11 days 21 hours 16 mins 34 secs
```

Assign username to guest users in a WLAN (CLI)

Use this procedure to assign a unique username to a guest user for a specific WLAN profile using commands.

Identify and authenticate guest devices on a WLAN by assigning usernames bound to their MAC addresses using commands.

Before you begin

- If a WLAN profile name is configured for a user, guest user authentication is allowed only from that WLAN.
- If a WLAN profile name is not configured for a user, guest user authentication is allowed on any WLAN.
- To operate in connected mode, configure the AAA policy override under both SSID policies before you assign a username to a guest user on a WLAN.

Procedure

1. Enter the configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Assign a username to the WLAN profile.

   Example

   ```
   Device# username user_name mac wlan-profile-name profile_name
   ```

   Note

   The wlan-profile-name per user is applicable for MAC type users.

3. Display the values of the WLAN profile.

   Example

   ```
   Device# show aaa local guest_user new_guest3
   ```

4. Return to the privileged EXEC mode.

Example

```
Device# end
```

73 Link Local Bridging

Topics:

- Link local bridging

Link local bridging

Introduces Link Local Bridging, explaining how it enables roaming wireless clients to access local services at their point of attachment, maintains VLAN continuity across intra- and inter-controller roaming, and describes its feature history and use cases for enhanced local service discovery.

Link local bridging is a wireless networking mechanism that

- allows roaming wireless clients to use local services at their current point of attachment (PoA)
- preserves access VLAN continuity during intra-controller and inter-controller roaming, and
- reduces the need to tunnel client packets to their original anchor controller.

Feature history

Table 82: Feature history for link local bridging

Feature	Release	Feature Information
Link Local Bridging	Cisco IOS XE 17.6.1	The Link Local Bridging feature allows you to manage link-local traffic in intercontroller and intracontroller roaming scenarios. In Cisco IOS XE Bengaluru 17.5.x and earlier releases, only mDNS multicast link-local frames are forwarded.

Intra-controller and inter-controller roaming scenario

In Cisco IOS XE 17.5.1 and earlier releases, client packets were forwarded through the access VLAN of a client. The client also received all the IPv4 packets or all the IPv6 packets from its assigned access VLAN.

When an L3 client roamed from one controller to another controller, the point-of-presence (PoP) remained with the first controller, also known as the anchor controller or the home controller, and the point-of-attachment (PoA) moved to the second controller, also known as the foreign controller or the visited controller. In this anchor-foreign scenario, the client packets were tunneled back to the anchor controller to be forwarded on the access VLAN of the client.

Similarly, in case of L3 intra-controller roaming, when the feature Roaming Across Policy Profile is enabled, the client access VLAN is maintained, regardless of the policy profile VLAN. In such a scenario, the PoA becomes the destination policy profile VLAN.

A roaming wireless client benefits more from local services available near its PoA than from discovering services at its PoP.

Therefore, from Cisco IOS XE 17.6.1 onward, you can manage these intra-controller and inter-controller roaming scenarios with the Link Local Bridging feature. Link Local Bridging is disabled by default.

Use case

Enhancing Local Service Discovery for Roaming Clients with Link Local Bridging: If you use a local mode deployment and L3 roaming to manage clients moving across different physical locations, the Link Local Bridging feature allows wireless clients to discover nearby services, such as those advertised through mDNS, that are physically close to them. This ensures that, even as clients roam across subnets, they can still find and access local network services in their immediate vicinity.

Guidelines and restrictions for link local bridging

Link Local Bridging guidelines and restrictions.

- The Link Local Bridging feature is supported in local mode or FlexConnect central switching.
- Only mDNS bridge mode is supported with Link Local Bridging.
- Guest profiles are not supported.
- Wired Guest LAN, Remote LAN (RLAN), and Inter-Release Controller Mobility (IRCM) are not supported.
- Mesh and IP Source Guard (IPSG) are not supported when the Link Local Bridging feature is enabled.
- Enable Link Local Bridging on both the anchor and foreign controllers to support roaming.
- Access VLAN and bridge VLAN must be operational to ensure the Link Local Bridging feature functions properly.
- Enable Link Local Bridging on all policy profiles that use the same SSID.
- Configure wireless multicast-over-multicast (**wireless multicast** *multicast IP address*) before enabling the Link Local Bridging feature. The **wireless multicast link-local** command is enabled by default when wireless multicast is enabled.

Enable link local bridging per policy profile (GUI)

Enable Link Local Bridging Per Policy Profile using GUI.

Enable link local traffic bridging for network policy profile using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**
2. Click **Add**.

   The **Add Policy Profile** page is displayed.
3. In the **Add Policy Profile** page, under the **General** tab, enter the name of the policy profile.
4. In the **Advanced** tab, check the **Link-Local Bridging** check box to enable link-local bridging on the policy profile.
5. Click **Apply to Device**.

The device configuration applies the policy profile with link-local bridging enabled.

Enable link local bridging per policy profile (CLI)

Enable Link Local Bridging Per Policy Profile using commands.

Enable link local bridging for wireless clients by configuring a policy profile on the device.

Before you begin

Ensure that both wireless multicast-over-multicast and wireless multicast link-local settings are enabled.

> **Note**
>
> In Cisco IOS XE Bengaluru 17.6.1 and later releases, the wireless multicast link-local setting is enabled by default when multicast is enabled. All downstream multicast link-local frames are forwarded to wireless clients. In Cisco IOS XE Bengaluru 17.5.x and earlier releases, only mDNS multicast link-local frames are forwarded.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create policy profile for the WLAN.

   Example

   ```
   Device(config)# wireless profile policy wireless-profile-name
   ```

3. Enable link-local bridging for the policy profile.

   Example

   ```
   Device(config-wireless-policy)# link-local-bridging
   ```

Link-local bridging is enabled for the specified policy profile.

```
Device# configure terminal
Device(config)# wireless profile policy wireless-profile1
Device(config-wireless-policy)# link-local-bridging
```

Verify link local bridging

Verify Link Local Bridging configurations.

To verify the configuration status of Link Local Bridging, use this command:

```
Device# show wireless profile policy detailed policy1
Policy Profile Name                 : policy1
Description                         :
Status                              : ENABLED
VLAN                                : 81
Multicast VLAN                      : 0
OSEN client VLAN                    :
Multicast Filter                    : DISABLED
QBSS Load                           : ENABLED
Passive Client                      : DISABLED
ET-Analytics                        : DISABLED
StaticIP Mobility                   : DISABLED
```

```
WLAN Switching Policy
  Flex Central Switching            : ENABLED
  Flex Central Authentication       : ENABLED
  Flex Central DHCP                 : ENABLED
  Flex NAT PAT                      : DISABLED
.
.
.
  ----------------------------------------------------
mDNS Gateway
  mDNS Service Policy name          : default-mdns-service-policy
User Defined (Private) Network              : Disabled
User Defined (Private) Network Unicast Drop  : Disabled
Policy Proxy Settings
  ARP Proxy State                   : DISABLED
  IPv6 Proxy State                  : None
Airtime-fairness Profile
  2.4Ghz ATF Policy                 : default-atf-policy
  5Ghz ATF Policy                   : default-atf-policy
Link-local bridging                 : ENABLED
```

To verify if Link Local Bridging VLAN is included, use this command:

```
Device# show wireless client mac 7xxx.3xxx.3xxx detail
Client MAC Address : 7xxx.3xxx.3xxx
.
.
.
Link-local bridging VLAN: 3
.
.
.
WiFi Direct Capabilities:
  WiFi Direct Capable           : No
```

To verify if link local multicast traffic is enabled, use this command:

```
Device# show wireless multicast
Multicast                               : Disabled
AP Capwap Multicast                     : Unicast
Wireless Broadcast                      : Disabled
Wireless Multicast non-ip-mcast         : Disabled
Wireless Multicast link-local           : Enabled
```

74 Web Admin Settings

Topics:

- Web admin settings

This chapter guides administrators through configuring web-based management: enabling HTTP/HTTPS and PIV authentication, installing CA trustpoints, enabling NETCONF/YANG, setting session timeouts, and managing VTY lines and access protocols for secure remote administration.

Web admin settings

Explains the configuration options and roles of web admin settings, which control access and management of a controller's web interface.

A web admin setting is a controller management configuration that

- determines how administrators access the controller's web interface
- specifies the protocols and interfaces available for remote management, and
- enables customization of user access, source interfaces, and file transfer protocols.

These tasks include setting up the controller for communication with other devices in the network and configuring the management interface to connect over IP.

Use the **Administration** > **Management** > **HTTP/HTTPS/Netconf/VTY** page to configure system-wide settings.

Configure HTTP/HTTPS access (GUI)

Use this task to enable and configure HTTP, HTTPS, and Personal Identity Verification (PIV) access for secure web management of your controller.

Securely grant users access to the controller's WebUI using HTTP, HTTPS, and two-factor authentication with PIV smart cards.

HTTP or HTTPS access allows users to access the controller's WebUI using its IP address. You can allow users to connect securely over HTTPS or over HTTP, which is not a secure connection.

Use the **Administration** > **Management** > **HTTP/HTTPs/Netconf/VTY** page to configure secure access to the controller.

Before you begin

- Ensure you have administrator privileges to configure the controller.
- Obtain the CA-signed server certificate and trustpoint configuration if you are enabling PIV authentication.
- Collect the client certificate signed by your CA for browsers using PIV.

Procedure

1. Enable **HTTP Access** then enter the port number for HTTP requests. The default port is 80, but you can also use port 80 or any value from 1025 to 65535.
2. Enable **HTTPS Access** on the device and enter the designated port to listen for HTTPS requests. The default port is 1025. You may use port 443 or any port between 1025 and 65535.

   Enabling HTTPS access allows users to access the controller's GUI using "https://ip-address". On a secure HTTPS connection, data to and from an HTTPS server is encrypted before being sent over the Internet. SSL encryption provides a secure connection for tasks such as configuring a switch from a web browser.
3. Enable Personal Identity Verification (PIV) for two-factor authentication.

   This method lets users access the WebUI with PIV-compatible smart cards, enabling login without a password. To use this authentication method, configure the trustpoint and CA server certificate on the device. Also, ensure that the browser has the client certificate signed by the CA server. If you do not provide the client certificate, access to the UI is denied.
4. Set the **Personal Identity Verification Authorization only** option to *Enabled*. This authorizes a user's permissions and restrictions based on a remote TACACS+ or RADIUS security server.

5. Click **Apply** to save the configuration.

> **Note**
>
> To use Personal Identity Verification (PIV) for two-factor authentication on Safari, perform these steps:
>
> a. Open Safari browser and go to **Settings > Advanced**
>
> 1. Check the **Show Develop in menu bar** check box. This enables the Develop option in the top menu bar.
> 2. Click **Develop**, and from the dropdown, select **Empty Caches**.
>
> b. Open the web url to login.

Configure HTTP trust point (GUI)

Learn how to configure an HTTP trust point to secure web administration on your device using a certificate authority (CA) trustpoint.

Use Secure HTTP (HTTPS) web administration on your device with a CA-signed certificate instead of a self-signed certificate.

Certificate authorities (CAs) manage certificate requests and issue certificates to participating network devices. These services provide centralized security key and certificate management for the participating devices. Specific CA servers are referred to as trustpoints. When a connection attempt is made, the HTTPS server provides a secure connection by issuing a certified X.509v3 certificate, obtained from a specified CA trustpoint, to the client. The client (usually a Web browser), in turn, has a public key that allows it to authenticate the certificate. For secure HTTP connections, we highly recommend that you configure a CA trustpoint. If a CA trustpoint is not configured for the device running the HTTPS server, the server certifies itself and generates the needed RSA key pair. Because a self-certified (self-signed) certificate does not provide adequate security, the connecting client generates a notification that the certificate is self-certified, and the user has the opportunity to accept or reject the connection. This option is useful for internal network topologies (such as testing).

If you do not configure a CA trustpoint and then enable a secure HTTP connection, the device automatically generates either a temporary or a persistent self-signed certificate for the secure HTTP server or client. If the device is not configured with a hostname and domain name, it generates a temporary self-signed certificate. If the switch reboots, it deletes any temporary self-signed certificate and assigns a new one. If the device has been configured with a hostname and domain name, it generates a persistent self-signed certificate. This certificate remains active if you reboot the device or disable the secure HTTP server. The certificate will be available when you re-enable a secure HTTP connection.

Use the **Trust Point Configuration** section of the **Administration** > **Management** > **HTTP/HTTPs/Netconf/VTY** page to make these changes.

Before you begin

You must have configured a trustpoint for web administration purposes.

Procedure

1. Tap to enable the trust point.
2. Select the appropriate trust point from the drop-down list to use for web administration purposes.

   If you have not configured a trust point earlier, you can navigate to the appropriate page and first configure it.

3. Click **Apply** to save the configuration.

Configure NETCONF YANG (GUI)

Use this task to configure NETCONF YANG access on your device through the GUI, including authentication and port settings.

Configure and enable NETCONF YANG to allow programmatic network management and device configuration using commands.

NETCONF provides a mechanism to install, manipulate, or delete the configuration of network devices.

If the NETCONF connection is configured to use AAA for authentication purposes, it uses only the default method list and cannot be pointed to use any other named method list.

Use the **Netconf Yang Configuration** section of the **Administration** > **Management** > **HTTP/HTTPs/Netconf/VTY** page to make these changes.

Procedure

1. Enable NETCONF.
2. Enter the SSH port number that will be used to facilitate communication between a client and a server. The default port is 830.
3. Click **Apply** to save the configuration.

Configure timeout policy (GUI)

Use this task to set session timeout limits for management connections, including HTTP, session idle, and server lifetime policies, to control how long sessions remain active before requiring reauthentication.

Define the duration a management session can remain idle or active before automatic closure. This improves security and resource management.

The Timeout Policy Configuration allows you to set the idle interval for management sessions. Once the timeout is reached, you must log in again to reestablish the connection.

Use the **Timeout Policy Configuration** section of the **Administration** > **Management** > **HTTP/HTTPs/Netconf/VTY** page to make these changes.

Procedure

1. Enter the maximum number of seconds a connection to the HTTP server remains open before timing out in the **HTTP Timeout-policy** field. Once the time value is reached, you must log in again to reestablish the connection.
2. In the **Session Idle Timeout** field, enter the maximum number of seconds the connection remains open if no data is received or if response data cannot be sent.

   Note that this value may not affect already existing connections. If the server is too busy, or if the limit on the life time or number of requests is reached, the connection may close sooner. The default value is 180 seconds (three minutes).

3. In the **Server Life Time** field, enter the maximum number of seconds the connection remains open from when the connection is established.

   Note that the new value might not affect existing connections. If the server is too busy or if the limit on idle time or number of requests is reached, the server may close the connection sooner. The server does not close the connection while actively processing a request. Therefore, the connection might remain open longer than the specified lifetime

if processing continues when the limit is reached. In this situation, the connection will close after processing finishes. The default value is 180 seconds (three minutes). The maximum value is 86400 seconds (24 hours).

4. Enter a value for the maximum limit on the number of requests processed on a persistent connection before it is closed in the **Max Number of Requests** field.

   Note that the new value might not affect existing connections. If the server is too busy, or if the limit on idle time or life time is reached, the connection may close before the maximum number of requests are processed. The default value is one. The maximum value is 86400.

5. Click **Apply** to save the configuration.

Configure VTY (GUI)

Use this task to configure Virtual Terminal (VTY) lines for secure remote access (SSH or Telnet) to your device using the web interface.

Configure the number of simultaneous remote CLI connections and apply security policies to inbound device sessions.

VTY is a virtual port that supports Telnet or SSH access for inbound device connections. You can configure the number of simultaneous connections to your device and add security to validate these connections.

Use the **VTY** section of the **Administration** > **Management** > **HTTP/HTTPs/Netconf/VTY** page to make these changes.

Procedure

1. Set the number of VTY lines to specify how many users can access the device remotely at the same time.

   Virtual Terminal Lines, also called Virtual TeleType (VTY), provide remote access to the controller's CLI without the need to physically connect a laptop to the controller console. The number of VTY lines determines the maximum number of simultaneous connections allowed. Setting the number to a value between zero and 50 allows up to fifty simultaneous Telnet or SSH sessions to the controller. The default is set at 15. We recommend increasing the number of VTY lines to 50 to prevent connectivity disruptions when multiple users access the device.

2. Select the protocol for the remote connection from the **VTY Transport Mode** drop-down list. You can split the connections based on protocol. For example, lines zero to five might allow SSH connections, and lines 10 to 20 might allow Telnet connections.
3. Optional: You can add security in the WebUI to validate login requests. To configure AAA authentication and authorization for inbound sessions to VTY lines on your system, first configure a RADIUS or TACACS+ authentication server. Then, select the authentication and authorization list from the appropriate drop-down menus.
4. Click **Apply** to save the configuration.

Post-quantum cryptography support

Post-quantum cryptography (PQC) support is a security enhancement for web server applications, such as Nginx that

- enables quantum-resistant key exchange within TLS 1.3
- protects data against "harvest now, decrypt later" attacks, and
- maintains backward compatibility with legacy clients using classical cryptographic algorithms.

Quantum-resistant security provides future-proof protection for enterprise and government communications by integrating schemes like ML-KEM into the TLS handshake process. ML-KEM reduces exposure to harvest-now-decrypt-later attacks, in which an attacker stores encrypted traffic for possible decryption by a future quantum computer.

As legacy TLS protocols cannot negotiate post-quantum cryptography (PQC) cipher suites, they remain vulnerable to future decryption threats. This feature allows web servers to deploy quantum-resistant algorithms, such as ML-KEM, alongside traditional cryptographic methods. In non-FIPS environments, the system supports a wide range of PQC, hybrid, and classic groups. In FIPS-compliant environments, PQC-only groups are disabled, and only specific hybrid and classical groups are permitted to ensure regulatory compliance.

Supported key-exchange groups

Non-FIPS mode

These PQC-only groups are supported:

- `mlkem512`
- `mlkem768`
- `mlkem1024`

These hybrid groups are supported:

- `p256_mlkem512`
- `p384_mlkem768`
- `p521_mlkem1024`
- `x25519_mlkem512`
- `x448_mlkem768`
- `X25519MLKEM768`
- `SecP256r1MLKEM768`
- `SecP384r1MLKEM1024`

The classical groups are:

- `P-256`
- `P-384`
- `P-521`

By default, the Nginx web server offers all supported PQC-only, hybrid, and classical groups.

FIPS mode

PQC-only groups are not supported in FIPS mode. These hybrid groups are supported:

- `p256_mlkem512`
- `p384_mlkem768`

The complete group set supported in FIPS mode is:

- `p256_mlkem512`
- `p384_mlkem768`
- `P-256`
- `P-384`

- `P-521`

PQC restrictions

While Post-Quantum Cryptography (PQC) provides enhanced security against future quantum threats, you must account for these limitations and requirements when deploying this feature:

- PQC key exchange groups are supported exclusively within **TLS 1.3**. Applications or clients restricted to TLS 1.2 or older cannot utilize PQC key exchange and will continue to use classic groups.
- PQC-only groups (such as `mlkem512`, `mlkem768`, `mlkem1024`) are not supported in FIPS mode.
- In FIPS mode, only a subset of hybrid groups (`p256_mlkem512`, `p384_mlkem768`) and specific classical groups (`P-256`, `P-384`, `P-521`) are allowed.
- Successful negotiation of PQC or hybrid key exchange requires both the client and the server to be PQC-compliant. If the client does not support these algorithms, the connection will fall back to classical groups.
- Initial TLS handshakes may experience increased latency due to the additional computational complexity of PQC algorithms like ML-KEM.
- In high-frequency connection scenarios, total connection throughput may be impacted, particularly on hardware with limited resources.
- This feature applies to web server applications (for example, WebUI). Other TLS-based applications, such as Web-auth, are not impacted by these PQC configurations.

Configure post-quantum cryptography support (CLI)

Provides instructions for enabling or disabling PQC algorithms on web servers using the CLI, including selecting between all, hybrid, non-PQC, and PQC types to set the cryptographic security level for TLS 1.3 connections.

Enable or disable Post-Quantum Cryptographic (PQC) algorithms for web server applications.

Use this task to define the cryptographic security level for web traffic. By default, the system supports all algorithms, including PQC, hybrid, and classic groups.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the PQC type for the HTTPS server.

   Example

   ```
   Device(config)# ip http secure-pqc-type {all | hybrid | non-pqc | pqc}
   ```

   The PQC types include:

   - **all**: Specifies that all PQC, non-PQC, and hybrid algorithms are supported.
   - **hybrid**: Specifies that the supported type is hybrid cryptographic algorithms (Key derived with PQC and non-PQC).
   - **non-pqc**: Specifies that the supported type is classic cryptographic algorithms.

- **pqc**: Specifies that the supported type is post-quantum cryptographic algorithms.

3. Exit global configuration mode.

Example

```
Device(config)# exit
```

The web server updates its security configuration to use the selected cryptographic algorithms for TLS 1.3 connections.

Verify secure key-exchange configuration

Confirm the current status and enabled cryptographic groups for the secure web server.

To validate your PQC configuration is active and to identify which algorithms are currently supported by the server, use this command:

```
Device# show ip http server secure status
HTTP secure server status:
 HTTP secure server ECDHE curve: secp384r1
 HTTP secure server PQC type: all
```

75 Web UI Configuration Command Accounting in TACACS Server

Topics:

- Web UI configuration command accounting in TACACS+ server

This chapter explains how to configure TACACS+ accounting for controller web UI configuration changes, showing CLI steps to enable AAA command accounting (default and HTTP named method lists) for centralized audit logging.

Web UI configuration command accounting in TACACS+ server

Explains how Web UI configuration command audits enable traceability and accountability for configuration changes made through the controller GUI, by logging these changes to a TACACS+ server.

A Web UI configuration command audit is a security feature that

- records configuration changes executed through the controller's GUI
- logs audit information to a TACACS+ server for centralized tracking, and
- includes details such as the executed command, the user responsible, and session parameters.

The Cisco Catalyst 9800 series wireless controller configuration is stored in databases.

Feature history for Web UI configuration command accounting in TACACS+ server

Web UI Configuration Command Accounting in TACACS+ Server feature history details.

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 83: Feature history for Web UI configuration command accounting in TACACS+ server

Release	Feature	Feature Information
Cisco IOS XE Cupertino 17.9.1	Logging Web UI-Based Configuration Changes in TACACS+ Server	This feature logs all configuration changes made in controller web UI. Support for logging configurations done in IOS console in TACACS+ server is already available.

Guidelines for Web UI configuration command accounting in TACACS+ server

This topic outlines the best practices for enabling and managing configuration command accounting for Web UI sessions in a TACACS+ server environment.

- By default, configuration commands are not logged to the TACACS+ server unless command accounting is configured.
- All commands are accounted for only when AAA default command accounting is configured for privilege level 15.
- If AAA default command accounting is not configured, and you want commands to be logged to the TACACS+ server, use these methods:
  1. The HTTP named method list command accounting.
  2. The AAA named method list (same as the one configured in **Step 1**) command accounting.

Configure AAA accounting using default method list (CLI)

Use this task to enable and configure AAA command accounting to monitor and record user command activity using the default method list on your device.

Use this task to monitor and record user command activity on devices through AAA accounting features.

Configure AAA accounting to track user commands executed on a controller, leveraging the default accounting method. This supports compliance and security needs.

Before you begin

- Confirm that AAA is enabled on the device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create an accounting method list and enables accounting.

   Example

   ```
   Device(config)# aaa accounting commands 15 default start-stop group group-name
   ```

   - *privilege_level*: AAA accounting level. The valid range is from zero to 15.
   - *group-name*: AAA accounting group that supports only TACACS+ group.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The controller records user command activities according to the configured accounting method.

Configure HTTP command accounting using named method list (CLI)

Use this task to configure HTTP command accounting on your device, allowing you to monitor and record user command activity with AAA features using a named method list.

Set up command accounting to track user actions on network devices via HTTP with a specified AAA method list using commands.

HTTP command accounting provides auditing and compliance by recording commands executed by users. Using a named method list offers flexibility for different accounting requirements.

Before you begin

- Ensure AAA accounting is enabled on your device.
- Have a predefined AAA accounting method list (if not, configure one).

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure HTTP command accounting using the named method list.

   Example

   ```
   Device(config)# ip http accounting commands 1 oneacct
   ```

   - *level*: The privilege value ranges from zero to 15. By default, the command privilege levels available on the controller are:
     - *0*: Includes the **disable**, **enable**, **exit**, **help**, and **logout** commands.
     - *1*: Includes all the user-level commands at the controller prompt (>).
     - *15*: Includes all the enable-level commands at the controller prompt (>).
   - *named-accounting-method-list*: Name of the predefined command accounting method list.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The device records user command activities in accordance with the configured accounting method.

76 Embedded Packet Capture

Topics:

- Embedded packet capture

This chapter explains how to configure and verify this troubleshooting feature on Cisco IOS XE Dublin 17.12.1, including increased buffers, continuous captures, filtering multiple inner MAC addresses, and exporting capture files for analysis.

Embedded packet capture

Explains embedded packet capture as a feature for troubleshooting and monitoring network packet flows directly on Cisco devices.

An embedded packet capture is a network troubleshooting feature that

- enables administrators to trace and analyze data packets in real time
- allows capture of packets flowing through, to, and from a Cisco device, and
- supports filtering, buffer management, and packet analysis for diagnosing complex network issues.

The Embedded Packet Capture on the controller is used for troubleshooting multiple issues, such as authentication failures with RADIUS, AP joining or disconnection, client forwarding, disconnection, and roaming. It can also help with specific features, including multicast, mDNS, Umbrella, and mobility. When troubleshooting an AP join or client onboarding issue, you might lose important information if you are unable to stop the capture immediately when the issue occurs. In most cases, a buffer of 100 MB is not sufficient for data capture. Moreover, the existing Embedded Packet Capture feature supports only the filtering of one inner MAC address, which captures the traffic of a specific client. At times, it is difficult to pinpoint which wireless client is experiencing an issue.

Beginning with Cisco IOS XE Dublin 17.12.1, the Embedded Packet Capture feature supports increased buffer size, continuous capture, and filtering of multiple MAC addresses in one session. You cannot configure the Embedded Packet Capture enhancement using the GUI.

Feature history for embedded packet capture

Embedded Packet Capture feature history details.

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 84: Feature history for embedded packet capture

Release	Feature	Feature Information
Cisco IOS XE Dublin 17.12.1	Embedded Packet Capture	The embedded packet capture feature is enhanced to support increased buffer size, continuous capture, and filtering of multiple MAC addresses in one embedded packet capture (EPC) session.

Configure embedded packet capture (CLI)

Use this procedure to configure embedded packet capture (EPC) using CLI commands, enabling you to capture, filter, and export network packet data for analysis.

Configure and manage embedded packet capture on your Cisco device using commands to analyze live packet traffic and gather data for troubleshooting.

With the Embedded Packet Capture feature enhancement, the buffer size is increased from 100 MB to 500 MB.

> **Note**
>
> Buffer is of memory type. You can either maintain a memory buffer or copy the memory buffer that is present in a file to store more information.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

   Enter your password, if prompted.

2. Configure the Gigabit Ethernet interface for inbound, outbound, or both inbound and outbound packets.

   Example

   ```
   Device# monitor capture epc_session-name interface GigabitEthernet
   interface-number {both | in | out}
   ```

   For Cisco 9800-CL controllers, use Gigabit interfaces such as Gi1, Gi2, or Gi3. For physical controllers, specify the port channel if one is configured. Supported physical interfaces include Te or Tw.

   > **Note**
   >
   > To capture packets punted to the CPU, run the **control-plane** command.

3. Optional: Configure monitor capture limit, in seconds.

   Example

   ```
   Device# monitor capture epc_session-name limit duration limit-duration
   ```

4. Optional: Configures the file in circular buffer. (The buffer can be circular or linear.)

   Example

   ```
   Device# monitor capture epc_session-name buffer circular file no-of-files
   file-size per-file-size
   ```

   When **circular** is configured, the files work as a ring buffer. The value range of the number of files to be configured is from two to five. The file size can range from 1 MB to 500 MB.

   The **buffer** command supports various keywords, such as **circular** , **file** , and **size**. The **circular** command is optional.

> **Note**
>
> A circular buffer is required for continuous capture.
>
> This step generates swap files on the controller. Swap files are not packet capture (PCAP) files and cannot be analyzed directly. When you run the **export** command, the swap files are combined and exported as one PCAP file.

5. Configure inline filters and configure a monitor capture specifying an access list as the filter for the packet capture.

   Example

   ```
   Device# monitor capture epc_session-name match {any | ipv4 | ipv6 | mac |
   pklen-range}
   ```

   ```
   Device# monitor capture epc_session-name access-list access-list-name
   ```

   > **Note**
   >
   > You can configure filters and ACLs.

6. Optional: Configure continuous packet capture.

   Example

   ```
   Device# monitor capture epc_session-name continuous-capture http:
   location/filename
   ```

   Enable the automatic export of files to a specific location before the buffer is overwritten.

   > **Note**
   >
   > - A circular buffer is required for continuous capture.
   > - Configure the filename with a .pcap extension.
   > - An example of the filename and nomenclature used to generate the filename is as follows:
   >   CONTINUOUS_CAP_20230601130203.pcap
   >
   >   CONTINUOUS_CAP_20230601130240.pcap
   > - After packets are exported automatically, the buffer is not cleared until it is overwritten by new incoming capture packets, cleared, or deleted using commands.

7. Optional: Configure up to 10 MAC addresses as inner MAC filter.

Example

```
Device# monitor capture epc_session-name inner mac [ MAC1 | MAC2 |
MAC3......... ]
```

```
Device# no monitor capture epc_session-name inner mac [ MAC1 | MAC2 |
MAC3......... ]
```

Note

- You cannot modify the inner MAC addresses while the capture is in progress.
- You can enter the MAC addresses in a single command or over multiple command lines. Because of the character string limitation, you can enter only five MAC addresses in a single command line. To enter the remaining MAC addresses, use the next command line.
- If you have configured 10 inner MAC addresses, you must delete an existing address before adding a new one.

8. Start capture of packet data and stop capture of packet data.

Example

```
Device# no monitor capture epc_session-name start
```

```
Device# no monitor capture epc_session-name stop
```

9. Export captured data for analysis when continuous capture is not configured.

Example

```
Device# monitor capture epc_session-name export filelocation/filename
```

Verify embedded packet capture

This section lists the commands and output fields relevant to verifying embedded packet capture (EPC) status, configuration, and buffer contents on a device.

To view the configured file number and per file size, run this command:

Note

This command is displayed irrespective of whether continuous capture is enabled or not. The configured inner MAC addresses are also displayed using this command.

```
Device# show monitor capture epc_session1
Status Information for Capture epc_session1
  Target Type:
 Interface: TwoGigabitEthernet0/0/0, Direction: BOTH
   Status : Inactive
  Filter Details:
    Capture all packets
                Inner Filter Details:
                Continuous capture: enabled
                Continuous capture path:
                 ftp://mgcusr:mgcusr@10.124.19.169//home/mgcusr/xij/repo.pcap

               Buffer Details: Buffer Type: CIRCULAR No of files: 5 File Size
 (in MB): 21
  Limit Details:
   Number of Packets to capture: 0 (no limit)
   Packet Capture duration: 3600
   Packet Size to capture: 0 (no limit)
   Maximum number of packets to capture per second: 1000
   Packet sampling rate: 0 (no sampling)
```

To view the configured Embedded Packet Capture buffer files, run these commands:

```
Device# show monitor capture epc_session1 buffer brief
 ----------------------------------------------------------------------------

 #   size   timestamp     source             destination      dscp    protocol

 ----------------------------------------------------------------------------

   0 1386    0.000000   192.168.10.117   ->  192.168.10.100   0  BE   UDP
   1 1378    0.000000   192.168.10.100   ->  192.168.10.117   0  BE   UDP
   2 1386    0.001007   192.168.10.117   ->  192.168.10.100   0  BE   UDP
```

Note

This command works for normal embedded packet capture (EPC) but is not supported in continuous captures.

```
Device# show monitor capture epc_session1 buffer dump
0
  0000:  6C8BD3FE AEC0F4BD 9E566E4B 8100000A   l........VnK....
  0010:  08004500 05500000 0000FF11 2073C0A8   ..E..P...... s..
  0020:  0A64C0A8 0A75147F 1480053C 00000010   .d...u.....<....
  0030:  03000000 00000288 0000C48E 8FC860CF   ..............`.
  0040:  DC8C3759 4B203468 95299EA5 00000000   ..7YK 4h.).......
  0050:  AAAA0300 00000800 4500050A 92154000   ........E.....@.
```

```
  0060:  40060BBC C0A80B67 C0A80B65 A7E0139D   @......g...e....
  0070:  32595FD8 0F2D6065 801001F6 EA440000   2Y_..-`e.....D..
  0080:  0101080A BFCB4934 A959414F 36373839   ......I4.YAO6789
  0090:  30313233 34353637 38393031 32333435   0123456789012345
  00A0:  36373839 30313233 34353637 38393031   6789012345678901
  00B0:  32333435 36373839 30313233 34353637   2345678901234567
  00C0:  38393031 32333435 36373839 30313233   8901234567890123
  00D0:  34353637 38393031 32333435 36373839   4567890123456789
  00E0:  30313233 34353637 38393031 32333435   0123456789012345
  00F0:  36373839 30313233 34353637 38393031   6789012345678901
  0100:  32333435 36373839 30313233 34353637   2345678901234567
.
.
.
```

77 Layer 3 Access

Topics:

- Layer 3 access
- OSPF protocols
- PIM sparse mode
- Network address translation (NAT)

This chapter explains configuring Layer 3 wireless controller deployments on Cisco Catalyst 9800 series, covering L3 access setup, OSPF and PIM multicast integration, DHCP/NAT with VRF, and verification commands.

Layer 3 access

Explains Layer 3 access and how it enables routing and advanced networking features on Cisco Catalyst 9800 series wireless controllers.

A Layer 3 access is a wireless controller deployment mode that

- terminates wireless client subnets directly on the controller
- supports Layer 3 routing functions such as OSPFv2 and PIM-SM, and
- enables client subnet segmentation, overlapping IP support, and a scalable network design.

Starting from Cisco IOS XE 17.13.1, the Cisco Catalyst 9800 series wireless controller platforms can be deployed as a Layer 3 (L3) network to perform routing functions.

In Cisco IOS XE 17.12.x and earlier releases, these platforms are deployed as Layer 2 network elements. In these deployments, wireless client subnets are terminated at an upstream network element. Upstream refers to the direction in which data is transferred from clients to a server. The controller forwards traffic based on the MAC address of the clients.

The L3 access feature terminates wireless client subnets in the controller and supports Layer 3 forwarding for wireless client traffic. When L3 is enabled on a specific SSID, the client VLAN of that SSID is terminated at the controller. In this scenario, the wireless controller forwards traffic based on the network layer (IP) address.

This enables:

- Segmentation and client overlapping IP address support using VRF.
- Flexible network design and faster convergence.
- Consistency in network design.
- Scalability for upstream switches and routers.

The core focus is seamless integration of OSPF and multicast routing. This transition empowers wireless networks to respond dynamically to changing business requirements, ensuring optimal performance and agility in dynamic networking environments.

Restrictions for Layer 3 access

This topic outlines the restrictions for Layer 3 access WLANs, including unsupported configurations and features you must avoid to ensure correct operation.

- By default, the L3 access is disabled on a WLAN.
- Only N+1 redundancy is supported with L3 access.
- You cannot configure multiple IP addresses in an SVI.
- High Availability SSO is not supported in L3 WLANs.
- In mixed mode (L2 and L3 WLANs), HA SSO with Loopback as WMI is not supported.
- The **ip radius source-interface vrf** global command is not supported.

- These NAT CLIs are not supported in Cisco IOS XE 17.13.1:

```
show ip nat aggregation
```

```
show ip nat bpa
```

```
show ip nat ha
```

```
show ip nat limits
```

```
show ip nat map
```

```
show ip nat platform
```

```
show ip nat pool
```

```
show ip nat portblock
```

```
show ip nat redundancy
```

```
show ip nat route-dia
```

```
show ip nat translations
```

```
clear ip nat translations
```

- Multicast stream is not supported with VRF.
- Client IPv6 is not supported. The controller or AP cannot have IPv6 addresses.

Use cases for Layer 3 access

This topic outlines the best practices for deploying Layer 3 access, including segmentation, NAT, and support for optimized network designs.

Layer 3 access support

- Segmentation and client overlapping IP address support.
- Flexible and optimized network design using L3 access.

Network Address Translation (NAT) support

- Translating client traffic in the guest network to reach the corporate services (For instance, Cisco ISE).
- Hiding the private IP addresses of clients from outside networks.

> **Note**
>
> Only NAT with IPv4 to IPv4 translation is supported in Cisco IOS XE 17.13.1.

Enable Layer 3 access on policy profile (GUI)

Use this task to enable Layer 3 access on a policy profile so that client traffic on a WLAN can benefit from Layer 3 forwarding.

Enable Layer 3 access on a policy profile to allow client devices connected to the WLAN to utilize Layer 3 forwarding using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Select a policy profile and in the **Edit Policy Profile** window, go to the advanced policy profile properties.
3. Under the **Advanced** tab, enable **L3 Access** on the policy profile so that client traffic on a WLAN that has this policy can benefit from Layer 3 forwarding.
4. Click **Apply to Device**.

Enable Layer 3 access on policy profile (CLI)

Use this task to enable Layer 3 (L3) access for a wireless policy profile using the commands.

Configure a wireless policy profile so that wireless clients can obtain IP addresses and network access through Layer 3 using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a wireless policy profile.

   Example

   ```
   Device(config)# wireless profile policy profile-policy
   ```

3. Disable the wireless policy profile.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

4. Enable L3 access in the wireless policy profile.

Example

```
Device(config-wireless-policy)# l3-access
```

5. Enable the wireless policy profile.

Example

```
Device(config-wireless-policy)# no shutdown
```

Configure a client gateway (GUI)

Use this procedure to configure a client gateway (SVI interface) and associate it with a VRF using the GUI.

Configure a client gateway by associating an SVI interface with a VRF and enabling Autostate Disable to keep the SVI up regardless of connected port status using the GUI.

Procedure

1. Choose **Configuration** > **Layer2** > **VLAN** and select the **SVI** tab.
2. Click an SVI interface. On the **General** tab of the **Edit SVI** window, select a VRF from the drop-down list to associate it with the SVI interface.
3. Enable the **Autostate Disable** to keep the SVI up, even when all ports on the VLAN are down.
4. Click **Save & Apply to Device**.

Configure a client gateway (CLI)

Use this procedure to configure a client gateway interface with a multiprotocol VRF and static IP address using commands.

Establish a client gateway interface, assign it to a VRF, and configure its operational parameters using the commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Specify an interface and enter the interface configuration mode.

Example

```
Device(config)# interface type number
```

3. Activate multiprotocol VRF in an interface.

Example

```
Device(config-if)# vrf forwarding vrf-name
```

4. Define the IP address for the VRF.

Example

```
Device(config-if)# ip address ip-address mask-address
```

5. Configure SVI to ensure that SVI is up even if the VLAN is not switched out.

Example

```
Device(config-if)# no autostate
```

6. Exit the interface configuration mode and enter the global configuration mode.

Example

```
Device(config-if)# end
```

Enable internal DHCP with VRF (CLI)

Use this task to enable the internal DHCP server with Virtual Routing and Forwarding (VRF) on your wireless network device using commands.

Enable the internal DHCP server on a WLAN with VRF support by configuring the policy profile and related settings using the device commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

Example

```
Device(config)# wireless profile policy profile-policy
```

3. Add a description for the policy profile and configure AAA policy override.

Example

```
Device(config-wireless-policy)# description profile-policy-description
```

```
Device(config-wireless-policy)# aaa-override
```

4. Enable DHCP Option 82 for the wireless clients and enable VRF on DHCP Option 82.

   Example

   ```
   Device(config-wireless-policy)# ipv4 dhcp opt82
   ```

   ```
   Device(config-wireless-policy)# ipv4 dhcp opt82 vrf
   ```

5. Configure the WLAN's IPv4 DHCP server IP address and VRF name.

   Example

   ```
   Device(config-wireless-policy)# ipv4 dhcp server ip-address vrf vrf-name
   ```

6. Disable the wireless policy profile and enable L3 access in the wireless policy profile.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

   ```
   Device(config-wireless-policy)# l3-access
   ```

7. Configure Network Access Control in the policy profile.

   Example

   ```
   Device(config-wireless-policy)# nac
   ```

8. Map the VLAN to a policy profile.

   Example

   ```
   Device(config-wireless-policy)# vlan vlan-id
   ```

   If vlan-id is not specified, the default native vlan 1 is applied. The valid range for *vlan-id* is 1 to 4096.

9. Enable the wireless policy profile.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

Verify internal DHCP with VRF details

This section lists commands and sample outputs for verifying internal DHCP and VRF configuration, NAT datapath, and device IP tracking on Cisco devices.

To verify the internal DHCP details, use this command:

```
Device# show run int Vlan55
Building configuration...
Current configuration : 290 bytes
!
interface Vlan55
vrf forwarding sample_guest
ip address 55.55.55.2 255.255.255.0
no ip proxy-arp
ip nat inside
ip cef accounting non-recursive external
ip ospf authentication message-digest
ip ospf message-digest-key 1 md5 cisco123
no autostate
no mop enabled
no mop sysid
end
```

To verify the NAT datapath statistics, use this command:

```
Device# show run int Loopback1
Building configuration...
Current configuration : 90 bytes
!
interface Loopback1
vrf forwarding sample_guest
ip address 7.7.7.1 255.255.255.0
end
ip dhcp pool l3_sample_guest
vrf sample_guest
network 55.55.55.0 255.255.255.0
default-router 55.55.55.2
```

To verify the IP entries from database, use this command:

```
Device# show wireless device-tracking database ip
IP                                          ZONE/VRF-TABLE-ID     STATE
 DISCOVERY     MAC                VRF-NAME
55.55.55.2                                  0x00000003          Reachable
   Local      001e.bd11.a0ff

               55.55.55.6                                     0x00000003
   Reachable      IPv4 DHCP   58a0.239b.d25f     sample_guest
```

Verify Layer 3 access details

This section lists commands and their output fields for verifying Layer 3 access status on policy profiles and clients.

To verify whether Layer 3 access is enabled for a specific policy profile, use this command:

```
Device# show wireless profile policy detailed default-policy-profile
Policy Profile Name                : default-policy-profile
Description                        : default policy profile
Status                             : ENABLED
VLAN                               : 20
.
.
.

                L3 Forwarding
     :ENABLED
```

To view whether the Layer 3 access is enabled under policy profile, use this command:

```
Device# show wireless profile policy all
Policy Profile Name                : default-policy-profile
Description                        : default policy profile
Status                             : ENABLED
VLAN                               : 20
.
.
.

                L3 Forwarding
     :ENABLED
```

To verify the client information, use this command:

```
Device# show wireless client mac-address <mac-address> detail
Client MAC Address : a886.ddb2.05e9
.
.
.

L3 Forwarding: Enabled
```

To verify the client gateway details, use this command:

```
Device# show wireless client mac-address 0024.d742.46e4 detail | inc Gateway
.
.
.

                Client Gateway IPv4 Address : 117.117.117.1
```

> **Note**
>
> The client gateway is displayed only if the client performs DHCP.
>
> If the client learns IP using static or ARP, the client gateway will not be displayed.

OSPF protocols

Explains OSPF protocols, their key functions as link-state routing protocols, core authentication methods, notable features and enhancements in OSPF deployments.

An OSPF protocol is a link-state routing protocol that

- calculates the best path through a network using the shortest path first algorithm
- enables dynamic route selection for IP networks, and
- supports multiple authentication methods for secure device communication.

OSPF is a widely used Interior Gateway Protocol (IGP). Each device can verify the identity of the other devices it communicates with.

OSPF supports several types of authentication:

- Simple password authentication: Each device uses a clear-text password configured for authentication with other devices. However, this method is not secure as the password appears in configuration files and OSPF messages. This is not a secure way to configure devices.
- MD5 authentication: This method uses the MD5 algorithm to compute a hash value from the OSPF packet contents and a password.

> **Note**
>
> Starting with Cisco IOS XE 17.13.1, OSPFv2 and ECMP are supported.

Configure OSPF

Use this task to enable and configure OSPF routing on your device interfaces.

Set up Open Shortest Path First (OSPF) to enable dynamic routing across device interfaces.

Procedure

1. Configure a clear-text password (or) message digest key in an OSPF-enabled interface.
2. Create an OSPF routing process.
3. Specify the range of IP addresses to associate with the routing process.
4. Assign area IDs to be associated with that range.

What to do next

What to do next

> **Note**
>
> To enable OSPF in SVI interfaces, you must enable Multicast over Multicast (MOM) using the **wireless multicast** *ip-address* command. This allows OSPF to establish neighbor adjacencies between SVIs.

These topics describe procedures to configure routing protocol:

Configure OSPF interfaces (GUI)

Use this task to configure OSPF interface settings on a device using the GUI.

Enable and customize OSPF parameters on specific interfaces through the GUI.

Procedure

1. Choose **Configuration** > **Interface** > **Ethernet** and select an interface to configure it with OSPF settings.
2. In the **Configure Interface** window, ensure that you have configured an IP address, subnet mask and optionally a secondary IP address.
3. In the **OSPF** section, enter the **Process ID** to enable OSPF on the interface.
4. Enable the **BFD** to create a Bidirectional Forwarding Detection session between two systems. BFD provides a short-duration method of detecting failures in the forwarding path between two adjacent peers.
5. Select the **Dead Interval Minimal** and enter the number of seconds in the **Hello Multiplier** field to set the interval at which at least one hello packet must be received, or else the neighbor is considered down.
6. Select **Message Digest Authentication** to configure the authentication supported by OSPF.
7. Under the **Message Digest Authentication- Key Map** association box, enter the Key, Type and Password.
8. Click **Save & Apply to Device**.

> **Note**
>
> To configure OSPF in SVI interfaces, you must enable Multicast over Multicast (MOM). This allows OSPF to establish neighbor adjacencies between SVIs.

Configure OSPF protocol (GUI)

Use this task to configure OSPF protocol settings on a router through the GUI.

Configure OSPF protocol for dynamic routing between network devices using the GUI.

Procedure

1. Choose **Configuration** > **Routing Protocol** > **OSPF** and click **Add**.

2. In the **Add Route** page, select the router from the drop-down list.
3. Enter the **Process ID**. It identifies the router's OSPF routing process to other routers.
4. Enter a **Router ID**.
5. Enable the **BFD** to create a Bidirectional Forwarding Detection session between two systems. BFD provides a short-duration method for detecting failures in the forwarding path between two adjacent switches. This includes interfaces, data links, and forwarding planes. OSPF registers as a protocol with BFD and receives forwarding path detection failure messages from BFD. You can configure BFD support for OSPF globally on all interfaces, or selectively on specific interfaces. BFD timers are negotiated. The BFD peers begin sending BFD control packets to each other at the negotiated interval.
6. Enable **NSR** on a router with redundant Route Processors (RPs) so that it maintains its Open Shortest Path First (OSPF) state and adjacencies during both planned and unplanned RP switchovers. The active RP transfers OSPF state information to the standby RP by checkpointing the data. After a switchover to the standby RP, OSPF uses the checkpointed information to continue operation without interruption.

   Optionally, you can check the corresponding check box to enable VRF and select the VRF Name. If you have not configured the VRF, use the link to configure it on the **Interface** > **VRF** page.
7. For advanced options, check the **Advanced** radio button and populate these fields:

   IP Address: Enter the address of the destination network for this route.

   Wildcard: Enter the subnet mask used on that network.

   Area: The OSPF area number for that network. Each router in a particular OSPF area maintains a topological database for that area.
8. Click **Save & Apply to Device**.

Configure basic OSPF parameters (CLI)

Learn how to configure basic OSPF parameters using commands to enable dynamic routing in your network device.

Enable OSPF routing and define key parameters for efficient dynamic route advertisement and network segmentation using commands.

Procedure

---

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Enable OSPF routing.

   Example

   ```
   Device(config)# router ospf process-id
   ```

   The *process-id* is an internally used identification parameter that is locally assigned and can be any positive integer. Each OSPF routing process has a unique value.

> **Note**
>
> The OSPF for Routed Access supports a maximum of 1000 dynamically learned routes.

3. Define a network on which the OSPF runs an area ID for that interface.

   Example

   ```
   Device(config-router)# network address wildcard-mask area area-id
   ```

   You can use the *wildcard-mask* to define one or more interfaces to be associated with a specific OSPF area. The *area-id* can be a decimal value or an IP address.

4. Enable Bidirectional Forwarding Detection (BFD) in all interfaces.

   Example

   ```
   Device(config-router)# bfd all-interfaces
   ```

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config-router)# end
   ```

Configure OSPF interfaces (CLI)

Use this procedure to configure OSPF interfaces using commands for device interfaces that require dynamic routing with OSPF.

Configure router interfaces to participate in OSPF routing, enabling dynamic route exchange within OSPF areas on the network.

Procedure

---

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify interface to configure OSPF interfaces.

   Example

   ```
   Device(config)# interface GigabitEthernet interface-number
   ```

3. Configure IP address for the OSPF interface.

Example

```
Device(config-if)# ip address ip-address mask-address
```

4. Enable message digest for a specific interface.

Example

```
Device(config-if)# ip ospf authentication message-digest
```

5. Enable message digest key for the OSPF.

Example

```
Device(config-if)# ip ospf authentication message-digest-key key-number md5
password
```

6. Assign interface and its network to OSPF process and area.

Example

```
Device(config-if)# ip ospf value area area-id
```

7. Enable BFD in an interface.

Example

```
Device(config-if)# ip ospf bfd
```

8. Return to privileged EXEC mode.

Example

```
Device(config-if)# end
```

Verify routing protocol details

This section lists common commands to verify routing protocol status and behavior. Each entry describes the command's purpose and summarizes key fields from its output to assist with prompt network troubleshooting.

To verify the OSPF details, use this command:

```
Device# show ip ospf 1
Routing Process "ospf 1" with ID 31.31.31.1
 Start time: 00:01:46.103, Time elapsed: 03:12:34.745
 Supports only single TOS(TOS0) routes
 Supports opaque LSA
 Supports Link-local Signaling (LLS)
 Supports area transit capability
 Supports NSSA (compatible with RFC 3101)
 Supports Database Exchange Summary List Optimization (RFC 5243)
 Event-log enabled, Maximum number of events: 1000, Mode: cyclic
 Router is not originating router-LSAs with maximum metric
```

```
 Initial SPF schedule delay 50 msecs
 Minimum hold time between two consecutive SPFs 200 msecs
 Maximum wait time between two consecutive SPFs 5000 msecs
 Incremental-SPF disabled
 Initial LSA throttle delay 50 msecs
 Minimum hold time for LSA throttle 200 msecs
 Maximum wait time for LSA throttle 5000 msecs
 Minimum LSA arrival 100 msecs
 LSA group pacing timer 240 secs
 Interface flood pacing timer 33 msecs
 Retransmission pacing timer 66 msecs
 EXCHANGE/LOADING adjacency limit: initial 300, process maximum 300
 Number of external LSA 0. Checksum Sum 0x000000
 Number of opaque AS LSA 0. Checksum Sum 0x000000
 Number of DCbitless external and opaque AS LSA 0
 Number of DoNotAge external and opaque AS LSA 0
 Number of areas in this router is 1. 1 normal 0 stub 0 nssa
 Number of areas transit capable is 0
 External flood list length 0
 IETF NSF helper support enabled
 Cisco NSF helper support enabled
 Reference bandwidth unit is 100 mbps
    Area 1
        Number of interfaces in this area is 3
 Area has no authentication
 SPF algorithm last executed 03:11:47.277 ago
 SPF algorithm executed 9 times
 Area ranges are
 Number of LSA 5. Checksum Sum 0x0212EE
 Number of opaque link LSA 0. Checksum Sum 0x000000
 Number of DCbitless LSA 0
 Number of indication LSA 0
 Number of DoNotAge LSA 0
 Flood list length 0
```

To verify the OSPF database details, use this command:

```
Device# show ip ospf 1 database
OSPF Router with ID (31.31.31.1) (Process ID 1)
  Router Link States (Area 1)
Link ID         ADV Router      Age         Seq#       Checksum Link count
31.31.31.1      31.31.31.1      1470        0x8000000C 0x00289A 3
50.50.50.1      50.50.50.1      1745        0x8000000A 0x001018 3
51.51.51.1      51.51.51.1      1500        0x8000000A 0x008EFB 2
  Net Link States (Area 1)
Link ID         ADV Router      Age         Seq#       Checksum
30.30.30.2      50.50.50.1      1745        0x80000006 0x00B793
31.31.31.2      51.51.51.1      1500        0x80000006 0x0093AE
```

To verify the IP route details, use this command:

```
Device# show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2, m - OMP
       n - NAT, Ni - NAT inside, No - NAT outside, Nd - NAT DIA
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static route

       H - NHRP, G - NHRP registered, g - NHRP registration summary
       o - ODR, P - periodic downloaded static route, l - LISP
```

```
       a - application route
       + - replicated route, % - next hop override, p - overrides from PfR
       & - replicated local route overrides by connected
Gateway of last resort is not set
      5.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        5.5.5.0/24 is directly connected, Vlan5
L        5.5.5.2/32 is directly connected, Vlan5
      6.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        6.6.6.0/24 is directly connected, Vlan6
L        6.6.6.2/32 is directly connected, Vlan6
      30.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        30.30.30.0/24 is directly connected, GigabitEthernet3
L        30.30.30.1/32 is directly connected, GigabitEthernet3
      31.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        31.31.31.0/24 is directly connected, GigabitEthernet4
L        31.31.31.1/32 is directly connected, GigabitEthernet4
      32.0.0.0/24 is subnetted, 1 subnets
O        32.32.32.0 [110/2] via 30.30.30.2, 03:11:58, GigabitEthernet3
      50.0.0.0/32 is subnetted, 1 subnets
O        50.50.50.1 [110/2] via 30.30.30.2, 03:11:58, GigabitEthernet3
      51.0.0.0/32 is subnetted, 1 subnets
O        51.51.51.1 [110/2] via 31.31.31.2, 03:12:00, GigabitEthernet4
```

To verify the IP OSPF route list details, use this command:

```
Device# show ip ospf 1 route-list
OSPF Router with ID (31.31.31.1) (Process ID 1)
  Base Topology (MTID 0)
    Area 1
    Intra-area Route List
*   31.31.31.0/24, Intra, cost 1, area 1, Connected
      via 31.31.31.1, GigabitEthernet4
*   30.30.30.0/24, Intra, cost 1, area 1, Connected
      via 30.30.30.1, GigabitEthernet3
*   6.6.6.0/24, Intra, cost 1, area 1, Connected
      via 6.6.6.2, Vlan6
*>  32.32.32.0/24, Intra, cost 2, area 1
      via 30.30.30.2, GigabitEthernet3
*>  50.50.50.1/32, Intra, cost 2, area 1
      via 30.30.30.2, GigabitEthernet3
*>  51.51.51.1/32, Intra, cost 2, area 1
      via 31.31.31.2, GigabitEthernet4
    First Hop Forwarding Gateway Tree
 31.31.31.1 on GigabitEthernet4, count 1
 31.31.31.2 on GigabitEthernet4, count 1
 30.30.30.1 on GigabitEthernet3, count 1
 30.30.30.2 on GigabitEthernet3, count 2
 6.6.6.2 on Vlan6, count 1
```

To verify the OSPF traffic details, use this command:

```
Device# show ip ospf 1 traffic
OSPF Router with ID (31.31.31.1) (Process ID 1)
OSPF queue statistics for process ID 1:
                  InputQ     UpdateQ    OutputQ
  Limit           0          200        0
  Drops           0          0          0
  Max delay [msec] 1         1          1
  Max size        2          2          2
    Invalid       0          0          0
    Hello         0          0          0
    DB des        0          0          1
```

```
    LS req         1          1          1
    LS upd         1          1          0
    LS ack         0          0          0
  Current size     0          0          0
    Invalid        0          0          0
    Hello          0          0          0
    DB des         0          0          0
    LS req         0          0          0
    LS upd         0          0          0
    LS ack         0          0          0
Interface statistics:
.
.
.
    Interface GigabitEthernet4
Summary traffic statistics for process ID 1:
OSPF packets received/sent
  Type          Packets              Bytes
  RX Invalid    0                    0
  RX Hello      2435                 116880
  RX DB des     17                   584
  RX LS req     2                    96
  RX LS upd     24                   2360
  RX LS ack     24                   1436
  RX Total      2502                 121356
  TX Failed     0                    0
  TX Hello      3653                 506540
  TX DB des     6                    704
  TX LS req     2                    144
  TX LS upd     31                   4204
  TX LS ack     14                   1560
  TX Total      3706                 513152
OSPF header errors
  Length 0, Instance ID 0, Checksum 0, Auth Type 0,
  Version 0, Bad Source 0, No Virtual Link 0,
  Area Mismatch 0, No Sham Link 0, Self Originated 0,
  Duplicate ID 0, Hello 0, MTU Mismatch 0,
  Nbr Ignored 0, LLS 0, Unknown Neighbor 0,
  Authentication 0, TTL Check Fail 0, Adjacency Throttle 0,
  BFD 0, Test discard 0
OSPF LSA errors
  Type 0, Length 0, Data 0, Checksum 0
```

To verify the OSPF neighbor details, use this command:

```
Device# show ip ospf 1 neighbor
Neighbor ID     Pri   State           Dead Time   Address         Interface
51.51.51.1        1   FULL/DR         00:00:37    31.31.31.2
GigabitEthernet4
50.50.50.1        1   FULL/DR         00:00:39    30.30.30.2
GigabitEthernet3
```

To verify the OSPF neighbor summary, use this command:

```
Device#show ip ospf 1 neighbor summary
            OSPF Router with ID (31.31.31.1) (Process ID 1)
DOWN         0
ATTEMPT      0
INIT         0
2WAY         0
EXSTART      0
EXCHANGE     0
```

```
LOADING      0
FULL         2
Total count  2    (Undergoing NSF 0)
```

To verify the OSPF event details, use this command:

```
               Device# show ip ospf 1 events
            OSPF Router with ID (31.31.31.1) (Process ID 1)
1    Sep 21 21:49:12.406: Generate Changed Type-1 LSA, LSID 31.31.31.1, Seq#
8000000C, Age 0, Area 1
2    Sep 21 21:48:44.064: Rcv Unchanged Type-2 LSA, LSID 31.31.31.2, Adv-Rtr
51.51.51.1, Seq# 80000006, Age 1, Area 1
3    Sep 21 21:48:44.064: Rcv Unchanged Type-1 LSA, LSID 51.51.51.1, Adv-Rtr
51.51.51.1, Seq# 8000000A, Age 1, Area 1
4    Sep 21 21:44:38.726: Rcv Unchanged Type-2 LSA, LSID 30.30.30.2, Adv-Rtr
50.50.50.1, Seq# 80000006, Age 1, Area 1
5    Sep 21 21:44:38.726: Rcv Unchanged Type-1 LSA, LSID 50.50.50.1, Adv-Rtr
50.50.50.1, Seq# 8000000A, Age 1, Area 1
.
.
.
30   Sep 21 19:01:45.594: End of SPF, Topo Base, SPF time 1ms, next
wait-interval 800ms
.
.
.
74

               Sep 21 19:01:44.676: Generic:  ospf_external_route_sync  0x1

               75   Sep 21 19:01:44.676: Generic:  ospf_external_route_sync
 0x1
76   Sep 21 19:01:44.676: Generic:  ospf_external_route_sync  0x0
77   Sep 21 19:01:44.676: Generic:  ospf_external_route_sync  0x0
78   Sep 21 19:01:44.676: Starting External processing, Topo Base in area 1
79   Sep 21 19:01:44.676: Starting External processing, Topo Base
80

               Sep 21 19:01:44.676: Generic:  ospf_inter_route_sync  0x1

81   Sep 21 19:01:44.676: Generic:  ospf_inter_route_sync  0x1
82   Sep 21 19:01:44.676: Starting summary processing, Topo Base, Area 1
83

               Sep 21 19:01:44.676: Generic:  post_spf_intra  0x0
84   Sep 21 19:01:44.676: Generic:  ospf_intra_route_sync  0x1

.
.
.
```

To verify the OSPF details in the database summary, use this command:

```
Device# show ip ospf 1 database database-summary
OSPF Router with ID (31.31.31.1) (Process ID 1)
Area 1 database summary
  LSA Type      Count    Delete   Maxage
  Router        3        0        0
  Network       2        0        0
  Summary Net   0        0        0
```

```
  Summary ASBR  0         0         0
  Type-7 Ext    0         0         0
    Prefixes redistributed in Type-7  0
  Opaque Link   0         0         0
  Opaque Area   0         0         0
  Subtotal      5         0         0
Process 1 database summary
  LSA Type      Count    Delete   Maxage
  Router        3        0        0
  Network       2        0        0
  Summary Net   0        0        0
  Summary ASBR  0        0        0
  Type-7 Ext    0        0        0
  Opaque Link   0        0        0
  Opaque Area   0        0        0
  Type-5 Ext    0        0        0
      Prefixes redistributed in Type-5  0
  Opaque AS     0        0        0
  Total         5        0        0
  Non-self      4
```

To verify the OSPF details in the internal database, use this command:

```
Device# show ip ospf 1 database internal
OSPF Router with ID (31.31.31.1) (Process ID 1)
  Stub Link States (Area 1)
Link ID         ADV Router      Age         Seq#       Checksum Mask
6.6.6.255       31.31.31.1      11545       0x0        0x006611 /24
30.30.30.255    31.31.31.1      11546       0x0        0x00032C /24
31.31.31.255    31.31.31.1      11548       0x0        0x00DE4D /24
32.32.32.255    50.50.50.1      11545       0x0        0x00F0FE /24
50.50.50.1      50.50.50.1      11545       0x0        0x005C5C /32
51.51.51.1      51.51.51.1      11547       0x0        0x002092 /32
  Router Link States (Area 1)
Link ID         ADV Router      Age         Seq#       Checksum Link count
31.31.31.1      31.31.31.1      1498        0x8000000C 0x00289A 3
50.50.50.1      50.50.50.1      1772        0x8000000A 0x001018 3
51.51.51.1      51.51.51.1      1527        0x8000000A 0x008EFB 2
  Net Link States (Area 1)
Link ID         ADV Router      Age         Seq#       Checksum
30.30.30.2      50.50.50.1      1772        0x80000006 0x00B793
31.31.31.2      51.51.51.1      1527        0x80000006 0x0093AE
```

To verify the OSPF details in the database network, use this command:

```
Device# show ip ospf 1 database network
OSPF Router with ID (31.31.31.1) (Process ID 1)
  Net Link States (Area 1)
  LS age: 1772
  Options: (No TOS-capability, DC)
  LS Type: Network Links
  Link State ID: 30.30.30.2 (address of Designated Router)
  Advertising Router: 50.50.50.1
  LS Seq Number: 80000006
  Checksum: 0xB793
  Length: 32
  Network Mask: /24
 Attached Router: 50.50.50.1
 Attached Router: 31.31.31.1
  LS age: 1527
  Options: (No TOS-capability, DC)
```

```
  LS Type: Network Links
  Link State ID: 31.31.31.2 (address of Designated Router)
  Advertising Router: 51.51.51.1
  LS Seq Number: 80000006
  Checksum: 0x93AE
  Length: 32
  Network Mask: /24
 Attached Router: 51.51.51.1
 Attached Router: 31.31.31.1
```

To verify the OSPF details in the database router, use this command:

```
Device# show ip ospf 1 database router
OSPF Router with ID (31.31.31.1) (Process ID 1)
  Router Link States (Area 1)
  LS age: 1498
  Options: (No TOS-capability, DC)
  LS Type: Router Links
  Link State ID: 31.31.31.1
  Advertising Router: 31.31.31.1
  LS Seq Number: 8000000C
  Checksum: 0x289A
  Length: 60
  Number of Links: 3
    Link connected to: a Transit Network
     (Link ID) Designated Router address: 31.31.31.2
     (Link Data) Router Interface address: 31.31.31.1
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
    Link connected to: a Transit Network
     (Link ID) Designated Router address: 30.30.30.2
     (Link Data) Router Interface address: 30.30.30.1
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
    Link connected to: a Stub Network
     (Link ID) Network/subnet number: 6.6.6.0
     (Link Data) Network Mask: 255.255.255.0
      Number of MTID metrics: 0
       TOS 0 Metrics: 1
.
.
.
```

To verify the OSPF details in the database topology, use this command:

```
Device# show ip ospf 1 database topology
OSPF Router with ID (31.31.31.1) (Process ID 1)
  Base Topology (MTID 0)
  Router Link States (Area 1)
Link ID         ADV Router      Age         Seq#       Checksum Link count
31.31.31.1      31.31.31.1      1498        0x8000000C 0x00289A 3
50.50.50.1      50.50.50.1      1772        0x8000000A 0x001018 3
51.51.51.1      51.51.51.1      1527        0x8000000A 0x008EFB 2
  Net Link States (Area 1)
Link ID         ADV Router      Age         Seq#       Checksum
30.30.30.2      50.50.50.1      1772        0x80000006 0x00B793
31.31.31.2      51.51.51.1      1527        0x80000006 0x0093AE
vWLC_TB1#
vWLC_TB1#show ip ospf 1 request-list
            OSPF Router with ID (31.31.31.1) (Process ID 1)
 Neighbor 51.51.51.1, interface GigabitEthernet4 address 31.31.31.2
 Request list size 0, maximum list size 1
```

```
 Neighbor 50.50.50.1, interface GigabitEthernet3 address 30.30.30.2
 Request list size 0, maximum list size 1
vWLC_TB1#
vWLC_TB1#show ip ospf flood-list
            OSPF Router with ID (31.31.31.1) (Process ID 1)
 Interface GigabitEthernet4, Queue length 0
 Interface GigabitEthernet3, Queue length 0
 Interface Vlan6, Queue length 0
```

To verify the OSPF request details, use this command:

```
Device# show ip ospf request-list Gi3 50.50.50.1
OSPF Router with ID (31.31.31.1) (Process ID 1)
 Neighbor 50.50.50.1, interface GigabitEthernet3 address 30.30.30.2
 Request list size 0, maximum list size 1
```

To verify the OSPF interface details, use this command:

```
Device# show ip ospf interface
GigabitEthernet4 is up, line protocol is up
  Internet Address 31.31.31.1/24, Interface ID 10, Area 1
  Attached via Network Statement
  Process ID 1, Router ID 31.31.31.1, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 51.51.51.1, Interface address 31.31.31.2
  Backup Designated router (ID) 31.31.31.1, Interface address 31.31.31.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:03
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Not Protected by per-prefix TI-LFA
  Index 1/3/3, flood queue length 0
  Next 0x0(0)/0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 2
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 51.51.51.1  (Designated Router)
  Suppress hello for 0 neighbor(s)
  Cryptographic authentication enabled
    Youngest key id is 1
GigabitEthernet3 is up, line protocol is up
  Internet Address 30.30.30.1/24, Interface ID 9, Area 1
  Attached via Network Statement
  Process ID 1, Router ID 31.31.31.1, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Transmit Delay is 1 sec, State BDR, Priority 1
  Designated Router (ID) 50.50.50.1, Interface address 30.30.30.2
  Backup Designated router (ID) 31.31.31.1, Interface address 30.30.30.1
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:06
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
```

```
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Not Protected by per-prefix TI-LFA
  Index 1/2/2, flood queue length 0
  Next 0x0(0)/0x0(0)/0x0(0)
  Last flood scan length is 1, maximum is 2
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 1, Adjacent neighbor count is 1
    Adjacent with neighbor 50.50.50.1  (Designated Router)
  Suppress hello for 0 neighbor(s)
  Cryptographic authentication enabled
    Youngest key id is 1
Vlan6 is up, line protocol is up
  Internet Address 6.6.6.2/24, Interface ID 16, Area 1
  Attached via Interface Enable
  Process ID 1, Router ID 31.31.31.1, Network Type BROADCAST, Cost: 1
  Topology-MTID    Cost    Disabled    Shutdown      Topology Name
        0           1         no          no            Base
  Enabled by interface config, including secondary ip addresses
  Transmit Delay is 1 sec, State DR, Priority 1
  Designated Router (ID) 31.31.31.1, Interface address 6.6.6.2
  No backup designated router on this network
  Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
    oob-resync timeout 40
    Hello due in 00:00:01
  Supports Link-local Signaling (LLS)
  Cisco NSF helper support enabled
  IETF NSF helper support enabled
  Can be protected by per-prefix Loop-Free FastReroute
  Can be used for per-prefix Loop-Free FastReroute repair paths
  Not Protected by per-prefix TI-LFA
  Index 1/1/1, flood queue length 0
  Next 0x0(0)/0x0(0)/0x0(0)
  Last flood scan length is 0, maximum is 0
  Last flood scan time is 0 msec, maximum is 0 msec
  Neighbor Count is 0, Adjacent neighbor count is 0
  Suppress hello for 0 neighbor(s)
  Cryptographic authentication enabled
    Youngest key id is 1
```

PIM sparse mode

Explains how PIM sparse modes operate as multicast routing protocols, their reliance on explicit group membership, and the role of the Rendezvous Point (RP) in managing group traffic.

A PIM sparse mode is a multicast routing protocol that

- assumes recipients for multicast groups are sparsely distributed throughout the network
- builds multicast distribution trees rooted at a selected Rendezvous Point (RP) node by default, and
- requires routers to explicitly signal interest in specific multicast groups or sources.

Protocol Independent Multicast (PIM) includes multicast routing protocols for various environments.

For information about PIM-SM, see https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipmulti_pim/configuration/15-sy/imc-pim-15-sy-book/ip6-mcast-pim-sm.html.

PIM-SM

Most network subnets do not require every multicast packet. Routers explicitly inform upstream neighbors of their interest in specific multicast groups and sources to receive multicast data.

A key requirement of PIM-SM is discovering the RP address for a multicast group through a shared tree.

Enable multicast traffic without VRF (GUI)

Use this task to enable multicast traffic on your device without configuring a VRF, using the GUI.

Enable multicast routing and required protocols to support multicast traffic delivery in VLANs without VRF using the GUI.

Procedure

1. Choose **Configuration** > **Services** > **Multicast**.
2. In the **PIM and Multicast Routing** section, configure multicast routing globally by enabling **Distributed Multicast-Routing**.
3. Configure PIM RP-Address in the **PIM Configuration** sub-section. This configuration is required so that receivers can find the multicast source in the network. Choose the configuration options from these options:
   - Enter the address to statically configure the RP Address.
   - Enable Auto RP Listener to dynamically discover RP in a PIM-SM network.
4. Click **Save & Apply to Device**.
5. Designate the interface on which multicast traffic should be sent. To do so, go to **Configuration** > **Layer 2** > **VLAN** and select the **SVI** interface.
6. Enable the **PIM Sparse Mode** protocol so that the SVI interface can participate in sparse mode multicast traffic transmission and join the multicast shared tree. This ensures that clients in that VLAN can receive multicast traffic from multiple multicast groups or sources.
7. Select the desired IGMP version from the drop-down list to improve multicast packet direction. When this feature is enabled, the controller gathers IGMP reports from the clients and processes them. It creates unique multicast group IDs (MGIDs) from the IGMP reports by selecting the Layer 3 multicast address and the VLAN number, then sends the IGMP reports to the infrastructure switch.
8. Select **IPv4** checkbox and enter the details.
9. Click **Save & Apply to Device**.

Enable multicast traffic without VRF (CLI)

Use this task to enable multicast traffic on a device without a VRF using commands, including IP multicast routing, PIM, OSPF authentication, and necessary interface configurations.

Enable multicast routing capabilities on a device in a flat (non-VRF) environment and ensure proper configuration for multicast traffic propagation using commands.

Procedure

1. Enter the global configuration mode and enable IP multicast routing.

   Example

   ```
   Device# configure terminal
   ```

   ```
   Device(config)# ip multicast-routing distributed
   ```

   The **distributed** keyword enables multicast globally.

2. Enable multicast traffic and configure the address of a PIM Rendezvous Point (RP).

   Example

   ```
   Device(config)# wireless multicast ip-address
   ```

   ```
   Device(config)# ip pim rp-address ip-address
   ```

3. Select an interface connected to hosts on which PIM can be enabled.

   Example

   ```
   Device(config)# interface interface-type-number
   ```

4. Add a description for the VLAN.

   Example

   ```
   Device(config-if)# description description
   ```

5. Enable IP address on an interface and disable proxy ARP.

   Example

   ```
   Device(config-if)# ip address ip-address mask-address
   ```

   ```
   Device(config-if)# no ip proxy-arp
   ```

6. Enable PIM-SM mode and enable OSPF authentication for a specific interface.

   Example

   ```
   Device(config-if)# ip pim sparse-mode
   ```

   ```
   Device(config-if)# ip ospf authentication message-digest
   ```

7. Enable message digest key for the OSPF.

   Example

   ```
   Device(config-if)# ip ospf authentication message-digest-key key-number md5
   password
   ```

8. Disable the maintenance operation protocol (MOP) for an interface.

   Example

   ```
   Device(config-if)# no mop enabled
   ```

9. Disable the task of sending MOP periodic system ID messages and return to the privileged EXEC mode.

Example

```
Device(config-if)# no mop sysid
```

```
Device(config-if)# end
```

Enable multicast traffic with PIM-SSM (CLI)

Use this task to enable multicast traffic on your device using Protocol Independent Multicast-Source Specific Multicast (PIM-SSM) using commands.

Enable efficient multicast traffic routing for applications that require directed streaming to specific receivers using commands.

Procedure

1. Enter the global configuration mode and enable IP multicast routing.

   Example

   ```
   Device# configure terminal
   ```

   ```
   Device(config)# ip multicast-routing distributed
   ```

   The **distributed** keyword enables MDS globally.

2. Enable multicast traffic.

   Example

   ```
   Device(config)# wireless multicast ip address
   ```

   For information about the multicast traffic, see Wireless Multicast.

3. Configure PIM-SSM on all network devices.

   Example

   ```
   Device(config)# ip pim ssm default
   ```

   > **Note**
   >
   > The default SSM range is **232.0.0.0/8**. So, if you do not configure different range, the default SSM range is used.

4. Define SSM range of IP multicast addresses.

Example

```
Device(config)# ip pim ssm range access-list
```

5. Select an interface connected to hosts on which PIM can be enabled.

Example

```
Device(config)# interface interface-type-number
```

6. Add a description for the VLAN.

Example

```
Device(config-if)# description description
```

7. Enable IP address on an interface.

Example

```
Device(config-if)# ip address ip address mask-address
```

8. Disable proxy ARP.

Example

```
Device(config-if)# no ip proxy-arp
```

9. Enable PIM-SM on an interface and return to privileged EXEC mode.

Example

```
Device(config-if)# ip pim sparse-mode
```

```
Device(config-if)# end
```

Verify multicast traffic details

This section lists the main commands and outputs used to verify multicast traffic details, group memberships, IGMP snooping, and troubleshoot multicast issues on Cisco devices.

To verify if a multicast group supports SSM or not, use this command:

```
Device# show ip mroute
IP Multicast Routing Table
Flags: D - Dense, S - Sparse, B - Bidir Group, s - SSM Group, C - Connected,
       L - Local, P - Pruned, R - RP-bit set, F - Register flag,
       T - SPT-bit set, J - Join SPT, M - MSDP created entry, E - Extranet,
       X - Proxy Join Timer Running, A - Candidate for MSDP Advertisement,
       U - URD, I - Received Source Specific Host Report,
       Z - Multicast Tunnel, z - MDT-data group sender,
```

```
       Y - Joined MDT-data group, y - Sending to MDT-data group,
       G - Received BGP C-Mroute, g - Sent BGP C-Mroute,
       N - Received BGP Shared-Tree Prune, n - BGP C-Mroute suppressed,
       Q - Received BGP S-A Route, q - Sent BGP S-A Route,
       V - RD & Vector, v - Vector, p - PIM Joins on route,
       x - VxLAN group, c - PFP-SA cache created entry,
       * - determined by Assert, # - iif-starg configured on rpf intf,
       e - encap-helper tunnel flag, l - LISP decap ref count contributor
Outgoing interface flags: H - Hardware switched, A - Assert winner, p - PIM
Join
                          t - LISP transit group
 Timers: Uptime/Expires
 Interface state: Interface, Next-Hop or VCD, State/Mode
(*, 239.0.0.158), 00:00:07/stopped, RP 15.1.1.2, flags: SJC
  Incoming interface: GigabitEthernet3, RPF nbr 13.1.1.2
  Outgoing interface list:
    Vlan12, Forward/Sparse, 00:00:07/00:02:52, flags:
(17.1.1.1, 239.0.0.158), 00:00:06/00:02:53, flags: JT
  Incoming interface: GigabitEthernet3, RPF nbr 13.1.1.2
  Outgoing interface list:
    Vlan12, Forward/Sparse, 00:00:06/00:02:53, flags:
(*, 231.1.1.1), 02:32:08/stopped, RP 15.1.1.2, flags: SJCF
  Incoming interface: GigabitEthernet3, RPF nbr 13.1.1.2
  Outgoing interface list:
    Vlan12, Forward/Sparse, 00:01:31/00:01:28, flags:
(12.1.0.198, 231.1.1.1), 02:32:08/00:02:53, flags: PFT
  Incoming interface: Vlan12, RPF nbr 0.0.0.0
  Outgoing interface list: Null
(*, 224.0.1.40), 02:32:14/00:02:47, RP 15.1.1.2, flags: SJPL
  Incoming interface: GigabitEthernet3, RPF nbr 13.1.1.2
  Outgoing interface list: Null
```

To verify the IGMP membership details, use this command:

```
Device# show ip igmp membership
Flags: A  - aggregate, T - tracked
       L  - Local, S - static, V - virtual, R - Reported through v3
       I - v3lite, U - Urd, M - SSM (S,G) channel
       1,2,3 - The version of IGMP, the group is in
Channel/Group-Flags:
       / - Filtering entry (Exclude mode (S,G), Include mode (G))
Reporter:
       <mac-or-ip-address> - last reporter if group is not explicitly tracked

       <n>/<m>      - <n> reporter in include mode, <m> reporter in exclude
 Channel/Group                Reporter       Uptime   Exp.  Flags  Interface

 *,239.255.255.250             11.1.1.4        00:01:38 02:57 2A     Vl12
 *,239.0.0.158                 11.1.1.3        00:00:05 02:54 2A     Vl12
 *,231.1.1.1                   12.1.0.8        00:00:07 02:52 2A     Vl12
 *,224.0.1.40                  13.1.1.1        02:34:15 02:45 2LA    Gi3
```

To verify the IGMP snooping details, use this command:

```
Device# show ip igmp snooping igmpv2-tracking
Client to SGV mappings
----------------------
Client: 11.1.1.3 Port: Ca2
    Group: 239.0.0.158 Vlan: 12 Source: 0.0.0.0 blacklisted: no
Client: 11.1.1.4 Port: Ca2
```

```
    Group: 239.255.255.250 Vlan: 12 Source: 0.0.0.0 blacklisted: no
SGV to Client mappings
---------------------
Group: 239.0.0.158 Source: 0.0.0.0 Vlan: 12
    Client: 11.1.1.3 Port: Ca2 Blacklisted: no
Group: 239.255.255.250 Source: 0.0.0.0 Vlan: 12
    Client: 11.1.1.4 Port: Ca2 Blacklisted: no
```

To verify the multicast group summary details, use this command:

```
Device# show wireless multicast group summary
IPv4 groups
-------------
MGID        Group              Vlan
---------------------------------------
4160        239.255.255.250    12
4161        239.255.255.250    12
 IPv6 groups
-------------
MGID        Group                          Vlan
-------------------------------------------------
```

To verify the IGMP snooping groups, use this command:

```
Device# show ip igmp snooping groups
Vlan      Group                Type        Version     Port List
------------------------------------------------------------------------------
12        239.0.0.158          igmp        v2          Ca2
12        239.255.255.250      igmp        v2          Ca2
```

To verify the IGMP snooping, use this command:

```
Device# show ip igmp snooping
Global IGMP Snooping configuration:
-------------------------------------------
IGMP snooping                : Enabled
Global PIM Snooping          : Disabled
IGMPv3 snooping (minimal)    : Enabled
Report suppression           : Enabled
TCN solicit query            : Disabled
TCN flood query count        : 2
Robustness variable          : 2
Last member query count      : 2
Last member query interval   : 1000
.
.
.
Vlan 11:
--------
IGMP snooping                        : Enabled
Pim Snooping                         : Disabled
IGMPv2 immediate leave               : Disabled
Multicast router learning mode       : pim-dvmrp
CGMP interoperability mode           : IGMP_ONLY
Robustness variable                  : 2
Last member query count              : 2
Last member query interval           : 1000
Vlan 12:
```

```
--------
IGMP snooping                        : Enabled
Pim Snooping                         : Disabled
IGMPv2 immediate leave               : Disabled
Multicast router learning mode       : pim-dvmrp
CGMP interoperability mode           : IGMP_ONLY
Robustness variable                  : 2
Last member query count              : 2
Last member query interval           : 1000
```

To verify the active streams from any sources, use this command:

```
Device# show ip mroute active
Active IP Multicast Sources - sending >= 4 kbps
Group: 239.255.0.1, (?)
   Source: 192.168.33.32 (?)
   Rate: 10 pps/115 kbps(1sec), 235 kbps(last 23 secs), 87 kbps(life avg)
```

To verify the TTL related issues in the path for the given stream, use this command:

```
Device# show ip traffic | include bad hop count
0 format errors, 0 checksum errors, 1529 bad hop count
```

To verify the RPF failures, use this command:

```
                Device# show ip mroute count | inc RPF failed|Other
Other counts: Total/RPF failed/Other drops(OIF-null, rate-limit etc)
  RP-tree: Forwarding: 0/0/0/0, Other: 2/2/0
  RP-tree: Forwarding: 3/0/74/0, Other: 3/0/0
  Source: 32.32.32.32/32, Forwarding: 218747/2/74/1, Other: 218747/0/0
  RP-tree: Forwarding: 0/0/0/0, Other: 0/0/0
  Source: 9.4.168.10/32, Forwarding: 31/0/146/0, Other: 3841861/0/3841830
```

Network address translation (NAT)

Explains how network address translation (NAT) mechanisms map private network addresses to public addresses, supporting secure and scalable internet access.

A network address translation is a networking technique that

- allows multiple local IP addresses within a priavte network to be mapped to public IP addresses
- enables devices on private networks to access external (Internet or Cloud) resources, and
- enhances network security by concealing internal addressing schemes from outside networks.

Port address translation (PAT) enables a single IP address to be shared by multiple hosts through the use of IP and port translations.

L3 access on the controller supports only these NAT use cases:

- translating client traffic in the guest network to reach corporate services, such as Cisco ISE, and
- hiding the private IP addresses of clients from outside networks.

These types of NAT are supported:

- Static address translation (static NAT) allows a one-to-one mapping between local and global addresses. The static translation is useful when a host from the inside is accessible from a fixed address from the outside.
- Dynamic address translation (dynamic NAT or PAT) maps between the client subnet and a public global IP address or source port pool.

  This can be achieved using these:

  - Dynamic NAT without VRF.
  - Dynamic NAT with VRF.

Selective NAT support

This topic explains selective NAT supports, which enable only specific NAT features within Cisco IOS XE 17.13.1.

A selective NAT support is a NAT configuration feature that

- enables only a specified subset of NAT options within a software release, and
- ensures compliance with recent Cisco IOS XE 17.13.1 platform capabilities.

Enable static NAT without VRF (CLI)

Use this procedure to configure static Network Address Translation (NAT) without VRF on a Cisco device using commands.

Configure static NAT to enable a device on the internal network to communicate with external networks using a fixed public IP address using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify an interface and enter the interface configuration mode.

   Example

   ```
   Device(config)# interface interface-type number
   ```

3. Set the IP address for an interface.

   Example

   ```
   Device(config-if)# ip address ip-address mask-address
   ```

4. Connect the interface to the outside network.

   Example

   ```
   Device(config-if)# ip nat outside
   ```

5. Exit the interface configuration mode and enter the global configuration mode.

Example

```
Device(config-if)# end
```

6. Specify a different interface and enter the interface configuration mode.

   Example

   ```
   Device(config)# interface interface-type number
   ```

7. Set the IP address for an interface.

   Example

   ```
   Device(config-if)# ip address ip-address mask-address
   ```

8. Mark the interface as connected to the inside.

   Example

   ```
   Device(config-if)# ip nat inside
   ```

9. Exit the interface configuration mode and enter the global configuration mode. Translate between an inside local address and inside global address.

   Example

   ```
   Device(config-if)# end
   ```

   ```
   Device(config)# ip nat inside source static 10.10.10.100 209.165.200.226
   ```

Enable static NAT with VRF (CLI)

Learn how to configure static NAT with VRF using commands to enable communication between devices in different virtual routing domains.

Establish a static Network Address Translation (NAT) mapping within a specific Virtual Routing and Forwarding (VRF) environment using commands.

Procedure

---

1. Enter the global configuration mode. Specify an interface and enter the interface configuration mode.

   Example

   ```
   Device# configure terminal
   ```

   ```
   Device(config)# interface interface-type-number
   ```

2. Activate multiprotocol VRF on an interface.

Example

```
Device(config-if)# vrf forwarding vrf-name
```

3. Enable IP address on an interface.

   Example

   ```
   Device(config-if)# ip address ip-address mask-address
   ```

4. Mark the interface as connected to the outside.

   Example

   ```
   Device(config-if)# ip nat outside
   ```

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config-if)# end
   ```

6. Specify an interface and enter the interface configuration mode.

   Example

   ```
   Device(config)# interface interface-type-number
   ```

7. Activate multiprotocol VRF on an interface.

   Example

   ```
   Device(config-if)# vrf forwarding vrf-name
   ```

8. Enable IP address on an interface. Mark the interface as connected to the inside.

   Example

   ```
   Device(config-if)# ip address ip-address mask-address
   ```

   ```
   Device(config-if)# ip nat inside
   ```

9. Return to the privileged EXEC mode. Translate between an inside local address and inside global address.

   Example

   ```
   Device(config-if)# end
   ```

   ```
   Device(config)# ip nat inside source static local-ip global-ip vrf vrf_name
   match-in-vrf
   ```

> **Note**
>
> The **match-in-vrf** keyword is optional and required when the same VRF is configured in the inside and outside NAT interface. For more information about match-in-vrf, see
>
> https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipaddr_nat/configuration/xe-16/nat-xe-16-book/iadnat-match-vrf.html

Enable dynamic NAT without VRF (CLI)

Learn how to enable dynamic NAT without VRF using commands, allowing devices on an internal network to share a single public IP address for outbound connections.

Configure a Cisco device to provide dynamic network address translation (NAT) for internal devices without using VRF using commands.

Procedure

1. Enter the global configuration mode. Specify an interface and enter the interface configuration mode.

   Example

   ```
   Device# configure terminal
   ```

   ```
   Device(config)# interface interface-type number
   ```

2. Set the IP address for an interface.

   Example

   ```
   Device(config-if)# ip address ip address mask-address
   ```

3. Mark the interface as connected to the outside.

   Example

   ```
   Device(config-if)# ip nat outside
   ```

4. Specify a different interface and enter the interface configuration mode.

   Example

   ```
   Device(config)# interface interface-type number
   ```

5. Set the IP address for an interface.

Example

```
Device(config-if)# ip address ip address mask-address
```

6. Mark the interface as connected to the inside.

Example

```
Device(config-if)# ip nat inside
```

7. Define a pool of network addresses for NAT.

Example

```
Device(config)# ip nat pool name start-ip end-ip {netmask netmask |
prefix-length prefix-length}
```

8. Define a standard access list for the addresses to be translated.

Example

```
Device(config)# access-list access-list-number permit ip source-address
source-wildcard-bits host destination-address
```

> **Note**
>
> The **host** keyword is optional for access-list configuration. It depends on the type of ACL you want to configure.

9. Establish dynamic source translation with overloading using the defined access list. Return to the privileged EXEC mode.

Example

```
Device(config)# ip nat inside source list access-list-number pool name overload
```

```
Device(config)# exit
```

Enable dynamic NAT with VRF (CLI)

Use this procedure to enable and configure dynamic NAT for a specific VRF on your device using commands.

Enable dynamic Network Address Translation (NAT) on interfaces associated with a specific Virtual Routing and Forwarding (VRF) instance using commands.

Procedure

---

1. Enter the global configuration mode. Specify an interface and enter the interface configuration mode.

Example

```
Device# configure terminal
```

```
Device(config)# interface interface-type-number
```

2. Activate multiprotocol VRF on an interface. Enable IP address on an interface.

   Example

   ```
   Device(config-if)# vrf forwarding vrf-name
   ```

   ```
   Device(config-if)# ip address ip address mask-address
   ```

3. Mark the interface as connected to the outside. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-if)# ip nat outside
   ```

   ```
   Device(config-if)# end
   ```

4. Specify an interface and enter the interface configuration mode. Activate multiprotocol VRF on an interface.

   Example

   ```
   Device(config)# interface interface-type-number
   ```

   ```
   Device(config-if)# vrf forwarding vrf-name
   ```

5. Enable IP address on an interface. Mark the interface as connected to the inside.

   Example

   ```
   Device(config-if)# ip address ip address mask-address
   ```

   ```
   Device(config-if)# ip nat inside
   ```

6. Return to privileged EXEC mode. Define a standard IPv4 access list using a name.

   Example

   ```
   Device(config-if)# end
   ```

   ```
   Device(config)# ip access-list standard name
   ```

   The *name* can be a number from one to 99.

7. Specify the forwarded packet. Exit interface configuration mode and return to the global configuration mode.

Example

```
Device(config-if)# sequence-number permit host-network wildcard-address
```

```
Device(config-if)# exit
```

> **Note**
>
> *sequence-number* refers to the number where the rule should be in the list. Here, lower the sequence number higher the priority for the rule.

8. Define a pool of network addresses for NAT. Establish dynamic source translation with overloading using the defined access list.

Example

```
Device(config)# ip nat pool name start-ip end-ip {netmask netmask |
prefix-length prefix-length}
```

```
Device(config)# ip nat inside source list access-list-number pool name vrf
vrf-name match-in-vrf overload
```

> **Note**
>
> The **match-in-vrf** keyword is optional and required when the same VRF is configured in the inside and outside NAT interface. For more information about match-in-vrf, see
>
> https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipaddr_nat/configuration/xe-16/nat-xe-16-book/iadnat-match-vrf.html

9. Return to the privileged EXEC mode.

Example

```
Device(config)# end
```

Enable timeout for NAT (CLI)

Learn how to set protocol-specific NAT session timeouts using commands.

Adjust the timeout values for NAT translations to optimize device performance using commands.

Procedure

---

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Specify timeouts for NAT translations.

Example

```
Device(config)# ip nat translation [icmp-timeout | tcp-timeout | timeout |
udp-timeout] number-of-seconds
```

These timeout options are supported:

- **icmp-timeout**: ICMP packets timeout.
- **tcp-timeout**: TCP packets timeout.
- **timeout**: Global timeout for all protocol types.
- **udp-timeout**: UDP packets timeout.

3. Return to the privileged EXEC mode.

Example

```
Device(config)# end
```

Verify static NAT details

This section lists the commands and typical outputs for verifying static NAT details, both with and without VRF, on supported devices.

Verify static NAT details without VRF

To verify the static IP NAT statistics without VRF, use this command:

```
Device# show ip nat statistics
Total active translations: 1 (1 static, 0 dynamic; 0 extended)
Outside interfaces:
Vlan62
Inside interfaces:
Vlan55
Hits: 1474 Misses: 0
Reserved port setting disabled provisioned no
Expired translations: 1
Dynamic mappings:
nat-limit statistics:
max entry: max allowed 0, used 0, missed 0
In-to-out drops: 0 Out-to-in drops: 0
Pool stats drop: 0 Mapping stats drop: 0
Port block alloc fail: 0
IP alias add fail: 0
Limit entry add fail: 0
```

To verify the static NAT without VRF on active chassis, use this command:

```
Device# show platform software nat chassis active F0 translation
Pro Inside global Inside local Outside local Outside global
--- 62.1.1.15 155.1.100.1 --- ---
--- 62.1.1.16 155.1.0.4 --- ---
udp 62.1.1.16:33334 155.1.0.4:33334 62.1.1.11:33333 62.1.1.11:33333
udp 62.1.1.16:30000 155.1.0.4:30000 62.1.1.11:30000 62.1.1.11:30000
Total number of translations: 4
```

Verify static NAT details with VRF

To verify the static IP NAT statistics with VRF, use this command:

```
Device# show ip nat statistics
Total active translations: 1 (1 static, 0 dynamic; 0 extended)
Outside interfaces:
Vlan62
Inside interfaces:
Vlan55
Hits: 1474 Misses: 0
Reserved port setting disabled provisioned no
Expired translations: 1
Dynamic mappings:
nat-limit statistics:
max entry: max allowed 0, used 0, missed 0
In-to-out drops: 0 Out-to-in drops: 0
Pool stats drop: 0 Mapping stats drop: 0
Port block alloc fail: 0
IP alias add fail: 0
Limit entry add fail: 0
```

To verify the static NAT with VRF on active chassis, use this command:

```
Device# show platform software nat chassis active F0 translation
Pro Inside global Inside local Outside local Outside global
--- 62.1.1.15 155.1.100.1 --- ---
--- 62.1.1.16 155.1.0.4 --- ---
udp 62.1.1.16:33334 155.1.0.4:33334 62.1.1.11:33333 62.1.1.11:33333
udp 62.1.1.16:30000 155.1.0.4:30000 62.1.1.11:30000 62.1.1.11:30000
Total number of translations: 4
```

Verify dynamic NAT details

This section lists the commands and example outputs used to verify dynamic NAT details, both with and without a VRF, on Cisco devices.

Verify dynamic NAT details without VRF

To verify the dynamic IP NAT statistics without VRF, use this command:

```
Device# show ip nat statistics
Total active translations: 1 (0 static, 1 dynamic; 1 extended)
Outside interfaces:
  Vlan62
Inside interfaces:
  Vlan155
Hits: 3  Misses: 1
 Reserved port setting disabled provisioned no
Expired translations: 0
```

```
Dynamic mappings:
-- Inside Source
[Id: 2] access-list dest_nat_acl pool test_nat_pool refcount 1
 pool test_nat_pool: id 1, netmask 255.255.255.252
    start 62.1.1.101 end 62.1.1.101
    type generic, total addresses 1, allocated 1 (100%), misses 0
longest chain in pool: test_nat_pool's addr-hash: 0, average len 0,chains 0/256
nat-limit statistics:
 max entry: max allowed 0, used 0, missed 0
In-to-out drops: 0  Out-to-in drops: 0
Pool stats drop: 0  Mapping stats drop: 0
Port block alloc fail: 0
IP alias add fail: 0
Limit entry add fail: 0
```

To verify the dynamic NAT without VRF on active chassis, use this command:

```
Device# show platform software nat chassis active F0 translation
Pro  Inside global         Inside local          Outside local         Outside
 global
udp  62.1.1.101:30000      155.1.100.1:30000     62.1.1.11:30000
62.1.1.11:30000
Total number of translations: 1
```

Verify dynamic NAT details with VRF

To verify the dynamic IP NAT statistics with VRF, use this command:

```
Device# show ip nat statistics
Total active translations: 1 (0 static, 1 dynamic; 1 extended)
Outside interfaces:
  Vlan62
Inside interfaces:
  Vlan155
Hits: 3  Misses: 1
 Reserved port setting disabled provisioned no
Expired translations: 0
Dynamic mappings:
-- Inside Source
[Id: 2] access-list dest_nat_acl pool test_nat_pool refcount 1
 pool test_nat_pool: id 1, netmask 255.255.255.252
    start 62.1.1.101 end 62.1.1.101
    type generic, total addresses 1, allocated 1 (100%), misses 0
longest chain in pool: test_nat_pool's addr-hash: 0, average len 0,chains 0/256
nat-limit statistics:
 max entry: max allowed 0, used 0, missed 0
In-to-out drops: 0  Out-to-in drops: 0
Pool stats drop: 0  Mapping stats drop: 0
Port block alloc fail: 0
IP alias add fail: 0
Limit entry add fail: 0
```

To verify the dynamic NAT with VRF on active chassis, use this command:

```
Device# show platform software nat chassis active F0 translation
Pro  Inside global         Inside local          Outside local         Outside
 global
udp  62.1.1.101:30000      155.1.100.1:30000     62.1.1.11:30000
62.1.1.11:30000
Total number of translations: 1
```

Verify NAT details

This section lists commands to verify Network Address Translation (NAT) datapath pool details and statistics, and provides reference details about the output fields and clearing procedures.

To verify the NAT datapath pool details, use this command:

```
Device# show platform hardware chassis active qfp feature nat datapath pool
pool_id 1 type 1 addroute 0 mask 0xfffffffc allocated 0 misses 0 rotary idx
0x0 ahash sz 4 size 1 max_pat_hash_size 1 next 0x0 hash_index 0x32, hilo ports
 0x0 pool mem 0xde480010 flags 0x1 pool_name: test_nat_pool pat_wl 0 no_ports_wl
 0 num_maps 1 num_overload_maps 1 vrf 0x0 port_used tcp 0 udp 0
Conf block info
start 62.1.1.102 end 62.1.1.102 flags 0x0 next 0x0 prev 0x0
TCP PAT block info
UDP PAT block info
ICMP PAT block info
GRE PAT block info
Alloced addr info
```

To verify the NAT datapath statistics, use this command:

```
Device# show platform hardware chassis active qfp feature nat datapath stats
Counter Value
-------------------------------------------------------------------------
number_of_session 0
udp 0
tcp 0
icmp 0
non_extended 0
statics 0
static_net 0
entry_timeouts 0
hits 0
misses 0
cgn_dest_log_timeouts 0
ipv4_nat_alg_bind_pkts 0
ipv4_nat_alg_sd_not_found 0
ipv4_nat_alg_sd_tail_not_found 0
ipv4_nat_rx_pkt 2043
ipv4_nat_tx_pkt 122169
ipv4_nat_flowdb_hits 0
ipv4_nat_stick_rx_pkts 0
ipv4_nat_stick_i2o_pkts 0
ipv4_nat_stick_o2i_pkts 0
ipv4_nat_stick_forus_hits_pkts 0
ipv4_nat_stick_hit_sb 0
ipv4_nat_stick_ha_divert_pkts 0
ipv4_nat_stick_ha_ar_pkts 0
ipv4_nat_stick_ha_tcp_fin 0
ipv4_nat_stick_ha_failed_pkts 0
ipv4_nat_non_natted_in2out_pkts 122165
ipv4_nat_non_nated_out2in_pkts 0
ipv4_nat_bypass_pkts 0
ipv4_nat_unmarked_pkts 0
ipv4_nat_res_port_in2out_pkts 0
ipv4_nat_res_port_out2in_pkts 0
ipv4_nat_ipc_retry_fail 0
ipv4_nat_cfg_rcvd 2
ipv4_nat_cfg_rsp 2
```

To clear the NAT details, use this commands:

```
clear platform software nat chassis active F0 translation forced
clear ip nat statistics
```

Verify NAT timeout details

This section lists the NAT timeout types, their timeout values in seconds, and whether each type is enabled by default.

To verify the NAT timeout details, use this command:

```
Device# show platform software nat chassis active r0 timeout
Dump NAT timeout config
  Type: generic, Timeout (sec): 86400, Enabled: Yes
  Type: tcp, Timeout (sec): 86400, Enabled: Yes
  Type: tcp-pptp, Timeout (sec): 86400, Enabled: Yes
  Type: udp, Timeout (sec): 60, Enabled: Yes
  Type: tcp-fin-reset, Timeout (sec): 60, Enabled: Yes
  Type: tcp-syn, Timeout (sec): 60, Enabled: Yes
  Type: dns, Timeout (sec): 60, Enabled: Yes
  Type: icmp, Timeout (sec): 60, Enabled: Yes
  Type: skinny, Timeout (sec): 60, Enabled: Yes
  Type: icmp-error, Timeout (sec): 60, Enabled: Yes
  Type: esp, Timeout (sec): 300, Enabled: Yes
  Type: rtmap, Timeout (sec): 3600, Enabled: Yes
```

78 IPv6 Layer 3 Access

Topics:

- IPv6 Layer 3 access

This chapter explains IPv6 Layer 3 Access.

IPv6 Layer 3 access

Explains how IPv6 layer 3 access enables controller-based wireless client subnet termination and advanced IPv6 services on the Cisco Catalyst 9800 Series Wireless Controller.

IPv6 Layer 3 access is a controller-based wireless client forwarding capability that

- terminates wireless client IPv6 subnets on switched virtual interfaces (SVIs) on the Cisco Catalyst 9800 Series Wireless Controller
- advertises wireless client routes to the upstream network through OSPFv3, and
- provides VRF-aware IPv6 services such as DHCPv6 relay, IPv6 multicast routing, and NAT64.

In a Layer 2 wireless architecture, an upstream switch or router terminates the wireless client subnets. With IPv6 Layer 3 access, the controller terminates the client subnets and participates in routing. This design can improve scalability, resiliency, and operational efficiency for IPv6-first and dual-stack networks.

IPv6 Layer 3 access supports the coexistence of Layer 2 and Layer 3 WLANs. You can enable Layer 3 access on the wireless policy profiles that require controller-based subnet termination and leave it disabled on other policy profiles.

The controller supports flow based equal-cost multipath (ECMP) forwarding over four equal-cost paths by default. ECMP distributes flows across multiple best paths that have the same routing metric.

Feature history for IPv6 Layer 3 Access

This section explains the feature history details of the IPv6 Layer 3 Access feature.

- This table provides release and related information for the feature explained in this module.
- This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 85: Feature history for IPv6 Layer 3 Access

Feature Name	Release Information	Feature Description
IPv6 Layer 3 Access	Cisco IOS XE 26.2.1	IPv6 Layer 3 Access is a controller-based wireless client forwarding capability that terminates IPv6 client subnets on switched virtual interfaces and advertises client routes to the upstream network through OSPFv3. It provides VRF-aware IPv6 services and improves scalability, resiliency, and operational efficiency for IPv6-first and dual-stack networks.

DHCPv6 relay agents

Explains the function and configuration of DHCPv6 relay agents on wireless controllers, including supported options and related Layer 2 relay operations.

A DHCPv6 relay agent is a network function on the wireless controller that

- encapsulates wireless client DHCPv6 messages into relay-forward messages
- sends the relay-forward messages to a DHCPv6 server through unicast communication, and

- decapsulates the client messages from relay-reply messages and forwards them to wireless clients.

You can configure DHCPv6 server settings on a wireless policy profile or an SVI. You can configure the DHCPv6 VPN option at the wireless policy profile, SVI, or global configuration level.

The configuration scope determines which clients the VPN option applies to. A wireless policy profile configuration applies only to clients associated with that policy profile. An SVI configuration applies to clients on the corresponding VLAN. A global configuration applies to all applicable DHCPv6 clients on the controller.

If the VPN option is configured at more than one scope, the controller applies the most specific configuration. A wireless policy profile configuration takes precedence over an SVI configuration, and an SVI configuration takes precedence over the global configuration.

The controller supports primary and secondary DHCPv6 server addresses at configuration scopes that allow multiple relay destinations. A wireless policy profile supports one DHCPv6 server address.

Supported DHCPv6 relay options

- Interface-ID: The Interface-ID option is enabled by default. No configuration is required.
- Virtual Subnet Selection (VSS): Carries the VRF name of the client interface. This option uses Cisco enterprise ID nine, and suboption three carries the VRF name. The DHCPv6 server can use this information to apply policy for the originating VRF.

The client interface VRF is carried in the VSS option. The server VRF determines how the controller routes relay packets to the DHCPv6 server.

Lightweight DHCPv6 relay agents

A lightweight DHCPv6 relay agent (LDRA) is a Layer 2 relay function that inserts relay-agent information while the controller bridges client traffic. LDRA is the functional equivalent of the DHCPv4 Layer 2 relay agent.

When you enable the VPN option on a wireless policy profile without configuring a DHCPv6 server, the controller applies LDRA processing. The controller automatically adds the mandatory Interface-ID option and adds the VSS option only when you configure it.

How IPv6 Layer 3 client forwarding works

Describes the sequence of events and key actors involved in IPv6 Layer 3 client forwarding through Cisco wireless infrastructure.

These stages describe how IPv6 Layer 3 forwarding works.

Summary

The key components involved in the process are:

- *Wireless client*: Connects to a WLAN that uses a wireless policy profile.
- *Access point*: Connects the wireless client to the controller.
- *Cisco Catalyst 9800 Series Wireless Controller*: Terminates the client SVI and forwards client traffic at Layer 3.
- *Upstream routers*: Exchange routes with the controller through OSPFv3 and forward traffic through the enterprise network.

Workflow

The process involves these stages:

1. The wireless client joins a WLAN whose wireless policy profile has Layer 3 access enabled.
2. The controller terminates the client subnet on the corresponding SVI.
3. OSPFv3 advertises the client subnet to the upstream routers.

4. The controller and the upstream routers select the best routing path. When multiple paths have equal metrics, ECMP distributes flows across the equal-cost paths.
5. The controller forwards client traffic through the selected path.

Result

The controller provides routed IPv6 connectivity for wireless clients and advertises the client subnets to the upstream network.

How DHCPv6 relay agents work

Describes how DHCPv6 relay agents facilitate IPv6 address allocation for wireless clients by relaying messages between clients and DHCPv6 servers.

These stages describe how DHCPv6 relay agents work.

Summary

The key components involved in the process are:

- *Wireless client*: Sends DHCPv6 messages to obtain IPv6 configuration.

  Cisco Catalyst 9800 Series Wireless Controller: Relays and processes DHCPv6 messages.

- *DHCPv6 server*: Allocates IPv6 configuration and returns DHCPv6 replies.

Workflow

The process involves these stages:

1. The wireless client sends a multicast DHCPv6 message.
2. The controller selects the effective relay configuration according to the policy profile, SVI, and global precedence.
3. The controller encapsulates the client message in a relay-forward message and adds the required relay options.
4. The controller converts the client multicast exchange into unicast communication with the DHCPv6 server and uses the server VRF to route the relay packet.
5. The DHCPv6 server returns a relay-reply message.
6. The controller decapsulates the relay-reply message and forwards the server-originated reply to the wireless client.

Result

The wireless client receives IPv6 configuration from an external DHCPv6 server through the controller.

Guidelines for IPv6 Layer 3 forwarding

Ensure your IPv6 Layer 3 forwarding deployment meets all mandatory guidelines for reliable wireless client termination, routing, and DHCP relay functionality.

Ensure that the IPv6 Layer 3 forwarding deployment meets these guidelines:

- Enable Layer 3 access on each wireless policy profile that uses controller-based client subnet termination. Layer 3 access is disabled by default.
- Use Local mode or FlexConnect central switching with central DHCP.
- Use OSPFv3 for dynamic routing and static routes for IPv6 reachability to infrastructure devices.
- Configure an external DHCPv6 server for full DHCPv6 relay operation.
- Apply DHCPv6 relay configuration at the intended scope. A wireless policy profile configuration overrides an SVI configuration, and an SVI configuration overrides the global configuration.
- Use the client interface VRF in the VSS option when the DHCPv6 server requires the originating VRF context.

- Use the server VRF to provide destination routing for relay packets to the DHCPv6 server.

Restrictions for IPv6 Layer 3 forwarding

Lists current restrictions on IPv6 layer 3 forwarding across various wireless deployment models.

These restrictions apply to IPv6 Layer 3 forwarding:

- FlexConnect local switching, MEWLC, and ECA deployments are not supported.
- BGP is not supported for this feature.
- The internal DHCPv6 server is not supported.
- Inter-controller mobility is not supported for Layer 3 access clients.
- A client cannot roam between Layer 2 and Layer 3 WLANs that use different VLANs, interfaces, or VRFs. The controller rejects the roam.
- Mobility anchor and static IP mobility configurations are not supported on a wireless policy profile that has Layer 3 access enabled.

Use cases for IPv6 Layer 3 forwarding

Explains how IPv6 Layer 3 forwarding enables routed client connectivity and supports advanced networking use cases in enterprise wireless environments.

IPv6 Layer 3 forwarding supports these use cases:

- *IPv6-first wireless access*: Provides routed client connectivity in an IPv6-first enterprise network.
- *Dual-stack wireless access*: Uses a single OSPFv3 process for IPv4 and IPv6 address families.
- *VRF-aware segmentation*: Maintains separate routing contexts for client networks and supports VRF-aware DHCPv6 relay.
- *Layer 2 and Layer 3 WLAN coexistence*: Allows selected WLANs to use controller-based Layer 3 termination while other WLANs continue to use Layer 2 termination.
- *Resilient upstream connectivity*: Uses OSPFv3 and flow-based ECMP to advertise client subnets and forward traffic across equal-cost paths.
- *IPv6 multicast services*: Supports PIM sparse mode, PIM source-specific multicast, and MLDv2 for wireless clients.
- *IPv6-to-IPv4 communication*: Supports NAT64 so that IPv6-only wireless clients can access IPv4-only resources.

Enable Layer 3 access on a wireless policy profile

Configure Layer 3 access settings on a wireless policy profile so the controller terminates and routes client traffic.

Enable the controller to terminate and route client traffic for a wireless policy profile.

Layer 3 access is disabled by default. Enable it only on wireless policy profiles that require controller-based client subnet termination.

Before you begin

- Configure the client VLAN and SVI.
- Ensure that the deployment uses Local mode or FlexConnect central switching with central DHCP.

Follow these steps to enable Layer 3 access on a wireless policy profile:

Procedure

1. Enter wireless policy profile configuration mode and enable Layer 3 access.

   Example

   ```
   wireless profile policy policy-profile-name l3-access
   ```

   Use the **no** form of this command to disable Layer 3 access.

2. Verify the Layer 3 access state for the policy profile.

   Example

   ```
   show wireless profile policy detailed policy-profile-name | include L3
   ```

   The command displays the L3 access as enabled.

3. Verify the Layer 3 access state for a connected wireless client.

   Example

   ```
   show wireless client mac-address client-mac-address detail | include L3
   ```

   The command displays the L3 access as enabled for a client.

Layer 3 access is enabled for clients that use the wireless policy profile.

Configure DHCPv6 relay or LDRA

Configure your controller to relay DHCPv6 messages or insert DHCPv6 relay-agent information, using DHCPv6 relay or LDRA at the appropriate scope for wireless clients.

Configure the controller to relay DHCPv6 messages or insert DHCPv6 relay-agent information for wireless clients.

You can configure DHCPv6 relay on a wireless policy profile or an SVI. You can enable the DHCPv6 VPN option at the wireless policy profile, SVI, or global configuration level. Select the configuration scope according to the clients to which the VPN option must apply:

- Wireless policy profile: Applies only to clients associated with the wireless policy profile.
- SVI: Applies to clients on the corresponding VLAN.
- Global configuration: Applies to all applicable DHCPv6 clients on the controller.

If the VPN option is configured at more than one scope, the controller applies the configuration in this order of precedence:

- Wireless policy profile
- SVI
- Global configuration

Use LDRA when the controller bridges client traffic and no DHCPv6 server is configured on the wireless policy profile.

Before you begin

- Obtain the IPv6 address of the external DHCPv6 server. You can configure one DHCPv6 server address on a wireless policy profile.

- Configure the VRF that provides reachability to the DHCPv6 server.
- Ensure that routes to the external DHCPv6 server are available.
- Select the configuration scope required by the deployment.

Follow these steps to configure DHCPv6 relay or LDRA:

Procedure

1. Select one of the following configurations according to the required scope and forwarding behavior.

   Full DHCPv6 relay on a wireless policy profile.

   a) Configure the DHCPv6 server address and server VRF. Enable the VPN option when the DHCPv6 server requires the client VRF context.

      Example

      ```
      wireless profile policy policy-profile-name
          ipv6 dhcp server dhcpv6-server-address vrf server-vrf-name
          ipv6 dhcp relay option vpn
      ```

      You can configure only one DHCPv6 server address on a wireless policy profile.

   LDRA on a wireless policy profile.

   a) Do not configure a DHCPv6 server on a wireless policy profile that uses LDRA. Enable the VPN option to apply LDRA processing.

      Example

      ```
      wireless profile policy policy-profile-name
          ipv6 dhcp relay option vpn
      ```

      The controller automatically inserts the mandatory Interface-ID option. It adds the VSS option only when the VPN option is configured.

   DHCPv6 relay on an SVI.

   a) Configure the relay destination. Include the server VRF, link address, source address, or source interface when required by the deployment.

      Example

      ```
      interface Vlan vlan-id
          ipv6 dhcp relay destination dhcpv6-server-address link-address link-address |
       source-address source-address
          ipv6 dhcp relay destination vrf server-vrf-name dhcpv6-server-address
      link-address link-address | source-address source-address
          ipv6 dhcp relay source-interface source-interface
          ipv6 dhcp relay option vpn
      ```

      Use only the relay destination form and optional parameters required by the deployment.

   Global DHCPv6 relay VPN option.

   a) Enable the VPN option globally when it must apply to all applicable DHCPv6 clients on the controller.

A wireless policy profile or SVI configuration takes precedence over the global configuration.

Example

```
ipv6 dhcp-relay option vpn
```

2. Connect a wireless client and confirm that it receives IPv6 configuration from the external DHCPv6 server.

The controller applies the selected DHCPv6 relay or LDRA behavior at the configured scope.

Configure OSPFv3 for IPv6 Layer 3 access

Configure OSPFv3 to advertise wireless client subnets and learn upstream IPv6 routes for Layer 3 access.

Enable IPv6 routing on wireless client VLANs by configuring OSPFv3 to advertise subnets and learn upstream routes.

OSPFv3 supports both IPv4 and IPv6 address families, including VRF-specific deployments.

Before you begin

- Configure the client VLAN, SVI, IPv6 address, and VRF.
- Confirm the OSPFv3 process ID and area for the deployment.

Follow these steps to configure OSPFv3 for IPv6 Layer 3 access:

Procedure

---

1. Configure the OSPFv3 process and IPv6 unicast address family. Configure a VRF-specific address family when required.

   Configure BFD and nonstop routing (NSR) only when required by the network design.

2. Associate the client SVI with the OSPFv3 process and area for IPv6.

   Example

   ```
   interface Vlan vlan-id
   vrf forwarding vrf-name
   ipv6 address ipv6-prefix prefix-length
   ipv6 enable
   ospfv3 process-id ipv6 area area-id
   ```

3. For a dual-stack deployment, configure the IPv4 address family and associate the SVI with the OSPFv3 process for IPv4.
4. Verify the OSPFv3 neighbors and learned IPv6 routes.

The controller exchanges IPv6 routes with its OSPFv3 neighbors and advertises the configured client subnets.

Configure IPv6 multicast routing for wireless clients

Configure IPv6 multicast routing on Layer 3 wireless clients to support PIM sparse mode, source-specific multicast, and MLDv2.

Enable Layer 3 wireless clients to receive IPv6 multicast traffic through proper routing and group management.

IPv6 multicast routing for wireless clients allows efficient group-based communication, supporting advanced protocols such as PIM and MLD.

Before you begin

- Configure the client VRF and IPv6 SVI.
- Identify the rendezvous point and wireless IPv6 multicast group for the deployment.

Follow these steps to configure IPv6 multicast routing for wireless clients:

Procedure

1. Enable IPv6 unicast routing, multicast routing for the client VRF, and MLD snooping.

   Example

   ```
   ipv6 unicast-routing
   ipv6 multicast-routing vrf vrf-name
   ipv6 mld snooping
   ipv6 mld vrf vrf-name state-limit state-limit
   ```

2. Configure the IPv6 PIM rendezvous point for the client VRF.

   Example

   ```
   ipv6 pim vrf vrf-name rp-address rp-ipv6-address
   ```

3. Enable wireless multicast and configure the AP CAPWAP IPv6 multicast group.

   Example

   ```
   wireless multicast
   wireless multicast ipv6 ipv6-multicast-group
   ```

4. Verify the wireless multicast configuration and IPv6 group membership.

The controller forwards supported IPv6 multicast traffic for Layer 3 wireless clients.

Part VIII

System Management

Topics:

- Network Mobility Services Protocol
- Application Visibility and Control
- Software-Defined Application Visibility and Control
- Cisco Hyperlocation
- FastLocate for Cisco Catalyst Series Access Points
- IoT Services Management
- IoT Module Management in the Controller
- EDCA Parameters
- Adaptive Client Load-Based EDCA
- 802.11 parameters and Band Selection
- NBAR Protocol Discovery
- Conditional Debug, Radioactive Tracing, and Packet Tracing
- Aggressive Client Load Balancing
- RF based Automatic AP Load Balancing
- Accounting Identity List
- Accounting Session ID
- Interim Accounting
- Wireless Multicast
- Map-Server Per-Site Support
- Volume Metering
- Enabling Syslog Messages in Access Points and Controller for Syslog Server
- Login Banner
- Wi-Fi Alliance Agile Multiband
- SNMP Traps
- Disabling Clients with Random MAC Address
- Dataplane Packet Logging
- Streaming Telemetry
- Application Performance Monitoring
- Wireless Clients Threshold Warning
- Intelligent Capture Hardening
- Amazon S3 Support
- Amazon Web Services CloudWatch
- Kernel Minidump and Trustzone Upgrade

79 Network Mobility Services Protocol

Topics:

- Network mobility services protocol
- Probe RSSI location
- RFID tag support

Network mobility services protocol

Explains the Network Mobility Services Protocol (NMSP) and its role in secure, bi-directional wireless communication between controllers and Cisco Connected Mobile Experiences (CMX) clients.

A Network Mobility Services Protocol (NMSP) is a secure two-way wireless communication protocol that

- enables bi-directional data exchange between Cisco wireless controllers and Cisco Connected Mobile Experiences (CMX)
- supports both publish-subscribe and request-reply communication models over connection-oriented transports such as TLS, and
- allows controllers to provide multiple services including device location, probe RSSI, hyperlocation, and wIPS to multiple CMX clients.

The wireless infrastructure runs the NMSP server, while Cisco Connected Mobile Experiences (Cisco CMX) acts as the NMSP client.

Cisco CMX communicates to the controller over a routed IP network. Typically, Cisco CMX establishes a subscription to receive services data from the controller in the form of periodic updates. The controller acts as a data publisher, broadcasting services data to multiple CMXs. In addition to subscriptions, Cisco CMX can send requests to the controller. The controller sends a response to Cisco CMX.

This is a list of the Network Mobility Services Protocol features:

- NMSP is disabled by default.
- NMSP communicates with Cisco CMX using TCP, and uses TLS for encryption.
- TCP and TLS support the wireless intrusion prevention system (wIPS).
- The system supports bidirectional communication, and Cisco CMX sends messages asynchronously over the established channel.

> **Note**
>
> HTTPS is not supported for data transport between the controller and Cisco CMX.

Radioactive tracing for NMSP

This feature collects and provides all CMX-related events.

A radioactive trace event is a CMX service feature that

- enables the collection of all CMX-related events from network controllers
- configures secure connections through SNMP and CLI for real-time event tracing, and
- provides comprehensive troubleshooting by gathering controller logs associated with the CMX IP via RA tracing.

This feature also configures the CMX hash key on the controller, and requests the controller to open an NMSP connection.

Enable NMSP on premises services (CLI)

Enable NMSP on premises services using commands.

Enable Network Mobility Services Protocol (NMSP) on your network controller to support communication with on-premises services.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable NMSP on premises services.

   Example

   ```
   Device(config)# nmsp enable
   ```

   Note

   By default, the NMSP is enabled on the controller.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   Alternatively, press Ctrl-Z to exit global configuration mode.

Modify the NMSP notification interval for clients, RFID tags, and rogues (CLI)

Modify the NMSP notification interval for clients, RFID tags, and rogues.

NMSP manages communication between the Cisco Connected Mobile Experience (Cisco CMX) and the controller for incoming and outgoing traffic. If your application requires more frequent location updates, you can set the NMSP notification interval to a value between one and 180 seconds for clients, active RFID tags, rogue access points, and clients.

Note

The TCP port (16113) used for communication between the controller and Cisco CMX must be open on any firewall separating the two systems to enable NMSP operation.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Set the NMSP notification interval value for clients, RFID tags, rogue clients, and APs.

Example

```
Device(config)# nmsp notification interval {rssi {clients | rfid | rogues {ap
 | client} | spectrum | interferers} interval 50}
```

interval-NMSP notification interval value, in seconds for RSSI measurement. The valid range is from one to 180.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

Alternatively, you can also press ***Ctrl-Z*** to exit global configuration mode.

Modify the NMSP notification threshold for clients, RFID tags, and rogues (CLI)

Modify the NMSP notification threshold for clients, RFID tags, and rogues using commands.

Change the threshold for NMSP notifications based on RSSI for clients, RFID tags, and rogue devices.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the NMSP notification threshold for clients, RFID tags, rogue clients, and APs.

Example

```
Device(config)# location notify-threshold {clients | rogues ap | tags}
threshold 5
```

The *threshold*- RSSI threshold value is in dB. The valid range is from zero to 10, with a default value of zero..

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

Alternatively, press ***Ctrl-Z*** to exit global configuration mode.

Configure NMSP strong cipher (CLI)

Configure NMSP Strong Cipher using commands.

Enable strong cipher suites for NMSP server to enhance security for device communication.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable strong ciphers for NMSP server.

   Example

   ```
   Device(config)# nmsp strong-cipher
   ```

   Strong ciphers for NMSP server contains, "ECDHE-RSA-AES128-GCM-SHA256:, ECDHE-ECDSA-AES128-GCM-SHA256:, AES256-SHA256:AES256-SHA:, and AES128-SHA256:AES128-SHA".

   Normal cipher suite contains, "ECDHE-RSA-AES128-GCM-SHA256:, ECDHE-ECDSA-AES128-GCM-SHA256:, and AES128-SHA".

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   Alternatively, press ***Ctrl-Z*** to exit global configuration mode.

Verify NMSP settings

Verify NMSP settings using the commands controller .

To view the NMSP capabilities of the controller, use the command:

```
Device# show nmsp capability
Service            Subservice
----------------------------
RSSI               Rogue, Tags, Mobile Station,
Spectrum           Aggregate Interferer, Air Quality, Interferer,
Info               Rogue, Mobile Station,
Statistics         Rogue, Tags, Mobile Station,
AP Monitor         Subscription
On Demand Services Device Info
AP Info            Subscription
```

To view the NMSP notification intervals, use the command:

```
Device# show nmsp notification interval
NMSP Notification Intervals
```

```
--------------------------

RSSI Interval:
Client              : 2 sec
RFID                : 50 sec
Rogue AP            : 2 sec
Rogue Client        : 2 sec
Spectrum            : 2 sec
```

To view the connection-specific statistics counters for all CMX connections, use the command:

```
Device# show nmsp statistics connection
NMSP Connection Counters
----------------------
 CMX IP Address: 10.22.244.31, Status: Active
 State:
 Connections : 1
 Disconnections : 0
 Rx Data Frames : 13
 Tx Data Frames : 99244
 Unsupported messages : 0
 Rx Message Counters:
 ID  Name                                  Count
 ------------------------------------------------
 1  Echo Request                          6076
 7  Capability Notification                  2
 13  Measurement Request                     5
 16  Information Request                     3
 20  Statistics Request                      2
 30  Service Subscribe Request               1

 Tx Message Counters:
 ID  Name                                  Count
 ------------------------------------------------
 2  Echo Response                          6076
 7  Capability Notification                   1
 14  Measurement Response                    13
 15  Measurement Notification             91120
 17  Information Response                     6
 18  Information Notification              7492
 21  Statistics Response                      2
 22  Statistics Notification                305
 31  Service Subscribe Response               1
 67  AP Info Notification                   304
```

To view the common statistic counter of the controller 's NMSP service, use the command:

```
Device# show nmsp statistics summary
NMSP Global Counters
--------------------
Number of restarts                :

SSL Statistics
--------------------
Total amount of verifications      : 6
Verification failures              : 6
Verification success               : 0
Amount of connections created      : 8
Amount of connections closed       : 7
Total amount of accept attempts    : 8
Failures in accept                 : 0
Amount of successful accepts       : 8
```

```
Amount of failed registrations     : 0

AAA Statistics
--------------------
Total amount of AAA requests       : 7
Failed to send requests            : 0
Requests sent to AAA               : 7
Responses from AAA                 : 7
Responses from AAA to validate     : 7
Responses validate error           : 6
Responses validate success         : 1
```

To view the overall NMSP connections, use the command:

```
Device# show nmsp status
NMSP Status
-----------

CMX IP Address   Active    Tx Echo Resp  Rx Echo Req   Tx Data     Rx Data
  Transport
-------------------------------------------------------------------------------------
127.0.0.1        Active    6             6             1           2
  TLS
```

To view all mobility services subscribed by all CMXs, use the command:

```
Device# show nmsp subscription detail
CMX IP address 127.0.0.1:
Service            Subservice
-----------------------------
RSSI               Rogue, Tags, Mobile Station,
Spectrum
Info               Rogue, Mobile Station,
Statistics         Tags, Mobile Station,
AP Info            Subscription
```

To view all mobility services subscribed by a specific CMX, use the command:

```
Device# show nmsp subscription detail
CMX IP address 127.0.0.1:
Service            Subservice
-----------------------------
RSSI               Rogue, Tags, Mobile Station,
Spectrum
Info               Rogue, Mobile Station,
Statistics         Tags, Mobile Station,
AP Info            Subscription

Device# show nmsp subscription summary
Service            Subservice
-----------------------------
RSSI               Rogue, Tags, Mobile Station,
Spectrum
Info               Rogue, Mobile Station,
Statistics         Tags, Mobile Station,
AP Info            Subscription
```

Examples: NMSP settings configuration

NMSP settings configuration examples.

This example shows how to configure the NMSP notification interval for RFID tags:

```
Device# configure terminal
Device(config)# nmsp notification interval rssi rfid 50
Device(config)# end
Device# show nmsp notification interval
```

This example shows how to configure the NMSP notification interval for clients:

```
Device# configure terminal
Device(config)# nmsp notification interval rssi clients 180
Device(config)# end
Device# show nmsp notification interval
```

NMSP by AP groups with subscription list from CMX

Cisco CMX group support allows sending only the required (NMSP) data to Cisco CMX.

An NMSP subscription by AP groups is a network data management feature that

- enables Cisco CMX to subscribe only to specific Network Mobility Services Protocol (NMSP) data from designated AP groups based on the active services in the wireless controller
- helps distribute and balance network data load among multiple CMX servers, and
- restricts NMSP data transmission to those services enabled on the wireless controller.

Cisco CMX group support lets you send the required Network Mobility Services Protocol (NMSP) data to Cisco CMX, whether it is deployed on-premises or in the cloud.

You can create a CMX AP group on the Cisco CMX server, assign a unique name, and group APs under it.

> **Note**
>
> The CMX AP group lists Cisco APs managed by Cisco CMX for location services. This group is different from the AP groups on the wireless controller.

This feature supports these services:

- Client
- Probe client filtering
- Hyperlocation
- BLE Services

> **Note**
>
> NMSP subscription is available only for services enabled on the wireless controller.

Verify NMSP by AP groups with subscription list from CMX

Verify NMSP by AP Groups with subscription list from CMX.

To verify mobility services group subscription summary of all CMX connections, use the command:

```
Device# show nmsp subscription group summary

CMX IP address: 127.0.0.1
Groups subscribed by this CMX server:
Group name: Group1
```

To view the services that are subscribed for an AP group by a CMX connection, use the command:

```
Device# show nmsp subscription group details services cmx-IP-addrress

CMX IP address: 127.0.0.1
CMX Group name: Group1
CMX Group filtered services:
Service            Subservice
----------------------------
RSSI               Mobile Station,
Spectrum
Info
Statistics
```

To view the AP MAC list that is subscribed for an AP group by a CMX connection, use the command:

```
Device# show nmsp subscription group detail ap-list group-name cmx-IP-addrress

CMX IP address: 127.0.0.1
CMX Group name: Group1
CMX Group AP MACs:
: 0000.0000.7002  0000.0000.6602  0099.0000.0002  0000.00bb.0002
  0000.0000.5502  0000.0000.5002  0033.0000.0002  00d0.0000.0002
  0010.0010.0002  0000.0006.0002  0000.0002.0002  0000.0000.4002
  0000.0099.0002  0000.0000.a002  0000.7700.0002  0022.0000.0002
  0000.0000.0092  0000.0000.0082  0000.0000.0302  aa00.0000.0002
  0000.0050.0042  0000.0d00.0002  0000.0000.0032  0000.00cc.0002
  0000.0088.0002  2000.0000.0002  1000.0000.0002  0100.0000.0002
  0000.0000.0002  0000.0000.0001  0000.0000.0000
```

To view CMX-AP grouping details for all CMXs, use the command:

```
Device# show nmsp subscription group detail all
CMX IP address: 127.0.0.1
Groups subscribed by this CMX server:
Group name: Group1
CMX Group filtered services:
Service            Subservice
----------------------------
```

```
RSSI               Mobile Station,
Spectrum
Info
Statistics

CMX Group AP MACs:
: 0000.0000.0003  0000.0000.0002  0000.0000.0001

Group name: Group2
CMX Group filtered services:
Service            Subservice
----------------------------
RSSI               Tags,
Spectrum
Info
Statistics

CMX Group AP MACs:
: 0000.0000.0300  0000.0000.0200  0000.0000.0100

Group name: Group3
CMX Group filtered services:
Service            Subservice
----------------------------
RSSI               Rogue,
Spectrum
Info
Statistics

CMX Group AP MACs:
: 0000.0003.0000  0000.0002.0000  0000.0001.0000
```

To view all the AP lists subscribed by all CMXs, use the command:

```
Device# show nmsp subscription group detail ap-list
```

To view all the services subscribed by all CMXs, use the command:

```
Device# show nmsp subscription group detail services
```

Probe RSSI location

Explains how Probe RSSI locations enable wireless controllers and Cisco CMX to use signal strength data from client probes for advanced location and network management functions.

A probe RSSI location is a wireless networking feature that

- allows wireless controllers and Cisco CMX to collect and use the received signal strength indication (RSSI) from client probe requests associated with the identified Service Set Identifiers (SSIDs)
- enables functions such as load balancing, coverage hole detection, and location updates, and
- supports the precise tracking of wireless clients as they move within the network.

After connecting to an AP, the client scans in the background and keeps a list of strong APs. If it loses connection, the client uses this list to reconnect. The APs in the WLAN gather probe requests, RSSI data, and MAC addresses of wireless clients, then forward this information to the wireless controller s. Cisco CMX receives this data from the controller and calculates the client's updated location.

Configure probe RSSI (CLI)

Configure probe RSSI using commands.

Improve client location accuracy by configuring probe RSSI handling and related settings using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable filtering of unacknowledged probe requests from AP to improve the location accuracy.

   Example

   ```
   Device(config)# wireless probe filter
   ```

   Filtering is enabled by default.

   Use the **no** form of the command to disable the feature. This forwards both acknowledged and unacknowledged probe requests to the controller.

3. Configure the number of probe request reported to the wireless controller from the AP for the same client on a given interval.

   Example

   ```
   Device(config)# wireless probe limit limit-value 10 interval 100
   ```

   Use the **no** form of the command to revert to the default limit, which is two probes at an interval of 500 ms.

4. Enable the reporting of probes from clients having locally administered MAC address.

   Example

   ```
   Device(config)# wireless probe locally-administered-mac
   ```

5. Set the probe RSSI measurement updates to a more accurate algorithm but with more CPU overhead.

   Example

   ```
   Device(config)# location algorithm rssi-average
   ```

6. Optional: Set the probe RSSI measurement updates to a faster algorithm with smaller CPU overhead, but less accuracy.

   Example

   ```
   Device(config)# location algorithm simple
   ```

   Use the **no** form of the command to revert the algorithm type to the default one, that is *rssi-average*.

7. Configure the timeout for RSSI values.

Example

```
Device(config)# location expiry client interval 300
```

The **no** form of the command sets it to a default value of 15.

8. Configure the notification threshold for clients.

Example

```
Device(config)# location notify-threshold client threshold-db 5
```

The **no** form of the command sets it to a default value of zero.

9. Configure half life when averaging two RSSI readings.

Example

```
Device(config)# location rssi-half-life client time-in-seconds 20
```

To disable this option, set the value to zero.

What to do next

Use the **show wireless client probing** command to view each probing client (associated and probing only) by batch of 10 MAC addresses.

RFID tag support

Explains the role and characteristics of RFID tags in wireless asset tracking.

An RFID tag is a small wireless battery-powered tracking device that

- broadcasts its own signal for real-time location tracking, and
- operates using special 802.11 packets processed by APs, wireless controllers, and Cisco CMX.

The controller enables you to configure radio frequency identification (RFID) tag tracking. Only active RFIDs are supported. A combination of active RFID tags and wireless controller allows you to track the current location of equipment. Use *Active* tags to track high-value assets in *closed-loop* systems. These tags remain within the control premises of the owner.

General guidelines

- You can verify the RFID tags on the controller .
- The controller supports high availability for RFID tags.

Configure RFID tag support (CLI)

Configure RFID tag support using commands.

Enable and configure RFID tag tracking features for WLANs managed by the device using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable RFID tag tracking.

   Example

   ```
   Device(config)# wireless rfid
   ```

   The default value is enabled.

   Use the **no** form of this command to disable RFID tag tracking.

3. Configure the RFID tag data timeout value to cleanup the table.

   Example

   ```
   Device(config)# wireless rfid timeout timeout-value 90
   ```

   The timeout value is the amount of time that the controller maintains tags before expiring them. For example, if a tag is configured to beacon every 30 seconds, we recommend that you set the timeout value to 90 seconds (approximately three times the beacon value). The default value is 1200 seconds.

Verify RFID tag support

Verify RFID tag support details.

To view the summary of RFID tags that are clients, use the command:

```
Device# show wireless rfid client
```

To view the detailed information for an RFID tag, use the command:

```
Device# show wireless rfid detail rfid-mac-address

RFID address  000c.cc96.0001
Vendor Cisco
Last Heard 6 seconds  ago
Packets Received 187
Bytes Received 226

Content Header
==============
  CCX Tag Version 0
  Tx power: 12
  Channel: 11
  Reg Class: 4
CCX Payload
==============
  Last Sequence Control 2735
  Payload length 221
```

```
  Payload Data Hex Dump:
  00000000  00 02 00 00 01 09 00 00  00 00 0c b8 ff ff ff 02  |................|

  00000010  07 42 03 20 00 00 0b b8  03 4b 00 00 00 00 00 00  |.B. .....K......|

  00000020  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|

  00000030  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|

  00000040  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
```

To view the summary information for all known RFID tags, use the command:

```
Device# show wireless rfid summary

Total RFID entries: : 16
Total Unique RFID entries : 16
RFID ID VENDOR Closet AP RSSI Time Since Last Heard
0012.b80a.c791 Cisco 7069.5a63.0520 -31 3 minutes 30 seconds ago
0012.b80a.c953 Cisco 7069.5a63.0460 -33 4 minutes 5 seconds ago
0012.b80b.806c Cisco 7069.5a63.0520 -46 15 seconds ago
0012.b80d.e9f9 Cisco 7069.5a63.0460 -38 4 minutes 28 seconds ago
0012.b80d.ea03 Cisco 7069.5a63.0520 -43 4 minutes 29 seconds ago
0012.b80d.ea6b Cisco 7069.5a63.0460 -39 4 minutes 26 seconds ago
0012.b80d.ebe8 Cisco 7069.5a63.0520 -43 3 minutes 21 seconds ago
0012.b80d.ebeb Cisco 7069.5a63.0520 -43 4 minutes 28 seconds ago
0012.b80d.ec48 Cisco 7069.5a63.0460 -42 4 minutes 7 seconds ago
0012.b80d.ec55 Cisco 7069.5a63.0520 -41 1 minute 52 seconds ago
```

To view the location-based system RFID statistics, use the command:

```
Device# show wireless rfid stats

RFID stats :
==============
RFID error db full : 0
RFID error invalid paylod : 0
RFID error invalid tag : 0
RFID error dot11 hdr : 0
RFID error pkt len : 0
RFID error state drop : 0
RFID total pkt received : 369
RFID populated error value : 0
RFID error insert records : 0
RFID error update records : 0
RFID total insert record : 16
RFID ccx payload error : 0
RFID total delete record : 0
RFID error exceeded ap count : 0
RFID error record remove : 0
RFID old rssi expired count: 0
RFId smallest rssi expireed count : 0
RFID total query insert : 0
RFID error invalid rssi count : 0
```

To view the NMSP notification interval, use the command:

```
Device# show nmsp notification interval
NMSP Notification Intervals
---------------------------
```

```
RSSI Interval:
  Client              : 2 sec
  RFID                : 50 sec
  Rogue AP            : 2 sec
  Rogue Client        : 2 sec
  Spectrum            : 2 sec
```

80 Application Visibility and Control

Topics:

- Application visibility and control
- Default DSCP on AVC
- AVC-based selective reanchoring

Application visibility and control

Explains how application visibility and control enables real-time identification, monitoring, and management of network applications using deep packet inspection and Flexible NetFlow integration.

Application visibility and control is a wireless network feature set that

- enables real-time identification and monitoring of applications using deep packet inspection
- allows creation of policy rules to manage application bandwidth and usage, and
- integrates with Flexible NetFlow (FNF) to report traffic statistics per application or protocol.

Application visibility and control (AVC) is a subset of the Flexible NetFlow (FNF) package. AVC provides traffic information. The AVC feature uses a distributed approach that benefits from NBAR running on the AP or controller. The goal is to run deep packet inspection (DPI) and report the results using FNF messages.

The NBAR2 engine analyzes and recognizes traffic flows. Each specific flow is marked with the detected protocol or application. You can use this per-flow information for application visibility through FNF. After application visibility is established, you can define control rules with policing mechanisms for a client.

Guidelines for application visibility and control

Guidelines for application visibility and control

- Using AVC rules, bandwidth for a particular application can be limited for all clients joined on the WLAN. These bandwidth contracts coexist with per-client downstream rate limiting. The per-client limits take precedence over the per-application rate limits.
- The FNF feature is supported in wireless deployments. It relies on NetFlow enablement on the controller for all modes: FlexConnect, local and Fabric.
- In local mode, NBAR runs on the controller hardware. Client traffic flows through the data plane of the controller using the AP CAPWAP tunnels.
- In FlexConnect or Fabric mode, NBAR runs on the AP. Only statistics are sent to the controller . In these two modes, APs regularly send FNFv9 reports to the controller . The FNF feature uses these reports to provide application statistics reported by AVC.
- In Fabric mode, FNF cache is not populated. The system relays FNFv9 reports as they arrive. As a result, some flow monitor configuration parameters, such as cache timeout, are not used.

The behavior of the AVC solution changes based on the wireless deployment mode. The next sections describe the commonalities and differences in all scenarios.

Local Mode

- NBAR is enabled on the controller .
- AVC does not push the FNF configuration to the APs.
- Roaming events are ignored.

  However, AVC supports Layer 3 roaming in local mode because traffic flows through the anchor controller, where NBAR first processed the client's traffic when the client joined.
- IOSd needs to trigger NBAR attach.
- The solution supports flow monitor cache and NetFlow exporter.

FlexConnect Mode

- NBAR is enabled on an AP.
- AVC pushes the FNF configuration to the APs.
- Context transfer for roaming is supported in AVC-FNF.
- Flow monitor cache is supported.
- NetFlow exporter is supported.

Fabric Mode

- NBAR is enabled on an AP.
- AVC pushes the FNF configuration to the APs.
- Context transfer for roaming is supported in AVC-FNF.
- Flow monitor cache is not supported.
- NetFlow exporter is supported. For the C9800 on Catalyst switches for SDA, FNF cache is not available on the device.

Prerequisites for application visibility and control

Prerequisites for application visibility and control

- The APs should be AVC capable.

  However, this requirement does not apply in Local mode.
- Configure the application visibility feature with FNF to enable AVC control (QoS).

Restrictions for application visibility and control

Restrictions for application visibility and control.

- IPv6 packet classification, including ICMPv6 traffic, is not supported in FlexConnect mode or Fabric mode. It is supported in Local mode.
- Layer 2 roaming is not supported across controller .
- Multicast traffic is not supported.
- AVC is supported only on the following APs:
  - Cisco Catalyst 9100 Series Access Points
  - Cisco Aironet 1800 Series Access Points
  - Cisco Aironet 2700 Series Access Point
  - Cisco Aironet 2800 Series Access Point
  - Cisco Aironet 3700 Series Access Points
  - Cisco Aironet 3800 Series Access Points
  - Cisco Aironet 4800 Series Access Points
  - Cisco Industrial Wireless 3702 Access Point
- AVC is not supported on Cisco Aironet 702W, 702I (128 M memory), and 1530 Series APs.

- Only applications recognized by App Visibility can be used for applying QoS control.
- Data link is not supported for NetFlow fields in AVC.
- You cannot map the same WLAN profile to both the AVC-not-enabled policy profile and the AVC-enabled policy profile.
- AVC is not supported on the management port (Gig 0/0).
- NBAR-based QoS policy configuration is allowed only on wired physical ports. Policy configuration is not supported on virtual interfaces. For example, VLAN, port channel and other logical interfaces.
- NBAR cannot classify traffic accurately when SaaS applications use end-to-end encryption, QUIC, or DoH because encryption affects classification. In these cases, encrypted traffic including DoH and QUIC without SNI prevents NBAR from sending the correct Protocol ID, which causes traffic classification issues.

When AVC is enabled, the profile supports a maximum of 23 rules, including the default DSCP rule. If the number of rules exceeds 23, the AVC policy will not be sent to the AP.

Configure AVC

To configure AVC, follow these steps:

Enable monitoring and policy enforcement for application usage on the wireless network.

Procedure

1. Create a flow monitor using the **record wireless avc basic** command.
2. Create a wireless policy profile.
3. Apply the flow monitor to the wireless policy profile.
4. Create a wireless policy tag.
5. Map the WLAN to the policy profile.
6. Attach the policy tag to the APs.

Create a flow monitor (CLI)

Create a flow monitor using commands.

The NetFlow configuration requires a flow record, a flow monitor, and a flow exporter. Make this configuration your first step in the overall AVC configuration.

Note

In FlexConnect mode and Local mode, the default values for **cache timeout active** and **cache timeout inactive** commands do not provide optimal AVC performance. Set both values to 60 in the flow monitor.

For Fabric mode, the cache timeout configuration does not apply.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Create a flow monitor.

Example

```
Device(config)# flow monitor monitor-name fm_avc
```

3. Specify the basic IPv4 or IPv6 wireless AVC flow template.

Example

```
Device(config-flow-monitor)# record wireless avc {ipv4 | ipv6} basic
```

> **Note**
>
> If you want to have both Application Performance Monitoring (APM) and AVC-FNF in the device simultaneously, use the **record wireless avc {ipv4 | ipv6} assurance** command, which is a superset of the fields contained in **record wireless avc {ipv4 | ipv6} basic** command. If the containing flow monitor is configured with the local exporter using **destination wlc local**command, AVC-FNF will populate the statistics exactly as that of the **record wireless avc {ipv4 | ipv6} basic** configuration. As a result, both APM and AVC-FNF can be configured simultaneously with two flow monitors per direction, per IP version, in local (central switching) mode.

> **Note**
>
> The **record wireless avc basic** command is same as **record wireless avc ipv4 basic** command. However, **record wireless avc ipv4 basic** command is not supported in FlexConnect or Fabric modes. In such scenarios, use the **record wireless avc basic** command.

4. Set the active flow timeout in seconds.

Example

```
Device(config-flow-monitor)# cache timeout active value 60
```

5. Set the inactive flow timeout in seconds.

Example

```
Device(config-flow-monitor)# cache timeout inactive value 60
```

Configure a flow monitor (GUI)

Configure a flow monitor using GUI.

Set up a flow monitor and export data to a collector using a pre-configured flow exporter.

Before you begin
Ensure you have already created a flow exporter.

Procedure

1. Choose **Configuration** > **Services** > **Application Visibility** and go to the **Flow Monitor** tab.
2. In the **Monitor** area, click **Add** to add a flow monitor.
3. In the **Flow Monitor** window, add a flow monitor and a description.
4. Select the Flow exporter from the drop-down list to export data from the flow monitor to the collector.

   Note

   To export wireless NetFlow data, use these templates:

   - ETA (Encrypted Traffic Analysis)
   - wireless avc basic
   - wireless avc basic IPv6

5. Click **Apply to Device** to save the configuration.

Create a flow record (CLI)

Create a flow record using commands.

The default flow record cannot be edited or deleted. If you require a new flow record, create one and map it to the flow monitor from the CLI.

Procedure

1. Create a flow record.

   Example

```
Device(config)# flow record flow_record_name record1
```

> **Note**
>
> When a custom flow record is configured in FlexConnect and Fabric modes, the optional fields (fields that are not present in record wireless avc basic) are ignored.

2. Optional: Describe the flow record as a maximum 63-character string.

   Example

   ```
   Device(config-flow-record)# description IPv4flow
   ```

3. Specify a match to the IPv4 protocol and specify a match to the IPv4 source address-based field.

   Example

   ```
   Device(config-flow-record)# match ipv4 protocol
   ```

   ```
   Device(config-flow-record)# match ipv4 source address
   ```

4. Specify a match to the IPv4 destination address-based field and specify a match to the transport layer's source port field.

   Example

   ```
   Device(config-flow-record)# match ipv4 destination address
   ```

   ```
   Device(config-flow-record)# match transport source-port
   ```

5. Specify a match to the transport layer's destination port field and specify a match to the direction the flow was monitored in.

   Example

   ```
   Device(config-flow-record)# match transport destination-port
   ```

   ```
   Device(config-flow-record)# match flow direction
   ```

6. Specify a match to the application name.

   Example

   ```
   Device(config-flow-record)# match application name
   ```

> **Note**
>
> This action is mandatory for AVC support because this allows the flow to be matched against the application.

7. Specify a match to the SSID name identifying the wireless network and collect the counter field's total bytes.

   Example

   ```
   Device(config-flow-record)# match wireless ssid
   ```

   ```
   Device(config-flow-record)# collect counter bytes long
   ```

8. Collect the counter field's total packets and collect the BSSID with the MAC addresses of the APs that the wireless client is associated with.

   Example

   ```
   Device(config-flow-record)# collect counter packets long
   ```

   ```
   Device(config-flow-record)# collect wireless ap mac address
   ```

9. Collect the MAC address of the client on the wireless network.

   Example

   ```
   Device(config-flow-record)# collect wireless client mac address
   ```

Create a flow exporter (CLI)

Create a flow exporter using commands.

Define export parameters for a flow exporter so AVC statistics are visible on the controller embedded wireless controller. This is an optional procedure for configuring flow exporter parameters.

Note

For the AVC statistics to be visible at the controller, you should configure a local flow exporter using these commands:

- **flow exporter** *my_local*
- **destination local wlc**

Also, your flow monitor must use this local exporter for the statistics to be visible at the controller .

Procedure

1. Create a flow monitor.

   Example

   ```
   Device(config)# flow exporter flow-export-name export-test
   ```

2. Describe the flow record as a maximum 63-character string.

   Example

   ```
   Device(config-flow-exporter)# description IPv4flow
   ```

3. **Example**

   ```
   Device(config-flow-exporter)# destination {hostname/ipv4address |
   hostname/ipv6address | local wlc}
   ```

4. Specify the local controller to which the exporter sends data.

   Example

   ```
   Device(config-flow-exporter)# destination {hostname/ipv4address |
   hostname/ipv6address | local wlc}
   ```

5. Optional: Configure the destination UDP port to reach the external collector.

   Example

   ```
   Device(config-flow-exporter)# transport udp port-value 1024
   ```

   The default value is 9995.

> **Note**
>
> This step is required only for external collectors; not required for local controller collector.

6. Optional: Specify the application table timeout option, in seconds.

   Example

   ```
   Device(config-flow-exporter)# option application-table timeout 500
   ```

   The valid range is from one to 86400.

7. Return to privileged EXEC mode.

   Example

   ```
   Device(config-flow-exporter)# end
   ```

8. Optional: Verify your configuration.

   Example

   ```
   Device# show flow exporter
   ```

Configure a policy tag (CLI)

Configure a policy tag using commands.

Establish a policy tag to control wireless device and SSID behavior.

Policy tags allow you to associate specific policy profiles with APs, enabling customized wireless network behavior.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure policy tag and enter policy tag configuration mode.

   Example

   ```
   Device(config-policy-tag)# wireless tag policy policy-tag-name
   rr-xyz-policy-tag
   ```

3. Save the configuration and exit configuration mode and return to privileged EXEC mode. **end**

Example

```
Device(config-policy-tag)# end
```

Attach a policy profile to a WLAN interface (GUI)

Attach a policy profile to a WLAN interface using GUI.

Link the appropriate policy configuration to a WLAN interface for centralized control. Use the GUI to ensure new or modified policy profiles are correctly associated with WLANs before deployment.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. On the **Manage Tags** page, click the **Policy** tab.
3. Click **Add** to open the **Add Policy Tag** window.
4. Enter a name and description for the policy tag.
5. Click **Add** to map WLAN and policy.
6. Select the WLAN profile to map with the appropriate policy profile, and click the tick icon.
7. Click **Save & Apply to Device**.

Attach a policy profile to a WLAN interface (CLI)

Attach a policy profile to a WLAN interface using commands.

Assign a policy profile to a WLAN interface to control traffic and services for that WLAN.

Before you begin

Procedure

1. Create a policy tag.

   Example

   ```
   Device(config)# wireless tag policy avc-tag
   ```

2. Attach a policy profile to a WLAN profile.

   Example

   ```
   Device(config-policy-tag)# wlan wlan_avc policy avc_pol
   ```

What to do next

- Do not attach different AVC policy profiles to the same WLAN across different policy tags.

This is an example of incorrect configuration:

```
wireless profile policy avc_pol1
ipv4 flow monitor fm-avc1 input
ipv4 flow monitor fm-avc1 outputno shutdown
wireless profile policy avc_pol2
ipv4 flow monitor fm-avc2 input
ipv4 flow monitor fm-avc2 output
no shutdown
wireless tag policy avc-tag1
wlan wlan1 policy avc_pol1
wireless tag policy avc-tag2
wlan wlan1 policy avc_pol2
```

This example violates the restriction stated earlier, that is, the WLAN *wlan1* is mapped to 2 policy profiles, *avc_pol1* and *avc_pol2*. This configuration is incorrect because the WLAN *wlan1* must be mapped to either *avc_pol1* or *avc_pol2* throughout the configuration..

- Conflicting policy profiles on the same WLAN are not supported. For example, policy profile (with and without AVC) applied to the same WLAN in different policy tags.

  This is an example of an incorrect configuration:

```
wireless profile policy avc_pol1
no shutdown
wireless profile policy avc_pol2
ipv4 flow monitor fm-avc2 input
ipv4 flow monitor fm-avc2 output
no shutdown
wireless tag policy avc-tag1
wlan wlan1 policy avc_pol1
wireless tag policy avc-tag2
wlan wlan1 policy avc_pol2
```

  In this example, a policy profile with and without AVC is applied to the same WLAN in different tags.

- Run the **no shutdown** command on the WLAN after completing the configuration.
- If the WLAN is already in **no shutdown** mode, run the **shutdown** command, then run the **no shutdown** command.

Attach a policy profile to an AP (CLI)

Attach a policy profile to an AP using commands.

Assign a specific policy profile to an AP through commands.

Procedure

1. Enter AP configuration mode.

   Example

```
Device(config)# ap ap-ether-mac 34a8.2ec7.4cf0
```

2. Specify the policy tag that is to be attached to the AP.

Example

```
Device(config)# policy-tag avc-tag
```

Verify the AVC configuration (CLI)

Verify the AVC configuration using commands.

Review and validate the Application Visibility and Control (AVC) configuration on a device.

Procedure

1. Display information about the top applications and the users who use them.

   Example

   ```
   Device# show avc wlan wlan_avc top 2 applications {aggregate | downstream | upstream}
   ```

   Note

   To collect accurate statistics, ensure that wireless clients are connected to the WLAN and transmitting data. After 90 seconds, run the command.

2. Display information about the top applications for the client.

   Example

   ```
   Device# show avc client 9.3.4 top 3 applications {aggregate | downstream | upstream}
   ```

   Note

   Ensure that wireless clients are connected to the WLAN and transmitting data. Wait 90 seconds for statistics to become available, then run the command.

3. Display information about the top applications and the users who uses them.

   Example

   ```
   Device# show avc wlan wlan_avc application app top 4 aggregate
   ```

4. Display a summary of all the APs attached to the controller .

Example

```
Device# show ap summary
```

5. Display a summary of all the APs with policy tags.

Example

```
Device# show ap tag summary
```

Default DSCP on AVC

Explains how default DSCP (Differentiated Services Code Point) values are applied within Application Visibility and Control (AVC).

Configuring default DSCP for AVC profile (GUI)

Configure default DSCP for AVC profile using GUI.

Assign a default DSCP value for an Application Visibility and Control (AVC) profile using the GUI.

Procedure

1. Choose **Configuration** > **Services** > **QoS**.
2. Click **Add**.
3. Enter the **Policy Name**.
4. Click **Add Class-Maps**.
5. Select **AVC** from the **AVC/User Defined** drop-down list.
6. Click either **Any** or **All** match type radio button.
7. Select **DSCP** from the **Mark Type** drop-down list.
8. a) Check the **Drop** check box to drop traffic from specific sources.
   b) If you do not want to drop the traffic, enter the **Police(kbps)** and select the match type from the **Match Type** drop-down list. Choose the items from the available list and click move them to the selected list. Click **Save**.
9. Click **Apply to Device**.

Guidelines to configure default DSCP marking for unclassified AVC packets

Configure Default DSCP for AVC Profile.

When using Cisco Catalyst 9800 Series Wireless Controller or , ensure you configure a default DSCP marking for packets not matched by AVC policy filters. This allows consistent QoS treatment, even when your policy exceeds the 32-filter limit or cannot individually classify certain packet types.

- Set a default marking action in your class map and policy map to handle all unclassified traffic.
- Apply this configuration to ensure unclassified packets receive the intended QoS treatment.

Create class map (CLI)

Create class map using commands.

Define a class map to identify and categorize network traffic based on application names, categories, subcategories, or application groups.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a class map.

   Example

   ```
   Device(config-pmap)# class-map avc-class
   ```

3. Specify match to the application name, category name, subcategory name, or application group.

   Example

   ```
   Device(config)# class-map avc-class
   Device(config-cmap)# match protocol avc-media
   Device(config)# class-map class-avc-category
   Device(config-cmap)# match protocol attribute category avc-media

   Device# class-map class-avc-sub-category
   Device(config-cmap)# match protocol attribute sub-category avc-media

   Device# class-map avcS-webex-application-group
   Device(config-cmap)# match protocol attribute application-group webex-media
   ```

4. Return to privileged EXEC mode. Alternatively, press Ctrl-Z to exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

Create policy map (CLI)

Create policy map using commands.

Define and configure a policy map in global configuration mode to classify and act on network traffic.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Create a policy map by entering the policy map name to enter policy-map configuration mode.

Example

```
Device(config)# policy-map avc-policy
```

By default, no policy maps are defined.

When a policy map is used, it sets the DSCP to zero for IP packets and sets the CoS to zero for tagged packets.

> **Note**
>
> To delete an existing policy map, use the no policy-map policy-map-name global configuration command.

3. Define a traffic classification to enter policy-map class configuration mode.

Example

```
Device(config-pmap)# class [class-map-name | class-default]
```

By default, no policy maps or class maps are defined.

If a traffic class has already been defined by using the class-map global configuration command, specify its name as class-map-name in this command.

The class-default traffic class is predefined and can be added to any policy. It is always placed at the end of a policy map. The class-default class includes an implied 'match any,' which ensures that all packets that have not already matched other traffic classes are matched by class-default.

> **Note**
>
> To delete an existing class map, use the no class class-map-name policy-map configuration command.

4. Classify IP traffic by setting a new value in the packet.

Example

```
Device(config-pmap-c)# set dscp new-dscp 45
```

For **dscp new-dscp** , enter a new DSCP value to assign to the classified traffic. The range is zero to 63.

5. Specify the default class so that you can configure or modify its policy.

Example

```
Device(config-pmap)# class class-default
```

6. Configure the default DSCP.

Example

```
Device(config-pmap)# set dscp default
```

7. Return to privileged EXEC mode. Alternatively, press Ctrl-Z to exit global configuration mode.

Example

```
Device(config-pmap)# end
```

AVC-based selective reanchoring

Explains how AVC-based selective reanchorings manage client mobility between wireless controllers based on application traffic and AVC profiles.

An AVC-based selective reanchoring is a wireless client mobility mechanism that

- reanchors clients as they roam between one controller to another and prevent depletion of available IP addresses, and
- uses AVC profile-based statistics to determine if a client should be reanchored or deferred.

This is useful when a client is actively running a voice or video application defined in the AVC rules.

The reanchoring process also involves deauthentication of anchored clients. The clients get deauthenticated when they do not transmit traffic for the applications listed in the AVC rules while roaming between controllers.

Restrictions for AVC-based selective reanchoring

AVC-based selective reanchoring restrictions.

- This feature operates only in local mode. FlexConnect and fabric modes are not supported.
- This feature does not operate in guest tunneling scenarios or export anchor scenarios.
- After reanchoring, the old IP address is retained until the IP address lease period ends.

Configure the flow exporter (CLI)

Configure the flow exporter using commands.

Define and configure a flow exporter to specify where and how NetFlow data is exported from the device.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Create a flow exporter and enter the flow exporter configuration mode.

   Example

   ```
   Device(config)# flow exporter avc-reanchor
   ```

   Note

   You can use this command to modify an existing flow exporter too.

3. Set the exporter as local.

   Example

   ```
   Device(config-flow-exporter)# destination local wlc
   ```

Configure the flow monitor (CLI)

Configure the flow monitor using commands.

Set up a flexible NetFlow flow monitor to track traffic statistics.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a flow monitor and enter the flexible NetFlow flow monitor configuration mode.

   Example

   ```
   Device(config)# flow monitor fm_avc
   ```

   Note

   You can use this command to modify an existing flow monitor too.

3. Specify the name of an exporter.

Example

```
Device(config-flow-monitor)# exporter avc-reanchor
```

4. Specify the flow record to use to define the cache.

Example

```
Device(config-flow-monitor)# record wireless avc basic
```

5. Set the active flow timeout, in seconds.

Example

```
Device(config-flow-monitor)# cache timeout active 60
```

6. Set the inactive flow timeout, in seconds.

Example

```
Device(config-flow-monitor)# cache timeout inactive 60
```

Configure the AVC reanchoring profile (CLI)

Configure the AVC reanchoring profile using commands.

Configure traffic reanchoring for AVC by creating and updating an AVC reanchoring profile using commands.

Before you begin

- Ensure that you use the AVC-Reanchor-Class class map. All other class-map names are ignored by Selective Reanchoring.
- During boot up, the system checks for the existence of the AVC-Reanchor-Class class map. If it is not found, the system creates default protocols, such as jabber-video, WiFi-calling, and others. If the AVC-Reanchor-Class class map is found, configuration changes are not made. Updates to protocols saved to the startup configuration persist across reboots.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the class map.

Example

```
Device(config)# class-map AVC-Reanchor-Class
```

3. Instruct the device to match with any of the protocols that pass through it.

   Example

   ```
   Device(config-cmap)# match any
   ```

4. Specify a match to the application name.

   Example

   ```
   Device(config-cmap)# match protocol jabber-audio
   ```

   You can edit the class-map configuration later to add or remove protocols such as **jabber-video and wifi-calling** if needed.

Configure the wireless WLAN profile policy (CLI)

Configure the wireless WLAN profile policy using commands.

Follow the procedure given below to configure the WLAN profile policy.

> **Note**
>
> Starting with Cisco IOS XE Amsterdam 17.1.1, IPv6 flow monitor is supported on Wave 2 APs. When NBAR runs in the controller and the AP operates in local (central switching) mode, you can attach two flow monitors per direction (input and output) and per IP version (IPv4 and IPv6) in a policy profile. In FlexConnect and fabric modes on Wave 2 APs, when NBAR runs on the AP, only one flow monitor per direction (input and output) and per IP version (IPv4 and IPv6) is supported.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile and enters wireless policy configuration mode and disable the policy profile.

   Example

   ```
   Device(config)# wireless profile policy default-policy-profile
   ```

   ```
   Device(config-wireless-policy)# shutdown
   ```

3. Disable central switching.

Example

```
Device(config-wireless-policy)# no central switching
```

4. Specify the name of the IPv4 ingress flow monitor.

Example

```
Device(config-wireless-policy)# ipv4 flow monitor fm_avc input
```

5. Specify the name of the IPv4 egress flow monitor.

Example

```
Device(config-wireless-policy)# ipv4 flow monitor fm_avc output
```

6. Specify the name of the IPv6 ingress and egress flow monitor.

Example

```
Device(config-wireless-policy)# ipv6 flow monitor fm_v6_avc input
```

```
Device(config-wireless-policy)# ipv6 flow monitor fm_v6_avc output
```

7. Enable the policy profile.

Example

```
Device(config-wireless-policy)# no shutdown
```

Verify AVC reanchoring

Verify AVC reanchoring.

Use the commands to verify the AVC reanchoring configuration:

```
Device# show wireless profile policy detailed avc_reanchor_policy

Policy Profile Name           : avc_reanchor_policy
Description                   :
Status                        : ENABLED
VLAN                          : 1
Wireless management interface VLAN        : 34
!
.
.
.
AVC VISIBILITY                : Enabled
Flow Monitor IPv4
  Flow Monitor Ingress Name   : fm_avc
  Flow Monitor Egress Name    : fm_avc
Flow Monitor IPv6
  Flow Monitor Ingress Name   : Not Configured
  Flow Monitor Egress Name    : Not Configured
```

```
NBAR Protocol Discovery       : Disabled
Reanchoring                   : Enabled
Classmap name for Reanchoring
  Reanchoring Classmap Name   : AVC-Reanchor-Class
!
.
.
.
  ------------------------------------------------------
```

```
Device# show platform software trace counter tag wstatsd chassis active R0
avc-stats debug

Counter Name Thread ID Counter Value
-----------------------------------------------------------------------------
Reanch_deassociated_clients 28340 1
Reanch_tracked_clients 28340 4
Reanch_deleted_clients 28340 3
```

```
Device# show platform software trace counter tag wncd chassis active R0 avc-afc
 debug

Counter Name Thread ID Counter Value
-----------------------------------------------------------------------------
Reanch_co_ignored_clients 30063 1
Reanch_co_anchored_clients 30063 5
Reanch_co_deauthed_clients 30063 4
```

```
Device# show platform software wlavc status wncd
Event history of WNCD DB:

AVC key: [1,wlan_avc,N/A,Reanc,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Reanchoring
Flow-mon-name : N/A
Policy-tag : default-policy-tag
Switching Mode : CENTRAL

Timestamp FSM State Event RC Ctx
------------------------- ------------------- ------------------------- ----
 ----
06/12/2018 16:45:30.630342 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:28.822780 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:28.822672 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:15.172073 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:12.738367 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:12.738261 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:01.162689 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:55.757643 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:55.757542 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:04.468749 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:02.18857 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:02.18717 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:38:20.164304 2 :READY 3 :FSM_AFM_SWEEP 0 2
06/12/2018 16:35:20.163877 2 :READY 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:18.593257 1 :INIT 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:18.593152 1 :INIT 24:CREATE_FSM 0 0
```

```
AVC key: [1,wlan_avc,fm_avc,v4-In,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Ingress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL

Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ----
 ----
06/12/2018 16:45:30.664772 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:28.822499 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:28.822222 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:15.207605 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:12.738105 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:12.737997 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:01.164225 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:55.757266 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:55.757181 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:04.472778 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:02.15413 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:02.15263 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:38:20.164254 2 :READY 3 :FSM_AFM_SWEEP 0 2
06/12/2018 16:35:20.163209 1 :INIT 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:20.163189 1 :INIT 24:CREATE_FSM 0 0

AVC key: [1,wlan_avc,fm_avc,v4-Ou,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Egress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL

Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ----
 ----
06/12/2018 16:45:30.630764 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:28.822621 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:28.822574 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:15.172357 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:45:12.738212 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:12.738167 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:45:01.164048 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:55.757403 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:55.757361 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:04.472561 3 :ZOMBIE 1 :FSM_AFM_BIND 0 2
06/12/2018 16:44:02.18660 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:44:02.18588 2 :READY 2 :FSM_AFM_UNBIND 0 0
06/12/2018 16:38:20.164293 2 :READY 3 :FSM_AFM_SWEEP 0 2
06/12/2018 16:35:20.163799 1 :INIT 1 :FSM_AFM_BIND 0 2
06/12/2018 16:35:20.163773 1 :INIT 24:CREATE_FSM 0 0
```

```
Device# show platform software wlavc status wncmgrd

Event history of WNCMgr DB:

AVC key: [1,wlan_avc,N/A,Reanc,default-policy-tag]
Current state : READY
```

```
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Reanchoring
Flow-mon-name : N/A
Policy-tag : default-policy-tag
Switching Mode : CENTRAL
Policy-profile : AVC_POL_PYATS

Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ----
 ----
06/12/2018 16:45:30.629278 3 :WLAN_READY 24:BIND_WNCD 0 0
06/12/2018 16:45:30.629223 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.629179 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.510867 2 :PLUMB_READY 22:BIND_IOSD 0 0
06/12/2018 16:45:30.510411 2 :PLUMB_READY 2 :FSM_WLAN_UP 0 0
06/12/2018 16:45:30.510371 2 :PLUMB_READY 1 :FSM_WLAN_FM_PLUMB 0 0
06/12/2018 16:45:28.886377 2 :PLUMB_READY 20:UNBIND_ACK_IOSD 0 0
!

AVC key: [1,wlan_avc,fm_avc,v4-In,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Ingress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL
Policy-profile : AVC_POL_PYATS

Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ----
 ----
06/12/2018 16:45:30.664032 3 :WLAN_READY 24:BIND_WNCD 0 0
06/12/2018 16:45:30.663958 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.663921 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.511151 2 :PLUMB_READY 22:BIND_IOSD 0 0
06/12/2018 16:45:30.510624 2 :PLUMB_READY 2 :FSM_WLAN_UP 0 0
06/12/2018 16:45:30.510608 2 :PLUMB_READY 1 :FSM_WLAN_FM_PLUMB 0 0
06/12/2018 16:45:28.810867 2 :PLUMB_READY 20:UNBIND_ACK_IOSD 0 0
06/12/2018 16:45:28.807239 4 :READY 25:UNBIND_WNCD 0 0
06/12/2018 16:45:28.807205 4 :READY 23:UNBIND_IOSD 0 0
06/12/2018 16:45:28.806734 4 :READY 3 :FSM_WLAN_DOWN 0 0
!

AVC key: [1,wlan_avc,fm_avc,v4-Ou,default-policy-tag]
Current state : READY
Wlan-id : 1
Wlan-name : wlan_avc
Feature type : Flow monitor IPv4 Egress
Flow-mon-name : fm_avc
Policy-tag : default-policy-tag
Switching Mode : CENTRAL
Policy-profile : AVC_POL_PYATS

Timestamp FSM State Event RC Ctx
------------------------- ------------------ ------------------------- ----
 ----
06/12/2018 16:45:30.629414 3 :WLAN_READY 24:BIND_WNCD 0 0
06/12/2018 16:45:30.629392 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.629380 3 :WLAN_READY 4 :FSM_BIND_ACK 0 0
06/12/2018 16:45:30.510954 2 :PLUMB_READY 22:BIND_IOSD 0 0
```

```
06/12/2018 16:45:30.510572 2 :PLUMB_READY 2 :FSM_WLAN_UP 0 0
06/12/2018 16:45:30.510532 2 :PLUMB_READY 1 :FSM_WLAN_FM_PLUMB 0 0
06/12/2018 16:45:28.886293 2 :PLUMB_READY 20:UNBIND_ACK_IOSD 0 0
06/12/2018 16:45:28.807844 4 :READY 25:UNBIND_WNCD 0 0
06/12/2018 16:45:28.807795 4 :READY 23:UNBIND_IOSD 0 0
06/12/2018 16:45:28.806990 4 :READY 3 :FSM_WLAN_DOWN 0 0
!
```

81 Software-Defined Application Visibility and Control

Topics:

- Software-defined application visibility and control
- List of package files

Software-defined application visibility and control

Introduces Software-Defined Application Visibility and Control (SD-AVC), detailing its centralized analytics, anomaly detection, per-WLAN configuration, integration with NBAR, platform and release support for IPv4/IPv6, and the associated feature history and commands for enterprise network traffic management.

A software-defined application visibility and control system is a network-level application visibility and control (AVC) controller that

- aggregates application data from multiple devices and sources
- provides composite analytics and telemetry on enterprise network traffic, and
- enables centralized deployment and management of protocol pack updates.

SD-AVC recognizes most enterprise network traffic and provides analytics, visibility, and telemetry into application recognition on the network. SD-AVC profiles all endpoints connected to access nodes, including wireless bridged virtual machines, to perform anomaly detection operations such as Network Address Translation (NAT). SD-AVC alerts you when the same MAC address is used at the same time on different networks.

You can enable the Software-Defined Application Visibility and Control feature on a per-WLAN basis. You can enable or disable Software-Defined Application Visibility and Control functionalities independently.

> **Note**
>
> Restart the Capwapd process or reload the AP to resume SD-AVC operation after the SD-AVC process (stilepd) crashes.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 86: Feature history for Software-defined application visibility and control

Feature Name	Release Information	Feature Description
Software-defined application visibility and control (SD-AVC) wireless support with IPv6	Cisco IOS XE 17.18.1	From Cisco IOS XE 17.18.1 onwards, this feature extends the support for adding IPv6 SD-AVC controller or end-point address. These platforms are supported: • Cisco Catalyst 9800 controllers: 9800-40, 9800-80, 9800-L, 9800-CL, 9800-SW, CW9800M, CW9800H1, and CW9800H2. • Cisco Catalyst 9300/9400 switches in Fabric mode. • Cisco Wave 2, Wi-Fi 6/6E, and Wi-Fi 7 APs. SD-AVC IPv6 is not supported on Cisco Wireless AireOS Controllers, Cisco Embedded Wireless Controller on Catalyst APs, and Cisco Wave 1 APs.
Software-defined application visibility and control (SD-AVC) wireless support with IPv4	Cisco IOS XE 17.17.1	From Cisco IOS XE 17.17.1 onwards, this feature extends the support for adding the AP and controller payload code for only IPv4 SD-AVC addresses.
Software-defined application visibility and control	Cisco IOS XE 17.7.1	Software-Defined AVC aggregates application data from multiple sources and provides composite application information. These commands are introduced: • address and avcsd-service • controller and destination-ports • dscp • segment • source-interface • transport and application-updates • vrf and showsdavc ap download status • showsdavc status ap

Supported platforms for SD-AVC IPv6

This topic lists the platforms that support the SD-AVC IPv6 feature.

- Cisco Catalyst 9800 controllers-9800-40, 9800-80, 9800-L, 9800-CL, and 9800-SW.
- Cisco Catalyst 9300/9400 switches in Fabric mode.
- Cisco Wave 2 and 11AX APs.

Non supported platforms for SD-AVC IPv6

- Cisco Wireless AireOS Controllers.
- Cisco Embedded Wireless Controller on Catalyst APs.
- Cisco Wave 1 APs

Enable software-defined application visibility and control on a WLAN (CLI)

Enable software-defined application visibility and control on a WLAN using commands.

Allow the system to recognize and manage applications on a WLAN by enabling software-defined application visibility and control using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy test-policy-profile
   ```

3. Disable central switching and enable local switching.

   Example

   ```
   Device(config-wireless-policy)# no central switching
   ```

4. Enable application recognition on the wireless policy profile by activating the NBAR2 engine.

   Example

   ```
   Device(config-wireless-policy)# ip nbar protocol-discovery
   ```

5. Exit wireless policy configuration mode and return to the privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

Configure software-defined application visibility and control global parameters (CLI)

Configure software-defined application visibility and control global parameters using commands.

Enable SD-AVC globally and configure connectivity parameters for SD-AVC controllers using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable SD-AVC and enter the software-definition service configuration mode.

   Example

   ```
   Device(config-sd-service)# avc sd-service
   ```

3. Configure a segment name identifying a group of devices sharing the same application services.

   Example

   ```
   Device(config-sd-service)# segment AppRecognition
   ```

4. Enter SD service controller configuration mode to configure connectivity parameters.

   Example

   ```
   Device(config-sd-service)# controller
   ```

5. Configure controller IP address. Supports only IPv4 address.

   Example

   ```
   Device(config-sd-service-controller)# address 209.165.201.0
   ```

6. Configure the destination port for communicating with the controller.

   Example

   ```
   Device(config-sd-service-controller)# destination-ports sensor-exporter 21730
   ```

7. Enable DSCP marking and configure source interface for communicating with the controller.

   Example

   ```
   Device(config-sd-service-controller)# dscp 16
   ```

   ```
   Device(config-sd-service-controller)# source-interface GigabitEthernet21
   ```

8. Configure transport protocols for communicating with the controller and associate the VRF with the source interface.

Example

```
Device(config-sd-service-controller)# transport application-updates https
url-prefix cisco
```

```
Device(config-sd-service-controller)# vrf doc-test
```

9. Exit the SD service controller configuration mode and enter the privileged EXEC mode.

Example

```
Device(config-sd-service-controller)# end
```

List of package files

Displays the list of package files that are received by the controller from the SD-AVC.

To view the list of package files list that are sent from the SD-AVC to the controller, use the following command:

```
Device# dir bootflash:sdavc
Directory of bootflash:/sdavc/

251671  -rw-             6728  Jun 26 2025 03:26:54 +00:00  sdavc_config.json
251674  -rw-              338  Jun 26 2025 03:21:55 +00:00
import_file_meta.json
251670  -rw-            10913  Jun 26 2025 03:20:52 +00:00
PPDK_C9400_HA_00a38d1dddb4fa0bc4b51f91f97744.pack
251669  -rw-             6487  Jun 26 2025 03:20:21 +00:00
pp_update_pp_minor_taxonomy_b72edc9e6ed2e42.json
251672  -rw-              912  Jun 26 2025 03:20:21 +00:00
pp_update_C9400_HA_a_v2_b30473cb9912.pack
348817  drwx             4096  Jun 26 2025 03:16:27 +00:00
container_application

11250098176 bytes total (3345641472 bytes free)
```

Verify SD-AVC controller connection status

This content displays the verification output for the SD-AVC and controller connection status.

To display the SD-AVC connection status and information summary in the controller, run the **show avc sd-service info summary** command.

```
Device# show avc sd-service info summary
Status: CONNECTED
Device ID: SDA_BOX
Device segment name: C9400_HA
Device address: 192.0.2.1
Device OS version: 17.18.01prd15
Device type: C9407R
Active controller:
   Type    : Primary
   Address : 192.0.2.1
   Status  : Connected
   Version : 4.4.0
```

```
   Last connection: 06:21:35.000 UTC Thu Jun 19 2025
Active SDAVC import files:
    Protocol pack:          Not loaded
    Secondary protocol pack: PPDK_C9400_HA_00a38d1dddb4fa0bc4b51f91f97744.pack

    Rules pack:             pp_update_C9400_HA_a_v2_b32d0890d8a3.pack
```

Verify SD-AVC details in AP

Verify SD-AVC package files in flash

To verify the protocol files received from the controller, run the **show flash sdavc-pkg** command in the AP.

```
Cisco-AP# show flash sdavc-pkg
Files under /usr/bin/sdavc/
total 36K
drwxr-xr-x    2 Cisco    root           40 Jun 19 06:10 container_application
-rw-r--r--    1 Cisco    root         6487 Jun 19 06:11
pp_update_pp_minor_taxonomy_b72edc9e6ed2e42.json
-rw-r--r--    1 Cisco    root        10922 Jun 19 06:11
PPDK_C9400_HA_00a38d1dddb4fa0bc4b51f91f97744.pack
-rw-r--r--    1 Cisco    root          959 Jun 19 06:11
pp_update_C9400_HA_a_v2_b32d0890d8a3.pack
-rw-r--r--    1 Cisco    root          314 Jun 19 06:11 import_file_meta.json
-rw-r--r--    1 Cisco    root         6728 Jun 19 06:20 sdavc_config.json
-------------------------------------------------------------------------------
Filesystem                Size      Used Available Use% Mounted on
/dev/mmcblk0p32           2.8G     12.7M      2.6G   0% /storage
```

Verify SD-AVC configurations from controller

To verify the SD-AVC configurations sent from the controller, run the **show ap fast-path configuration sd-avc** command in the AP.

```
Cisco-AP# show ap fast-path configuration sd-avc
  source IP address     : 0.0.0.0
  source port (WLC)     : 9999
  destination IPV4 address: 9.2.39.36
  destination port (SD-AVC): 50000
  gateway MAC           : 00:0C:29:3C:00:8D
  ewlc SD-AVC MAC       : 00:00:00:05:00:01
  DSCP                  : 0
  Fast Path Admin State : enabled
  AP SD-AVC Status      : enabled
```

Verify SD-AVC enablement in AP

To verify the SD-AVC enablement in the AP, run the **show avc status** command.

```
Cisco-AP# show avc status
VAP FNF-STATUS AVC-QOS-STATUS SD AVC-STATUS APM-STATUS
  0   Disabled       Disabled       Enabled   Disabled
  1   Disabled       Disabled       Enabled   Disabled
  2   Disabled       Disabled       Enabled   Disabled
  3   Disabled       Disabled      Disabled   Disabled
  4   Disabled       Disabled      Disabled   Disabled
  5   Disabled       Disabled      Disabled   Disabled
  6   Disabled       Disabled      Disabled   Disabled
  7   Disabled       Disabled      Disabled   Disabled
```

```
 8   Disabled       Disabled      Disabled   Disabled
 9   Disabled       Disabled      Disabled   Disabled
10   Disabled       Disabled      Disabled   Disabled
11   Disabled       Disabled      Disabled   Disabled
12   Disabled       Disabled      Disabled   Disabled
13   Disabled       Disabled      Disabled   Disabled
14   Disabled       Disabled      Disabled   Disabled
15   Disabled       Disabled      Disabled   Disabled
```

82 Cisco Hyperlocation

Topics:

- Cisco Hyperlocation
- NTPS and BLE configuration

Cisco Hyperlocation

Explains how Cisco Hyperlocation solutions use specialized radio modules and algorithms to achieve precise real-time locating of Wi-Fi and BLE devices, supporting advanced asset tracking and targeted content delivery in wireless environments.

A hyperlocation solution is a wireless location technology that

- uses specialized radio modules and algorithms to pinpoint the real-time position of Wi-Fi and BLE devices
- enables advanced asset tracking and targeted content delivery by achieving sub-meter location accuracy, and
- integrates with wireless network infrastructure and management platforms (such as controller and CMX)

You can use the Cisco Hyperlocation radio module in Cisco Aironet 3600, 3700, and 4800 Series APs to enable hyperlocation.

Hyperlocation is also supported in Fabric mode. In particular, when the wireless controller runs on the switch, the controller provisions the APs. The APs generate Hyperlocation VxLAN packets that can travel across the fabric network. The network infrastructure delivers these packets to the CMX destination.

The Hyperlocation VxLAN packets are special packets marked with SGT 0 and use the L3VNID of the APs. For more information, refer to the SDA documentation.

The Cisco Hyperlocation radio module provides these functions:

- WSM or WSM2 radio module functions extended to support:
  - 802.11ac
  - Wi-Fi Transmit
  - 20-MHz, 40-MHz, and 80-MHz channel bandwidth.
- Expanded location functionality:
  - Low-latency location optimized channel scanning
  - 32-antenna angle of arrival (AoA); available only with the WSM2 module.

Using the controller, you can configure Cisco Hyperlocation for APs based on their profile.

Note

When you use the WSM2 module (which includes the WSM module and the antenna add-on), the accuracy of tracking the location of wireless clients can be as close as one meter.

Cisco Hyperlocation works with Cisco Connected Mobile Experiences (CMX). When you combine the Cisco Hyperlocation feature on a Cisco Catalyst 9800 Series Wireless Controller with a CMX device, you can achieve better location accuracy, enabling more targeted content delivery to users. With CMX and Cisco CleanAir frequency scanning, it is easy to locate failed, lost, and rogue beacons.

The Cisco Hyperlocation radio module with an integrated BLE radio allows transmission of Bluetooth Low Energy (BLE) broadcast messages by using up to five BLE transmitters. The Cisco Catalyst 9800 Series Wireless Controller configures transmission parameters, such as the beacon interval, universally unique identifier (UUID), and transmission power for each beacon, globally for all APs. You must enable the Cisco Hyperlocation feature on the controller. The controller configures the major, minor, and transmission power values for each AP to provide more beacon granularity.

> **Note**
>
> The Cisco Hyperlocation feature must be enabled on the controller. In addition, CMX must be connected for BLE to work.

If a Cisco Hyperlocation radio module is not present, you can use *Hyperlocation Local Mode*. This option provides location accuracy between five and seven meters by using CPU cycle stealing.

Restrictions on Cisco Hyperlocation

Cisco Hyperlocation restrictions.

- You cannot modify detection, trigger, or reset thresholds while Hyperlocation is enabled.
- You can set the reset threshold to any value from zero up to one less than the current threshold. For example, if the reset threshold is 10, set it to any value from 0 to 9.
- Ensure that CMX is reachable through an SVI interface (VLAN) when you run Cisco Hyperlocation on the Cisco Catalyst 9800 Series Wireless Controller in a non-Fabric deployment. If CMX is reachable only through an L3 port, an error occurs.
- In Fabric deployments, do not place the wireless management interface (typically a loopback interface) in Fabric.
- You cannot assign the wireless management interface to a loopback interface in non-Fabric deployments.

Configure Cisco Hyperlocation (GUI)

Configure Cisco Hyperlocation using GUI.

Enable and configure Cisco Hyperlocation to achieve high-accuracy wireless client location tracking.

Cisco Hyperlocation is a location solution that allows you to track the location of wireless clients with an accuracy of one meter. When you select this option, all other fields on the screen are disabled except for NTP Server.

Procedure

1. Choose **Configuration > Tags & Profiles > AP Join**. Click **Add**.

   The **Add AP Join Profile** dialog box appears.
2. Under the **AP** > **Hyperlocation** tab, check the **Enable Hyperlocation** check box.
3. In the **Detection Threshold (dBm)** field, enter a value to filter out packets with low RSSI. The valid value range is between -100 dBm and -50 dBm.
4. In the **Trigger Threshold (cycles)** field, enter a value to set the number of scan cycles before sending a BAR to clients. The valid value range is between 0 and 99.
5. In the **Reset Threshold is required** field, enter a value to reset value in scan cycles after trigger. The valid value range is between 0 and 99.
6. Click **Save & Apply to Device**.

Configure Cisco Hyperlocation (CLI)

Configure Cisco Hyperlocation using commands.

Enable and customize Cisco Hyperlocation features on supported APs using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

3. Enable Cisco Hyperlocation feature on all the supported APs that are associated with this AP profile.

   Example

   ```
   Device(config-ap-profile)# hyperlocation
   ```

   Use the **no** form of the command to disable the Cisco Hyperlocation feature.

4. Set threshold to filter out packets with low RSSI.

   Example

   ```
   Device(config-ap-profile)# hyperlocation threshold detection -100
   ```

   The **no** form of this command resets the threshold to its default value. The valid value range is between -100 and -50.

5. Reset the value of scan cycles after a trigger.

   Example

   ```
   Device(config-ap-profile)# hyperlocation threshold reset 8
   ```

   The **no** form of this command resets the threshold to its default value.

6. Set the number of scan cycles before sending a block acknowledgment request (BAR) to clients.

   Example

   ```
   Device(config-ap-profile)# hyperlocation threshold trigger 10
   ```

   The **no** form of this command resets the threshold to its default value.

7. Set the IP address of the NTP server.

   Example

   ```
   Device(config-ap-profile)# ntp ip 9.0.0.4
   ```

The **no** form of this command removes the NTP server.

Verify Cisco Hyperlocation

Verify Cisco Hyperlocation.

To display the hyperlocation status values and parameters for all the AP profiles, use this command:

```
Device# show ap hyperlocation summary

Profile Name: custom-profile
Hyperlocation operational status: Down
Reason: Hyperlocation is administratively disabled
Hyperlocation NTP server: 209.165.200.224
Hyperlocation admin status: Disabled
Hyperlocation detection threshold (dBm): -100
Hyperlocation trigger threshold: 10
Hyperlocation reset threshold: 8
Profile Name: default-ap-profile
Hyperlocation operational status: Up
Reason: N/A
Hyperlocation NTP server: 209.165.200.224
Hyperlocation admin status: Enabled
Hyperlocation detection threshold (dBm): -90
Hyperlocation trigger threshold: 22
Hyperlocation reset threshold: 8
```

To display both the overall and the per-AP configuration values and operational status, use this command:

```
Device# show ap hyperlocation detail

Profile Name: house24
Hyperlocation operational status: Up
 Reason: NTP server is not properly configured
Hyperlocation NTP server: 198.51.100.1
Hyperlocation admin status: Enabled
Hyperlocation detection threshold (dBm): -90
Hyperlocation trigger threshold: 8
Hyperlocation reset threshold: 7
AP Name                 Radio MAC         Method      CMX IP            AP
Profile
-----------------------------------------------------------------------------------------
APe865.49d9.bfe0        e865.49ea.a4b0    WSM2+Ant    198.51.100.2
house24
APa89d.21b9.69d0        a89d.21b9.69d0    Local       198.51.100.3
house24
APe4aa.5d3f.d750        e4aa.5d5f.3630    WSM         198.51.100.4
house24
```

To display the overall (profile specific) configuration values and operational status for a given profile, use this command:

```
Device# show ap profile profile-name hyperlocation summary

Profile Name:
               profile-name
Hyperlocation operational status: Up
   Reason: N/A
Hyperlocation NTP server: 209.165.200.224
Hyperlocation admin status: Enabled
```

```
Hyperlocation detection threshold (dBm): -100
Hyperlocation trigger threshold: 10
Hyperlocation reset threshold: 8
```

To display both the overall (profile specific) and per-AP configuration values and operational status for a given profile, use this command. The APs listed are only those APs that belong to the specified join profile.

```
Device# show ap profile profile-name hyperlocation detail

Profile Name:
                profile-name
Hyperlocation operational status: Up
 Reason: N/A
Hyperlocation NTP server: 209.165.200.224
Hyperlocation admin status: Enabled
Hyperlocation detection threshold (dBm): -90
Hyperlocation trigger threshold: 8
Hyperlocation reset threshold: 7
AP Name                 Radio MAC         Method      CMX IP
-----------------------------------------------------------------
APf07f.0635.2d40        f07f.0635.2d40    WSM2+Ant    198.51.100.2
APf07f.0635.2d41        f07f.0635.2d41    Local       198.51.100.3
APf07f.0635.2d42        f07f.0635.2d42    WSM         198.51.100.4
```

To display configuration values for an AP profile, use this command:

```
Device# show ap profile profile-name detailed

Hyperlocation :
  Admin State             : ENABLED
  PAK RSSI Threshold Detection: -100
  PAK RSSI Threshold Trigger  : 10
  PAK RSSI Threshold Reset    : 8
.
.
.
```

To display the Cisco CMXs that are correctly joined and used by hyperlocation, use this command:

```
Device# show ap hyperlocation cmx summary

Hyperlocation-enabled CMXs
IP            Port Dest MAC        Egress src MAC Egress VLAN Ingress src MAC
 Join time
-------------------------------------------------------------------------------------------
198.51.100.4  2003 aaaa.bbbb.cccc  aabb.ccdd.eeff 2           0000.0001.0001
 12/14/18 09:27:14
```

To display the hyperlocation client statistics, use this command:

```
Device# show platform hardware chassis active qfp feature wireless wlclient
cpp-client summary

Client Type Abbreviations:
  RG - REGULAR BL - BLE
  HL - HALO    LI - LWFL INT
Auth State Abbreviations:
  UK - UNKNOWN IP - LEARN IP IV - INVALID
  L3 - L3 AUTH RN - RUN
Mobility State Abbreviations:
  UK - UNKNOWN IN - INIT
```

```
  LC - LOCAL   AN - ANCHOR
  FR - FOREIGN MT - MTE
  IV - INVALID
EoGRE Abbreviations:
  N - NON EOGRE Y - EOGRE
 CPP IF_H     DPIDX      MAC Address    VLAN  CT  MCVL  AS  MS  E  WLAN POA
------------------------------------------------------------------------------

   0X32     0XF0000001  0000.0001.0001   9    HL   0    RN  LC  N       NULL
```

To display the interface handle value statistics, use this command:

```
Device# show platform hardware chassis active qfp feature wireless wlclient
datapath cpp-if-handle 0x32 statistics start
```

To display the recorded flow, use this command:

```
Device# show platform hardware chassis active qfp feature wireless wlclient
datapath cpp-if-handle 0X32 statistics

                    Pkts                Bytes
Rx                   26                 3628
```

To stop statistics capture, use this command:

```
Device# show platform hardware chassis active qfp feature wireless wlclient
datapath cpp-if-handle 0x32 statistics stop
```

To display the APs requested by Cisco CMX with AP group support, use this command:

```
Device# show nmsp subscription group summary

CMX IP address: 198.51.100.4
  Groups subscribed by this CMX server:
  Group name: CMX_1198.51.100.4
Device# show nmsp subscription group detail ap-list CMX_198.51.100.1
198.51.100.1

CMX IP address: 198.51.100.1
CMX Group name: CMX_198.51.100.1
CMX Group AP MACs:
: aa:bb:cc:dd:ee:01 aa:bb:cc:dd:ee:02 aa:bb:cc:dd:ee:03 aa:bb:cc:dd:ee:03
```

To display the NTP IP address and authentication parameters, use this command:

```
Device# show ap profile profile-name detailed

.
.
.
NTP Authentication : ENABLED
Key id : 2
Key type : SHA1
Key format : HEX
Key : 3a2275c74c250c362ca63e4af06fa3f3cd8d4aec
Encryption type : Clear
.
.
.
```

To display the NTP status for each AP, use this command:

```
Device# show ap name AP-G1-230 ntp status

ap-name       enabled v4/v6 IPAddress       Status     Stratum LastSync
SyncOffset
AP-G1-230     Y       v4    198.51.100.5    AuthFail   4       1000      100
```

To display NTP status for all the APs, use this command:

```
Device# show ap ntp status

ap-name       enabled v4/v6 IPAddress  Status  Stratum LastSync  SyncOffset
AP-G1-230         Y      v4   5.5.5.5  AuthFail 2        Never
AP-G1-231         Y      v4   5.5.5.10  Synced  3        1000       100
AP-G1-232         Y      v4   5.5.5.15  Synced  16       2000        50
```

To display the instant status of NTP synchronization for an AP, use this command. The output is from an AP and not from the controller.

```
Device# show ntp

!This error message is displayed when NTP is not configured.
%Error: ntpd is not running
!The following output is displayed when NTP is configured.
Stratum  Version  Last Received    Delay     Offset    Jitter   NTP server
13       4        7sec ago         1.124ms   0.536ms   0.001ms  198.51.100.5
```

To display AP timezone information, use this command:

```
Device# show ap timezone

AP Name   Status   Offsets(h/m)
-------------------------
AP1      Disabled    0:0
AP2      Enabled     1:0
```

NTPS and BLE configuration

Explains how NTP servers provide accurate time synchronization for network devices and ensure the operation of Cisco Hyperlocation services and describes how BLE configurations enable flexible beacon management for access points using per-AP profile and per-AP settings.

A network time protocol server is a network device that

- provides authoritative time information to other devices on the network, and
- enables accurate time synchronization across distributed systems and applications.

For Cisco Hyperlocation to work, synchronize the AP with accurate time. The controller sends NTP information to the AP to achieve this synchronization. The AP then uses the NTP server to synchronize its clock. Therefore, the AP needs connectivity to the NTP server.

APs can be geographically dispersed. Therefore, it is necessary to provide different NTP servers to different APs. Configure NTP server information for each AP profile. If you do not configure NTP information on the AP profile, the controller uses one of the global NTP peers in its configuration. If the controller acts as an NTP server, it uses its management IP address for synchronization. If the NTP server is unavailable, Cisco Hyperlocation is disabled.

> **Note**
>
> In scale setup, the NTP server should be configured on the respective AP profiles, so that the APs and CA servers used for LSC provisioning are time synchronized. If the NTP server is not configured, a few APs would fail in LSC provisioning.

Bluetooth Low Energy Configuration

The BLE configuration is split into two parts: per-AP profile and per AP. You can configure the BLE feature partially from the AP profile (by default, the AP profile BLE configuration is applied) and partially per AP (some or all the attributes are applied).

Table 87: BLE Configuration Details

Attribute	BLE Configuration Per AP Profile	BLE Configuration Per AP
Attributes with per-AP granularity (global for all the beacons)	• Interval • Advertised transmission power	• Interval • Advertised transmission power
Attributes with per-AP per0-beacon granularity	• Transmission power • UUID • Status	• Transmission power • UUID • Status • Major • Minor

> **Note**
>
> The *default-ap-profile* BLE configuration serves as the default BLE configuration because all APs join the *default-ap-profile* AP profile if the other profiles are removed.
>
> For more information about Cisco Hyperlocation, see the following documents:
>
> - Cisco Hyperlocation Solution
> - Cisco CMX Configuration Guide to enable Cisco Hyperlocation
> - Cisco CMX Release Notes

Best practices for NTP configuration

Follow these best practices when configuring NTP :

- Configure multiple redundant NTP time sources to avoid reliance on a single server and improve reliability.
- Use a loopback interface as the NTP source for consistency across device reboots and interface changes.
- Enable NTP authentication to secure communications and prevent malicious time alterations.

- Monitor NTP status and logs regularly; implement detection and alerting mechanisms using commands like **show ntp status** and **show ntp associations**.
- Use the **ntp panic-threshold-apply** command to check the configuration sanity checks, which reject large clock changes and prevent network instability.
- Address failure modes related to time-based authentication keychains by configuring overlapping key lifetimes to avoid routing disruptions.
- Regularly monitor and validate NTP synchronization status with monitoring tools and SNMP traps.
- Ensure operational practices include multiple NTP sources and configuration checks to prevent routing disruptions caused by time inaccuracies.

These practices help maintain accurate time synchronization, improve NTP security, and minimize the risk of network instability caused by incorrect time settings.

AP group NTP server

AP group NTP server details.

An AP group NTP server is a time synchronization mechanism that

- ensures precise time alignment across all APs within an AP group to achieve location accuracy
- reduces synchronization delays compared to using a WAN-based global NTP server, and
- supports features that rely on accurate timing, such as Cisco Hyperlocation, BLE AoA, and Intelligent Capture.

If your controller and global NTP server are configured on the WAN, the APs may experience significant synchronization delays. These delays can reduce location accuracy.

When all APs in a group synchronize with the same NTP server, you can obtain accurate data for location calculations. Configure the NTP server locally for all APs in an AP group to improve synchronization among the APs.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 88: Feature history for AP group NTP server

Feature Name	Release Information	Feature Description
AP group NTP server	Cisco IOS XE 17.6.1	From this release, the global NTP server configuration is replaced with per-AP-group NTP server configuration. Per-group NTP is now required for the Cisco Hyperlocation feature. These commands are introduced: • ntpauth-key • timezonedelta • timezoneuse-controller • showap name ntp status • showap ntp status • showap timezone

Configure an AP group NTP server (CLI)

Configure an AP group NTP server using commands.

Set a Network Time Protocol (NTP) server for AP groups to ensure time synchronization.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

3. Set the IP address of the NTP server.

   Example

   ```
   Device(config-ap-profile)# [no] ntp ip 9.0.0.4
   ```

   The **no** form of this command removes the NTP server.

4. Configure the NTP server per AP profile to support authentication.

Example

```
Device(config-ap-profile)# [no] ntp auth-key index type type format format
key encryption-type server-key
```

The **no ntp auth-key** command removes the NTP server from each AP profile.

> **Note**
>
> For an ASCII key, ensure that the length is fewer than 21 bytes. For a HEX key, the length must be fewer than 41 bytes and can use only numbers from 0-9 and characters a-f.

Configure AP timezone (CLI)

Configure AP Timezone using commands.

Set the correct timezone for APs to ensure accurate time-stamping and scheduling using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile test
   ```

3. Configure the timezone offset for AP.

   Example

   ```
   Device(config-ap-profile)# timezone {use-controller | delta hour -12 | minute
    2}
   ```

   You can configure the AP timezone for each AP profile only. It is not possible to configure the timezone for individual APs.

   To configure the timezone, you can either apply the current controller timezone or specify the time difference. By default, timezone is disabled.

BLE concurrent scanning and beaconing

BLE concurrent scanning and beaconing details.

A BLE concurrent scan and beacon feature is a wireless AP capability that

- enables the BLE radio on an AP to scan for device signals and transmit beacons simultaneously
- allows the radio to temporarily pause and return to scanning while sending beacon transmissions, and
- supports seamless switching between scanning and beaconing operations to maximize BLE functionality.

To enable BLE concurrent scanning and beaconing, use Cisco Spaces and select **Dual** mode under **Device Management > AP Beacon > Settings**.

> **Note**
>
> If you use v-IBeacon, BLE concurrent scanning and beaconing is not supported. Supported advertisement profiles are iBeacon, Eddystone-URL, and Eddystone-UID.

Supported APs

This feature is supported only on:

- Cisco Catalyst 9105AX APs.
- Cisco Catalyst 9120AX APs.
- Cisco Catalyst 9124AX APs.
- Cisco Catalyst 9130AX APs.
- Cisco Catalyst 9136AX APs.
- Cisco Catalyst 916X APs.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 89: Feature history for BLE concurrent scanning and beaconing

Feature Name	Release Information	Feature Description
BLE concurrent scanning and beaconing	Cisco IOS XE 17.9.1	From this release, BLE concurrent scanning and beaconing is supported on Cisco Catalyst Wi-Fi 6 APs in basic mode or Cisco IOx mode. The BLE radio on an AP can stop a scan for beacon transmission, and return to the scan after completing the beacon transmission.

Verify BLE concurrent scanning and beaconing

Verify BLE concurrent scanning and beaconing.

Use these commands to verify the status of concurrent scanning and beaconing.

To display the advertisement profile, scan counters, advertisement counters, and other information, enter this command from the AP console:

```
Device# show controllers IOTRadio ble 0 interface
Active BLE host interface       : /dev/ttyiot0
BLE Radio Status                : Powered on
Device Status                   : Open
Device Mode                     : Native
Device resets                   : 4
Last Reset Reason               : Watchdog Timer Reset
Heart beat status               : On
                Scan Status : Enabled Active Transmit Profile : Eddystone URL
BLE MAC                         : 80:6F:B0:31:EF:74
Transmitted advertisement count
 since BLE enabled              : 0
Total scan records received     : 3617
```

To determine whether the transmit profile configuration was successfully pushed from Cisco Spaces to the AP, enter this command from the AP console:

```
Device# show controllers iOTRadio ble 0 broadcast
BLE Profile Config
------------------
                Active profile : Eddystone UID
Profile 0 (iBeacon)
UUID                      : 00000001023012120312032130012111
Interval (ms)             : 100
Power (dBm)               : 0
Advertised Power (dBm)    : -45
Minor                     : 29219
Major                     : 35826
TxPower byte              :
Profile 1 (Eddystone UID)
Namespace (hex)           : 00000000000000000999
Instance-ID (hex)         : 000000000555
Profile 2 (Eddystone URL)
URL                       :
Profile 3 (v-iBeacon)
v-iBeacon status          : Disabled
Chirping interval (ms)    : 100
Profile 4 (Custom Profile)
Adv Data                  :
0000000000000000000000000000000000000000000000000000000000000000
Scan Data                 :
0000000000000000000000000000000000000000000000000000000000000000
Simulator mode            : Disabled
Beacon-ID              Mac                              UUID Major Minor Status

        1 10:F9:20:FE:1D:8F 00000000000000000000000000000000     0     0
 0
        2 10:F9:20:FE:1D:8E 00000000000000000000000000000000     0     0
 0
        3 10:F9:20:FE:1D:8D 00000000000000000000000000000000     0     0
 0
        4 10:F9:20:FE:1D:8C 00000000000000000000000000000000     0     0
 0
        5 10:F9:20:FE:1D:8B 00000000000000000000000000000000     0     0
 0
Beacon-ID Transmit power(dBm) Advertised power(dBm)
        1                -21                  -256
```

```
        2                   -21                   -256
        3                   -21                   -256
        4                   -21                   -256
        5                   -21                   -256
```

To view the list of joined APs that support the BLE management feature, along with BLE details for each AP, enter this command from the controller:

```
Device# show ap ble summary
AP Name AP Model   AP Ethernet MAC BLE       BLE    BLE mode      BLE MAC
     BLE Profile     BLE Scan
                                   Interface Admin
                                   State     State
-------------------------------------------------------------------------------
AP794   C9130AXI-B 04eb.409e.xxxx  Open     Up    Base (Native) 806f.b031.xxxx
  Eddystone URL   Enabled
AP50    C9130AXI-B 04eb.409e.xxxx  Close     Down   Base (Native) Unknown
     Not Configured  Disabled
AP28    C9136I     687d.b45c.xxxx  Close     Down   Base (Native) Unknown
     Not Configured  Disabled
```

83 FastLocate for Cisco Catalyst Series Access Points

Topics:

- FastLocate

FastLocate

Introduces FastLocate technology, detailing deployment restrictions, compatible access points, core network components and functions, step-by-step GUI-based configuration instructions, and verification procedures for effective location tracking with Cisco access points.

A FastLocate solution is a Wi-Fi location technology that

- collects received signal strength indication (RSSI) data or location data from data packets rather than probe requests
- increases the frequency of location updates, and
- improves the effectiveness of location-based services (LBS).

Feature history

Feature Name	Release	Description
FastLocate for Cisco Catalyst Series Access Points	Cisco IOS XE 17.1.1s	This feature enables higher location refresh rates by collecting RSSI or location information from data packets that APs receive. The network can then initiate location-based services (LBS) updates more frequently, leading to more accurate and timely device location tracking for Cisco Catalyst Series APs.

Traditional location tracking versus FastLocate

Traditional Wi-Fi location tracking relies on probe request messages from mobile devices. These probe requests are detected by multiple APs on different channels, and the RSSI or estimated location is sent to the APs to estimate the device's position.

However, Wi-Fi clients are increasingly probing less frequently to conserve battery life. Factors such as client type, operating system, driver, battery level, and current client activity influence how often a device sends probe requests, with intervals ranging from 10 seconds to five minutes. This inconsistency in probing leads to fewer data points and can affect the precision of real-world movement tracking.

FastLocate overcomes this limitation by using the more frequent data packets—rather than relying solely on probe requests—to collect RSSI and location data. These packets are aggregated by APs, enabling the network to initiate LBS updates more frequently and deliver higher location refresh rates.

Restriction for fabric deployments

Do not configure the Wireless Management Interface (WMI) as a loopback (Layer 3) interface in fabric deployments. Always use a VLAN interface for WMI to enable FastLocate functionality.

Supported access points

Supported Access Points for FastLocate.

To view all supported access points, see the AP Feature Matrix

Cisco Aironet 4800 Series also supports Hyperlocation using Angle of Arrival location calculation.

When you enable FastLocate on a supported AP

- the Cisco RF Application-specific integrated circuit (ASIC) radios act as a Wireless Services Software Interface (WSSI) module. The radios then switch to monitoring mode and to off-channel scanning mode.
- radios scan all 2.4-GHz and 5-GHz channels sequentially, spending 150 milliseconds (also called dwell time) on each channel, and
- RF ASIC radios synchronize with the NTP server. After each off-channel dwell period, FastPath sends all collected RSSI records in a specific packet format to the controller.

FastLocate network components and functions

Certain network components are essential for successful packet RSSI location computation using FastLocate.

Network Components	Functionality
Wireless clients	send data, management, and control packets.
Cisco Catalyst 9800 Series Wireless Controller	• configures NTP server data and location parameters on AP. • forwards RSSI information to Cisco CMX or Mobility Services Engine through FastPath or datapath.
Cisco Catalyst 9120 Series AP	• operates location radio in monitor or equivalent role. • synchronizes time with NTP server. • collects RSSI data from all clients (both associated and unassociated). • forwards this RSSI data to the controller through CAPWAP.
Cisco CMX	• parses Fastpath location data received from the controller. • computes client physical locations. • renders this position on the GUI using location algorithms.

Configure FastLocate (GUI)

Configure FastLocate using GUI.

Configure the AP join profile to enable FastLocate on your APs.

Use this task to activate FastLocate location services for APs managed by the controller.

Before you begin

Identify the AP join profile you wish to update.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. On the **AP Join** window, click the **default-ap-profile** AP join profile.

3. In the **Edit AP Join Profile** window, click **AP**.
4. Under **Hyperlocation**, check the **Enable Hyperlocation**.
5. Click **Update & Apply to Device**.

Your selected AP join profile now has FastLocate (Hyperlocation) enabled and is applied to the device.

What to do next

Verify the configuration status and confirm location services are operating as expected.

Verify FastLocate

You can verify FastLocate on the controller and on APs.

Verify FastLocate on APs

```
Device# show ntp

Stratum   Version   Last Received   Delay    Offset    Jitter    NTP server
   1         4         123sec ago   1.169ms  -3.262ms  10.050ms    10.7.7.2
```

```
Device# show ap fast-path statistics

total packets sent   : 90001
invalid app ID drops : 0
application                  : 0 (HALO)
packets sent (CAPWAP)        : 90001
packets sent (APP HOST INTF) : 0
admin state drops            : 0
no dest IP drops             : 0
```

View FastLocate admin status details on the AP.

```
Device# show capwap client rcb

Hyperlocation Admin State    : Enabled
MSE Gateway MAC              : 00:50:56:86:0F:9D
WLC Hyperlocation Source Port: 9999
MSE IP Address               : 10.0.0.1
```

View FastPath related parameters on the AP such as source and destination IP addresses, port numbers, and the gateway MAC address.

```
Device# show ap fast-path configuration hyperlocation

source IP address     : 10.0.0.2
destination IP address: 10.0.0.1
source port (WLC)     : 9999
destination port (MSE): 2003
gateway MAC           : 00:50:56:86:0F:9D
ewlc hyperlocation MAC: 00:00:00:01:00:01
```

Verify FastLocate on Controller

View the summary of applications that send Fastpath or datapath data. The summary displays the hex code for the HyperLocation and BLE port numbers.

```
Device# show platform hardware chassis active qfp feature wireless wlclient cpp-client summary


Client Type Abbreviations:
RG - REGULAR     BL - BLE
HL - HALO        LI - LWFL INT
Auth State Abbreviations:
UK - UNKNOWN     IP - LEARN IP    IV - INVALID
L3 - L3 AUTH     RN - RUN
Mobility State Abbreviations:
UK - UNKNOWN     IN - INIT
LC - LOCAL       AN - ANCHOR
FR - FOREIGN     MT - MTE
IV - INVALID
EoGRE Abbreviations:
N - NON EOGRE    Y - EOGRE
CPP IF_H    DPIDX        MAC Address VLAN CT MCVL AS MS E WLAN POA
------------------------------------------------------------------------------
0X31   0XF0000002 0000.0003.0001 122 BL 0 RN LC N NULL 0X32 0XF0000001
0000.0001.0001 122 HL 0 RN LC N NULL
```

Capture statistics of a selected application.

```
Device# show platform hardware chassis active qfp feature wireless wlclient datapath
cpp-if-handle register-code statistics start
```

The hex value of the register-code is obtained from the **show platform hardware chassis active qfp feature wireless wlclient cpp-client summary** command.

```
Device# show platform hardware chassis active qfp feature wireless wlclient datapath cpp-if-handle 0x32
 statistics start
```

Display the statistics for the specified application.

```
Device# show platform hardware chassis active qfp feature wireless wlclient datapath
cpp-if-handle register-code statistics
```

The hex value of the register code is obtained from the **show platform hardware chassis active qfp feature wireless wlclient cpp-client summary** command.

```
Device# show platform hardware chassis active qfp feature wireless wlclient datapath cpp-if-handle 0x32
 statistics
    Pkts   Bytes
Rx   232   38850
```

84 IoT Services Management

Topics:

- IoT services management

IoT services management

Explains how IoT services management automates device integration, connectivity, and data analytics in enterprise environments.

An IoT service is a network-enabled function that

- automates the integration and management of IoT devices within enterprise environments
- provides connectivity and control for a wide range of sensor and actuator devices, and
- supports real-time data collection and analytics for operational optimization.

Cisco Catalyst 9800 devices running the Cisco IOS XE image version 17.3.2 support Cisco Spaces: IoT Services along with the Network Assurance on Cisco Catalyst Center. IoT Services and Intelligent Capture (iCAP) port configuration are mutually exclusive.

If the iCAP feature is enabled on a device, IoT Services cannot be deployed. Conversely, if IoT Services is enabled, the iCAP feature cannot be deployed.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 90: Feature history for IoT services management

Feature Name	Release Information	Feature Description
BLE management on Cisco Catalyst 9136 Series APs	Cisco IOS XE 17.8.1	This feature allows Bluetooth Low Energy (BLE) management support in Cisco Catalyst 9136 Series APs.
Coexistence of intelligent capture (iCAP) or IoT Services - Dual gRPC channel	Cisco IOS XE 17.7.1	From Cisco IOS XE Cupertino 17.7.1, IoT services and iCAP port configurations may co-exist, creating two gRPC connections from the AP: one to Cisco Catalyst Center for iCAP, and one to Cisco Spaces Connector for IoT.
Assurance and IoT services coexistence without iCAP	Cisco IOS XE 17.3.2	From this release onwards, the controller supports deployment of both Cisco Catalyst Spaces IoT Services and network assurance on Cisco Catalyst Center. However, IoT Services and intelligent capture (iCAP) port configuration remain mutually exclusive.

gRPC connection configuration details

gRPC connection configuration details.

There are two gRPC connections from AP.

- One connection is to Cisco Catalyst Center for iCAP.
- The other connection is to Cisco Catalyst Center Connector for IoT Services.

The table lists configuration pairs that may or may not coexist on IOS-XE image versions 17.3.2.

Cisco DNA-C Configuration	Cisco Spaces Configuration	Coexistence on IOS-XE Image Version 17.3.2
network-assurance enable	**ap cisco-dna token** *token*	yes
network-assurance icap server port *port*	**ap cisco-dna token** *token*	no

Cisco Spaces: IoT Services is an end-to-end solution. You do not need to manually enable IoT services or Dot15 radio on the controller because Cisco Spaces automatically manages Dot15 radio. You can verify whether Dot15 radio is enabled from the controller interface.

Additionally, Cisco Spaces enables gRPC in the default *AP profile configuration*. Manual activation is not needed, but you can verify the settings on the controller.

Cisco Spaces also enables the *apphost* configuration, required for the default *AP profile* configuration. If Cisco Spaces does not enable *apphost*, you must enable it manually to host IOx applications on an AP.

Enable the 802.15 radio (CLI)

Enable the Dot15 radio using commands.

When you enable the BLE radio configuration globally, APs joined to the controller enable their BLE radio if they have the BLE radio chip in their hardware. This configuration applies to all APs that join the controller after you enable it.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable the dot15 radios for APs, globally.

   Example

   ```
   Device(config)# no ap dot15 shutdown
   ```

3. Disable the dot15 radio for all APs, globally.

   Example

   ```
   Device(config)# ap dot15 shutdown
   ```

Configure the gRPC token (CLI)

Configure the gRPC Token.

Configure the gRPC token so Cisco Spaces can push configuration and certificates to the controller using commands.

> **Note**
>
> - The configuration is pushed automatically from Cisco Spaces. You do not need to manually enable gRPC on the default *ap profile* configuration. You can verify the same on the controller
> - The NETCONF (NETCONF or YANG configuration) must be enabled on the device to allow Cisco Spaces to push the required configuration to the controller. Enable Secure Copy (**ip scp server enable**) on the controller to allow Cisco Spaces to push the gRPC certificate to the controller.
> - The iCAP server port configuration should not be present in the configuration. If the iCAP server port configuration exists, run the **iCAP server port zero** command.

Before you begin

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the Cisco Spaces gRPC token.

   Example

   ```
   Device(config)# ap cisco-dna token {0 | 8} cisco-token-number
   ```

   0 : Specifies the string as an UNENCRYPTED password.

   8 : Indicates the placeholder for backward compatibility.

Enable gRPC in an AP profile (CLI)

Enable gRPC in an AP profile using commands.

Currently, the Manage Streams feature of Cisco Spaces pushes the gRPC configuration only to the default AP profile. If you are using a different AP profile, you must manually configure gRPC.

This procedure explains how to enable gRPC on an AP profile that is not the default-ap-profile. Cisco Spaces may not push gRPC to all AP profiles. Use these commands to enable gRPC for individual AP profiles.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile-name
   ```

3. Enable the gRPC channel on the APs, in the AP profile.

   Example

   ```
   Device(config-ap-profile)# cisco-dna grpc
   ```

Verify BLE state and mode

Verify BLE state and mode.

To verify the BLE state and mode, run the command:

```
Device# show ap ble summary
AP Name          BLE AP State          BLE mode
-------------------------------------------------------------------------
Axel-1            Up                   Advanced (IOx)
Axel-2            Up                   Advanced (IOx)
9117-1            Up                   Advanced (IOx)
3800-1            Up                   Base (Native)
1815              Up                   Base (Native)
9120-3            Up                   Advanced (IOx)
9120-1            Up                   Base (Native)
9115-ax           Up                   Base (Native)
9120-2            Up                   Base (Native)
```

Verify BLE details

Verify BLE details.

To verify BLE details, run the command:

```
Device# show ap name APXXXX.BDXX.29XX ble detail
Mode report time          : 07/28/2020 09:40:57
Mode                      : Base (Native)
Radio mode                : BLE
Admin state report time   : 07/28/2020 09:40:57
Admin state               : Up
Interface report time     : 07/28/2020 09:40:57
Interface                 : MSM1
Interface state           : Open
```

```
Type                        : Integrated
Capability report time    : 07/14/2020 17:10:49
Capability                : BLE, Zigbee, USB,
Host data report time     : 07/28/2020 09:52:04
Host data
  Device name                 : APXXXXBDX
  Dot15 Radio MAC             : 18:04:ed:c5:0e:c8
  API version                 : 1
  FW version                  : 2.7.16
  Broadcast count             : 4389
  Uptime                      : 596050 deciseconds
  Active profile              : viBeacon
Scan Statistics report time    : 07/28/2020 09:40:57
Scan statistics
  Total scan records          : 0
Scan role report time  : 07/28/2020 09:43:19
Scan role
  Scan state                  : Disable
  Scan interval               : 0 seconds
  Scan window                 : 800 milliseconds
  Scan max value              : 8
  Scan filter                 : Enable
Broadcaster role
  Current profile type: iBeacon
  Last report time            : N/A
    UUID                      : Unknown
    Major                     : Unknown
    Minor                     : Unknown
    Transmit power            : Unknown
    Frequency                 : Unknown
    Advertised transmit power : Unknown
  Current profile type: Eddystone URL
  Last report time            : 07/28/2020 09:47:17
    URL                       : https://www.cisco.com
  Current profile type: Eddystone UID
  Last report time            : 07/28/2020 09:43:25
    Namespace                 : 04d77XXXXXXXXXXXXXX
    Instance id               : 5df5XXXXXXXX
  Current profile type: viBeacon
  Last report time            : 07/28/2020 09:52:04
    Interval                  : 450 milliseconds
    Beacon ID                 : 0
      UUID                      : 30XXXXXX-3XXX-4XXX-9XXX-d3XXXXXXXXXX
      Major                     : 36341
      Minor                     : 33196
      Transmit power            : 3 dBm
      Advertised transmit power  : 60 dBm
      Enable                    : Enable
    Beacon ID                 : 1
      UUID                      : 57XXXXXX-cXXX-4XXX-aXXX-85XXXXXXXXXX
      Major                     : 3875
      Minor                     : 567
      Transmit power            : 2 dBm
      Advertised transmit power  : 69 dBm
      Enable                    : Enable
    .
    .
    .
```

Verify gRPC summary, status, and statistics

Verify gRPC summary, status, and statistics.

To verify the gRPC summary, run the command:

```
Device# show ap grpc summary
AP Name              AP Mac             gRPC Status
-------------------------------------------------------------------------------

APXXXX.BDXX.F2XX     0cXX.bdXX.66XX     Up
```

To verify the packet statistics on the gRPC channel that also shows the transmit and receive failures, run the command:

```
Device# show ap name APXXXX.BDXX.F2XX grpc detail
gRPC channel status : Up
Packets transmit attempts : 62
Packets transmit failures : 0
Packets receive count : 62
Packets receive failures : 0
```

85 IoT Module Management in the Controller

Topics:

- IoT module management in the controller

IoT module management in the controller

Explains how IoT modules extend access points and centralize IoT application management through Cisco Catalyst Center.

An IoT module is a hardware extension for APs that

- connects to the controller via the USB interface for IoT integration, and
- supports provisioning, deployment and life cycle management of of third-party containerized applications, that act as containers through Cisco Catalyst Center on the APs.

After connecting the USB modules to the APs, log in to the controller and run commands to enable the USB modules and the Cisco IOx application in the APs associated with an AP profile group.

Supported APs

The IoT Module Management feature uses the USB interface to connect to the Cisco Internet of Things (IoT) connector on these APs:

- Cisco Catalyst 9105AXI
- Cisco Catalyst 9105AXW
- Cisco Catalyst 9115AX
- Cisco Catalyst 9117AX
- Cisco Catalyst 9120AX
- Cisco Catalyst 9130AX

Table 91: Feature history for IoT module management in the controller

Feature Name	Release Information	Feature Description
IoT module management in the controller	Cisco IOS XE 17.3.x	From this release, IoT Module Management enables Cisco Catalyst Series APs to connect IoT hardware modules via the USB interface. This feature allows provisioning, deployment, and lifecycle management of third-party containerized applications on the APs through Cisco Catalyst Center, using the Cisco IOx framework for seamless IoT integration and centralized management.

Enable a USB on the controller

Enable a USB on the controller using commands.

To enable a USB for all the APs connected in an AP profile and to enable Cisco IOx on all the APs, follow this procedure using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile-test
   ```

   Note

   You can use the default AP profile (default-ap-profile) or create a named AP profile, as shown in the example in the adjacent column.

3. Enable the apphost framework on Cisco APs.

   Example

   ```
   Device(config-ap-profile)# apphost
   ```

4. Enable a USB for Cisco APs.

   Example

   ```
   Device(config-ap-profile)# usb-enable
   ```

5. Exit the AP profile configuration mode.

   Example

   ```
   Device(config-ap-profile)# exit
   ```

6. Write running configuration to the memory.

   Example

   ```
   Device(config)# copy running-config startup-config
   ```

Verify the USB modules

Verify the USB modules.

To verify the state of USB modules, run the command:

```
Device# show ap config general
USB Module Type            : USB Module
USB Module State           : Enabled
USB Operational State      : Enabled
USB Override               : Disabled
```

To verify the apphost status, run the command:

```
Device# show ap apphost summary
AP Name                 AP Mac               Apphost Status            CAF
Port        Apphost HW capable
-------------------------------------------------------------------------------------
SS-2027                 00xx.abXX.bXXX       Up                        8443
          Yes
Axel-2036               04xx.40XX.aXXX       Up                        8443
          Yes
Haida-PrePilot          0cxx.f8XX.0XXX       Up                        8443
          Yes
Somer-infra-2022        3cxx.0eXX.0XXX       Up                        8443
          Yes
AP5C71.0DEC.DB5C        3cxx.0eXX.0XXX       Up                        8443
          Yes
AP5C71.0DEC.E3D8        3cxx.0eXX.4XXX       Up                        8443
          Yes
Somer-WP-2021           3cxx.0eXX.5XXX       Up                        8443
          Yes
AP5C71.0DEC.EC60        3cxx.0eXX.9XXX       Up                        8443
          Yes
SS-2005                 6cXX.05XX.dXXX       Up                        8443
          Yes
Vanc-2042               d4XX.bdXX.2XXX       Up                        8443
          Yes
```

To verify the apphost status, run the command:

```
Device# show ap module summary
AP Name         External Module       External Module PID  External Module
Description
-----------------------------------------------------------------------------------
Axel-2036       Enable 10xx/eaXX/100  CP2XXXX              USB to UART Bridge
 C
Haxx-PrePilot   Enable 10xx/eaXX/100  CP2XXXX              USB to UART Bridge
 C
APXXX.0XXX.EXX  Enable 10xx/eaXX/100  CP2XXXX              USB to UART Bridge
 C
SS-2005         Enable 10xx/eaXX/100  CP2XXXX              USB to UART Bridge
 C
Vaxx-2006       Enable 10xx/eaXX/100  CP2XXXX              USB to UART Bridge
 C
```

86 EDCA Parameters

Topics:

- EDCA parameters

EDCA parameters

Introduces EDCA parameters and their role in prioritizing wireless traffic, detailing how to customize channel access for QoS using controller GUI or CLI interfaces.

An Enhanced Distributed Channel Access (EDCA) parameter is a wireless channel access configuration that

- assigns preferential access to voice, video, and other quality of service (QoS) traffic
- differentiates wireless traffic priority through distinct parameter settings, and
- enables customization of wireless performance using controller GUI or CLI interfaces.

Configure EDCA parameters (GUI)

Configure EDCA Parameters using GUI.

Set global EDCA parameters to optimize channel access for voice, video, and quality-of-service traffic on your wireless networks using GUI.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **Parameters**. On this window, you can configure global parameters for 802.11a/n/ac (5 GHz) and 802.11b/g/n (2.4 GHz) radios.

   Note

   You cannot configure or modify parameters, if the radio network is enabled. Disable the network status in the **Configuration** > **Radio Configurations** > **Network** window before proceeding. To apply the EDCA settings to the WLANs, disable and then re-enable the WLANs.

2. In the **EDCA Parameters** section, select an EDCA profile from the **EDCA Profile** drop-down list. Enhanced Distributed Channel Access (EDCA) parameters provide preferential wireless channel access for voice, video, and other QoS traffic.
3. For 802.11a/n/ac (5 GHZ) radios, in the (DFS 802.11h) section, enter the local power constraint. You cannot configure power constraint if the DTPC Support check box on the **Configure** > **Radio Configurations** > **Network** page is checked. The valid range is 0 dBm to 30 dBm.
4. Check the **Channel Switch Announcement Mode** check box, if you want the AP to announce when it is switching to a new channel and the new channel number. The default value is disabled.
5. Check the **Smart DFS** check box to enable Dynamic Frequency Selection (DFS) and avoid radar signal interference.
6. Click **Apply**.

Configure EDCA parameters (CLI)

Configure EDCA parameters using commands.

Set EDCA parameters on a wireless controller to optimize Quality of Service (QoS) for different traffic profiles and improve network performance using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Disable the radio network.

   Example

   ```
   Device(config)# ap dot11 {5ghz | 24ghz | 6ghz} shutdown
   ```

3. Enable specific EDCA parameters for the 802.11a, 802.11b/g, or 802.11 6-GHz network.

   Example

   ```
   Device(config)# ap dot11 {5ghz | 24ghz | 6ghz} edca-parameters
   {client-load-based | custom-voice | fastlane | optimized-video-voice |
   optimized-voice | svp-voice | wmm-default}
   ```

   Note

   The **custom-voice** option is not supported for Cisco Catalyst 9800 Series Wireless Controller .

   - **client-load-based**: Enables client load based EDCA configuration.
   - **custom-voice**: Enables custom voice parameters for the 802.11a or 802.11b/g network.
   - **fastlane**: Enables the fastlane parameters for the 802.11a or 802.11b/g network.
   - **optimized-video-voice**: Enables EDCA voice-optimized and video-optimized parameters for the 802.11a or 802.11b/g network. Choose this option when both voice and video services are deployed on your network.
   - **optimized-voice**: Enables non-SpectraLink voice-optimized profile parameters for the 802.11a or 802.11b/g network. Choose this option when voice services other than SpectraLink are deployed on your network.
   - **svp-voice**: Enables SpectraLink voice-priority parameters for the 802.11a or 802.11b/g network. Choose this option if SpectraLink phones are deployed on your network to improve the quality of calls.
   - **wmm-default**: Enables the Wi-Fi Multimedia (WMM) default parameters for the 802.11a or 802.11b/g network. This is the default option. Choose this option when voice or video services are not deployed on your network.

4. Enable the radio network and return to the privileged EXEC mode.

   Example

   ```
   Device(config)# no ap dot11 {5ghz | 24ghz | 6ghz} shutdown
   ```

   ```
   Device(config)# end
   ```

5. Display the current status of MAC optimization for voice.

Example

```
Device# show ap dot11 {5ghz | 24ghz | 6ghz} network
```

87 Adaptive Client Load-Based EDCA

Topics:

- Adaptive client load-based EDCA

Adaptive client load-based EDCA

Explains how adaptive client load-based EDCA mechanisms dynamically adjust wireless channel access parameters to improve performance in dense deployments.

An adaptive client load-based EDCA mechanism is a wireless medium access control feature that

- dynamically adjusts Enhanced Distributed Channel Access (EDCA) parameters for clients
- responds in real-time to the number of active clients and corresponding load, and
- significantly reduces radio collisions and performance degradation in dense deployments.

Feature Scenario

Run-time EDCA configuration based on active clients and load.

Feature history for adaptive client load-based EDCA

This table provides release and related information for the feature explained in this module.

This feature is available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

Table 92: Feature History for Adaptive Client Load-Based EDCA

Feature Name	Release Information	Feature Description
Adaptive client load-based EDCA	Cisco IOS XE 17.5.1	This Adaptive Client Load-Based EDCA feature dynamically changes Enhanced Distributed Channel Access (EDCA) parameters of clients based on the active client and load that significantly reduce collisions.

Use case for adaptive client load-based EDCA

In a dense multiclient deployment scenario, a customer tested 40 iPads in a classroom or auditorium. He observed channel utilization was 60 to 70 percent.

- The overall AP throughput was lower due to air collision and RTS retries.
- After the adaptive client load-based EDCA feature was enabled, throughput increased by 15 to 20 percent, and collisions decreased by 30 to 40 percent.

Restrictions for adaptive client load-based EDCA

Adaptive client load-based EDCA restrictions.

You must disable the 802.11b network if you want to access the 802.11a network.

Configure adaptive client load-based EDCA (GUI)

Configure adaptive client load-based EDCA using GUI.

Adjust EDCA settings to optimize wireless radio performance based on client load using the GUI.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **Parameters** to configure global parameters for 802.11a/n/ac (5-GHz) and 802.11b/g/n (2.4-GHz) radios.
2. In the **EDCA Parameters** section, select an **EDCA Profile** from the drop-down list.
3. Click the **Client Load Based Configuration** toggle button to enable or disable this feature. It is enabled by default.
4. In the **DFS (802.11h)** section for 802.11a/n/ac (5-GHz) radios, enter the local power constraint. You cannot configure power constraint if the **DPTC Support** check box in **Configuration** > **Radio Configurations** > **Network** is checked. The valid range for power constraint is 0 dBm to 30 dBm.
5. From the **Channel Switch Announcement Mode** drop-down list, select either the **Loud** or **Quiet** mode.
6. Click the **Smart DFS** toggle button to enable or disable this feature. It is enabled by default.
7. In the **11ax Parameters** section, use the corresponding toggle button to enable or disable the following features:
   - **Target Wakeup Time**
   - **Target Wakeup Time Broadcast**
   - **Multiple Bssid**
8. To enable BSS color for both the 5-GHz and 2.4-GHz radios, check the **BSS Color** check box.
9. Click **Apply**.

Configure adaptive client load-based EDCA (CLI)

Configure adaptive client load-based EDCA using commands.

Enable adaptive client load-based EDCA on 802.11 radios to optimize wireless traffic management based on client load using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable client load-based EDCA configuration for 802.11 radios.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} edca-parameters
   client-load-based
   ```

   Use the **no** form of this command to disable the configuration.

> **Note**
>
> To enable the configuration on an 802.11a radio, you must disable the 802.11b network.

3. Return to the privileged EXEC mode.

Example

```
Device(config)# end
```

Verify adaptive client load-based EDCA configuration

Verify adaptive client load-based EDCA configuration.

To verify whether the adaptive client load-based EDCA feature is enabled on an 802.11a or an 802.11b radio, use this command:

```
Device# show ap dot11 24ghz network
Device# show ap dot11 5ghz network
EDCA profile type check                    : default-wmm
Client Load Based EDCA Config              : Enabled
```

To verify whether the adaptive client load-based EDCA feature is enabled on an 802.11 6-GHz radio, use this command:

```
Device# show ap dot11 6ghz network
.
.
.
EDCA profile type check                    : default-wmm
Client Load Based EDCA Config              : Enabled
```

To verify whether the adaptive client load-based EDCA feature is enabled on APs, use this command:

```
Device# show capwap client config
Client Load Based EDCA    : Enabled
```

To view the adaptive EDCA parameters running on the driver, use this command:

```
Device# show controllers dot11Radio 0/1
EDCA Config:
====================
L:Local C:Cell A:Adaptive EDCA params
AC Type CwMin CwMax Aifs Txop ACM
AC_BE L 4  6  3  0  0
AC_BK L 4  10 7  0  0
AC_VI L 3  4  1  94 0
AC_VO L 2  3  1  47 0
AC_BE C 4  10 3  0  0
AC_BK C 4  10 7  0  0
AC_VI C 3  4  2  94 0
```

```
AC_VO C 2 3 2 47 0
AC_BE A 4 10 7 0 0
AC_BK A 4 10 3 0 0
AC_VI A 3 4 2 94 0
AC_VO A 2 3 2 47 0
```

88 802.11 parameters and Band Selection

Topics:

- Band select

Band select

Explains how band select enables dual-band WLAN clients to connect to less congested 5-GHz channels and outlines how the controller regulates probe responses to achieve higher performance.

Band selection is a WLAN feature that

- enables dual-band client radios to connect to less congested 5-GHz APs
- improves overall network performance by reducing interference experienced on the controller, and
- uses probe response regulation to make 5-GHz channels more attractive to clients by delaying probe responses to clients on 2.4-GHz channels.

The 2.4-GHz band is often congested. Clients using the 2.4-GHz band often experience interference from Bluetooth devices, microwave ovens, cordless phones, and other access points. The 802.11b/g limit of three nonoverlapping channels also causes co-channel interference.

Band select works by regulating probe responses to clients and can be enabled on a per-WLAN basis. To view the band select table on an AP, run the **show dot11 band-select** command. It can also be viewed by running the **show cont d0/d1 | begin Lru** command.

You can enable both band selection and aggressive load balancing on the controller. They run independently and do not impact one another.

Band Select Algorithm

The band select algorithm affects clients that use 2.4-GHz band. When a client sends a probe request to an access point, the client's probe Active and Count values become one as seen in the band select table. The algorithm functions based on the following scenarios:

- Scenario 1: The client RSSI (as seen from the show cont d0/d1 | begin RSSIcommand output) is greater than both Mid RSSI and Acceptable Client RSSI.
  1. For dual-band clients, the access point does not send 2.4-GHz probe responses but sends 5-GHz probe responses for all 5-GHz probe requests.
  2. For single-band (2.4-GHz) clients, the access point sends 2.4-GHz probe responses only after the probe suppression cycle.
  3. When the client's probe count matches the configured probe cycle count, the algorithm waits for the Age Out Suppression time, marks the client probe's Active value as zero, and restarts.
- Scenario 2: Client RSSI (as seen from show cont d0/d1 | begin RSSI) lies between Mid-RSSI and Acceptable Client RSSI.
  1. All 2.4-GHz and 5-GHz probe requests are responded to without any restrictions.
  2. This scenario is similar to when band select is disabled.

     Note: The client RSSI value (from the sh cont d0 | begin RSSI command output) averages all client packets received. The Mid RSSI feature reports the instantaneous RSSI value of probe packets. Therefore, the client RSSI often appears weaker than the configured Mid RSSI value (by 7 dB). The 802.11b probes from the client are suppressed to push the client to associate with the 802.11a band.

802.11 bands

Configure the 802.11b/g/n (2.4 GHz) and 802.11a/n (5 GHz) on the controller to meet your country's regulatory requirements. By default, both 802.11b/g/n and 802.11a/n are enabled.

- 802.11n Parameters
- 802.11h Parameters

802.11n Parameters

This section provides instructions for managing 802.11n APs on your network. The 802.11n devices support the 2.4 and 5-GHz bands and offer high throughput data rates.

The 802.11n high throughput rates are available on all the 802.11n APs for the WLANs using WMM with no Layer 2 encryption or with WPA2/AES encryption enabled.

> **Note**
>
> If you want to disable MCS rates for 802.11n, 802.11ac, or 802.11ax, make sure at least one MCS rate remains enabled. To use only legacy 802.11a/b/g rates on your APs, first disable 802.11ax and 802.11ac on the controller for the specific band, then disable 802.11n. When you disable 802.11n globally on the controller, it affects all APs, even those mapped to a Custom RF Profile.

802.11h Parameters

802.11h informs client devices about channel changes and can limit the transmit power for those devices.

Restrictions for band selection, 802.11 bands, and parameters

Band Selection, 802.11 bands and parameters restrictions.

- Band selection-enabled WLANs do not support time-sensitive applications, such as voice and video, because roaming delays impact performance.
- Band selection is supported only on Cisco Wave 2 and 802.11ax APs.

  For more specific APs, see https://www.cisco.com/c/en/us/td/docs/wireless/access_point/feature-matrix/ap-feature-matrix.html .
- Band selection works only when your AP is connected to a controller. If you use a FlexConnect AP without a controller connection, it does not perform band selection after a reboot.
- Dual-band clients are directed from the 2.4-GHz radio to the 5-GHz radio of the same AP, but only when both radios are operational.
- To enable or disable band selection and client load balancing, configure these features for a specific WLAN. These settings are enabled globally by default, and you cannot change them globally through the controller GUI or CLI.

Configure band selection (GUI)

Configure band selection using GUI.

Before you begin

Ensure that you have configured an AP Join profile prior to configuring the primary and backup controllers.

Optimize wireless client performance by controlling band selection settings on the controller using GUI.

Procedure

1. Choose **Configuration** > **Wireless Advanced** > **Band Select**.
2. In the **Cycle Count** field, enter a value between one and 10. The cycle count sets the number of suppression cycles for a new client. The default cycle count is two.
3. In the **Cycle Threshold (milliseconds)** field, enter a value between one and 1000 milliseconds for the scan cycle period threshold. This setting determines the time threshold during which new probe requests from a client come from a new scanning cycle. The default cycle threshold is 200 milliseconds.
4. In the **Age Out Suppression (seconds)** field, enter a value between 10 and 200 seconds. Age-out suppression sets the expiration time for pruning previously known 802.11b/g/n clients. The default value is 20 seconds. After this time elapses, clients become new and are subject to probe response suppression.
5. In the **Age Out Dual Band (seconds)** field, enter a value between 10 and 300 seconds. The age-out period sets the expiration time for pruning previously known dual-band clients. The default value is 50 seconds. After this time elapses, clients become new and are subject to probe response suppression.
6. In the Client RSSI (dbm) field, enter a value between -90 to -20. This is the average of the client packets received.
7. In the Client Mid RSSI (dbm) field, enter a value between -90 to -20. This is the instantaneous RSSI value of the probe packets.
8. On the **AP Join Profile** page, click the AP Join Profile name.
9. Click **Apply**.

Configure band selection (CLI)

Configure band selection using commands.

Enable and fine-tune band selection parameters for wireless clients to optimize network performance using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Set the probe cycle count for band select.

   Example

   ```
   Device(config)# wireless client band-select cycle-count 3
   ```

   The valid range is between one and 10.

3. Set the time threshold for a new scanning cycle period.

   Example

   ```
   Device(config)# wireless client band-select cycle-threshold 5000
   ```

   The valid range is between one and 1000.

4. Set the suppression expire to the band select.

Example

```
Device(config)# wireless client band-select expire suppression 100
```

The valid range is between 10 and 200.

5. Set the dual band expire.

Example

```
Device(config)# wireless client band-select expire dual-band 100
```

The valid range is between 10 and 300.

6. Set the client RSSI threshold.

Example

```
Device(config)# wireless client band-select client-rssi 40
```

The valid range is between 20 and 90.

7. Configure band selection on specific WLANs.

Example

```
Device(config)# wlan wlan1 25 ssid12
```

```
Device(config-wlan)# band-select
```

The valid range is between one and 512. You can specify a value for the *SSID_network_name* parameter using up to 32 alphanumeric characters.

Configure the 802.11 bands (GUI)

Configure the 802.11 bands using GUI.

Modify wireless radio settings to optimize network performance for 2.4 GHz and 5 GHz bands using commands.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **Network**.
2. Click on either **5 GHz Band** or **2.4 GHz Band**. Uncheck the **Network Status** check box to disable the network. This allows configuration of the network parameters.
3. In the **Beacon Interval** field, enter the rate at which the SSID is broadcast by the APs, from 100 to 600 milliseconds. The default is 100 milliseconds. For 802.11b/g/n (2.4-GHz) radios, to enable short preamble on the radio, check the **Short Preamble** check box. A short preamble improves throughput performance.
4. In the **Fragmentation Threshold (in bytes)** field, enter a value between 256 to 2346 bytes. Packets larger than the size you specify here will be fragmented.
5. Check the **DTPC Support** check box to advertise the transmit power level of the radio in the beacons and the probe responses. Client devices using dynamic transmit power control (DTPC) receive channel and power level information from APs. They then adjust their settings automatically. For example, a client device used primarily in Japan could rely on DTPC to adjust its channel and power settings automatically. When it travels to Italy and joins a network there,

the settings are updated as needed. If the **DTPC Support** check box is checked, you cannot configure a power constraint value on your 802.11a/n/ac (5-GHz) radio network. Click **Apply**.

6. In the **CCX Location Measurement** section, check the **Mode** check box to globally enable CCX radio management for the network. Once enabled, APs connected to this device issue broadcast radio measurement requests to clients running CCX v2 or later releases.
7. In the **Interval** field, enter a value to specify how often the APs must issue broadcast radio measurement requests. Click **Apply**.
8. In the **Data Rates** section, choose a value to specify the rates at which data can be transmitted between the AP and the client:
   - Mandatory: Clients must support this data rate to associate with an AP on the controllerembedded wireless controller.
   - Supported: Any associated client supporting this data rate may communicate with the AP using that rate.
   - Disabled: Clients specify the data rates used for communication.
9. Click **Apply** and save the configuration.

Configure the 802.11 bands (CLI)

Configure the 802.11 bands using commands.

Configure the various 802.11 wireless bands and essential parameters using commands.

Follow the procedure given below to configure 802.11 bands and parameters:

Procedure

1. Disable the 802.11a and 802.11b bands.

   Example

   ```
   Device(config)# ap dot11 5ghz shutdown
   ```

   ```
   Device(config)# ap dot11 24ghz shutdown
   ```

   Note

   You must disable the 802.11a band before configuring the 802.11a network parameters.

   You must disable the 802.11b band before configuring the 802.11b network parameters.

2. Disable the 802.11 6-GHz band.

   Example

   ```
   Device(config)# ap dot11 6ghz shutdown
   ```

> **Note**
>
> You must disable the 802.11 6-GHz band before configuring the 802.11 6-GHz network parameters.

3. Specify the rate at which the SSID is broadcast by the corresponding AP and the size at which packets are fragmented.

   Example

   ```
   Device(config)# ap dot11 {5ghz | 24ghz | 6ghz} beaconperiod 500
   ```

   ```
   Device(config)# ap dot11 {5ghz | 24ghz | 6ghz} fragmentation 300
   ```

   The beacon interval is measured in time units (TUs). One TU is 1024 microseconds. You can configure the AP to send a beacon every 20 to 1000 milliseconds.

   The threshold is a value between 256 and 2346 bytes (inclusive). Specify a low number for areas where communication is poor or where there is a great deal of radio interference.

4. Enable APs to advertise their channels and transmit the power levels in beacons and probe responses.

   Example

   ```
   Device(config)# ap dot11 {5ghz | 24ghz | 6ghz} dtpc
   ```

   ```
   Device(config)# no ap dot11 {5ghz | 24ghz | 6ghz} dtpc
   ```

   The default value is enabled. Client devices using dynamic transmit power control (DTPC) receive the channel-level and power-level information from the APs and adjust their settings automatically. For example, a client device used primarily in Japan can rely on DTPC to adjust its channel and power settings automatically when it travels to Italy and joins a network there.

   The **no** form of the command disables the DTPC setting.

5. Specify the maximum allowed number of clients that can be configured.

   Example

   ```
   Device(config)# wireless client association limit number 50 interval
   milliseconds 1000
   ```

   You can configure the maximum number of association requests on a single access point slot at a given interval. The range of association limit that you can configure is from one to 100.

   The association request limit interval is measured between 100 to 10000 milliseconds.

6. Specify the rate at which data can be transmitted between the controllerembedded wireless controller and the client.

   Example

   ```
   Device(config)# ap dot11 {5ghz | 24ghz} rate 36 {disable | mandatory |
   supported}
   ```

   - **disable**: Defines that the clients specify the data rates used for communication.

- **mandatory**: Defines that the clients support this data rate in order to associate to an AP on the controllerembedded wireless controller.
- **supported**: Any associated clients that support this data rate can communicate with the AP using that rate. However, the clients are not required to use this rate in order to associate.
- *rate*: Specifies the rate at which data is transmitted. For the 802.11a and 802.11b bands, the data is transmitted at the rate of one, two, 5.5, six, nine, 11, 12, 18, 24, 36, 48, or 54 Mbps.

7. Enable the 802.11a and 802.11b bands.

   Example

   ```
   Device(config)# no ap dot11 5ghz shutdown
   ```

   ```
   Device(config)# no ap dot11 24ghz shutdown
   ```

   Note

   The default value is enabled.

8. Enable the 802.11 6-GHz band.

   Example

   ```
   Device(config)# no ap dot11 6ghz shutdown
   ```

   Note

   The default value is enabled.

9. Enable or disable 802.11g network support. Then return to privileged EXEC mode.

   Example

   ```
   Device(config)# ap dot11 24ghz dot11g
   ```

   ```
   Device(config)# end
   ```

   The default value is enabled. You can use this command only if the 802.11b band is enabled. If you disable this feature, the 802.11b band is enabled without 802.11g support.

Configure a band-select RF profile (GUI)

Configure a band-select RF profile using GUI.

Configure specific parameters in a band-select RF profile to optimize client band steering behavior on your wireless network.

Procedure

1. Choose **Configuration** > **Wireless** > **Advanced**.
2. In the **Band Select** tab, enter a value between one and 10 in the **Cycle Count** field. The cycle count determines how many suppression cycles are applied to each new client. The default cycle count is two.
3. In the **Cycle Threshold** field, enter a value between one and 1000 milliseconds to define the scan cycle period threshold. his setting specifies the time period in which probe requests from a client are recognized as part of a new scanning cycle. The default cycle threshold value is 200 milliseconds.
4. In the **Age Out Suppression** field, enter a value between 10 and 200 seconds. Age-out suppression establishes the expiration time for removing records of previously known 802.11b/g/n clients. The default value is 20 seconds. After the expiration time, clients are treated as new and become subject to probe response suppression.
5. In the **Age Out Dual Band** field, enter a value between 10 and 300 seconds. The age-out period establishes the expiration time for removing records of previously known dual-band clients. The default value is 50 seconds. After the expiration period, clients are considered new and are subject to probe response suppression.
6. In the **Client RSSI** field, enter a value between -90 dBm and -20 dBm. This value sets the minimum RSSI required for a client to respond to a probe.
7. In the **Client Mid RSSI** field, enter a value between -20 dBm and -90 dBm. This parameter defines the mid-RSSI value, which can be used to toggle 2.4 GHz probe suppression depending on the RSSI.
8. Click **Apply**.

Configure a band-select RF profile (CLI)

Configure a band-select RF profile using commands.

Customize how band selection operates on 2.4 GHz radios by defining parameters in an RF profile using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the RF profile name and enter the RF profile configuration mode.

   Example

   ```
   Device(config)# ap dot11 24ghz rf-profile test1
   ```

3. Set the band-select client threshold.

   Example

   ```
   Device(config-rf-profile)# band-select client {mid-rssi | rssi} -90
   ```

4. Set the band-select cycle parameters.

Example

```
Device(config-rf-profile)# band-select cycle {count | threshold} 10
```

5. Configure the RF profile's band-select expiry time.

Example

```
Device(config-rf-profile)# band-select expire {dual-band | suppression} 100
```

6. Enable the RF profile's band-select probe response.

Example

```
Device(config-rf-profile)# band-select probe-response
```

Configure 802.11n parameters (GUI)

Configure 802.11n parameters using GUI.

Apply custom 802.11n wireless settings to an RF profile in the device management GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF**.
2. Click **Add** to view the **Add RF Profile** window.
3. In the **802.11** tab, complete these steps:
   a) Choose the required operational rates.
   b) Select the required **802.11n MCS Rates** by checking the corresponding check boxes.
4. Click **Save & Apply to Device**.

Configure 802.11n parameters (CLI)

Configure 802.11n parameters using commands.

Configure and optimize 802.11n settings on a wireless AP using commands.

Procedure

1. Enable 802.11n support on the network.

Example

```
Device(config)# ap dot11 {5ghz | 24ghz} dot11n
```

The **no** form of this command disables the 802.11n support on the network.

2. Specify the modulation and coding scheme (MCS) rates for data transmission between the access point and the client.

Example

```
Device(config)# ap dot11 {5ghz | 24ghz} dot11n mcs tx 20
```

rtu: The valid range is zero to 23.

The **no** form of this command disables the MCS rates that are configured.

3. Enable WMM on the WLAN and uses the 802.11n data rates that you configured.

Example

```
Device(config)# wlan wlan1 25 ssid12
```

```
Device(config-wlan)# wmm require
```

The **require** keyword requires client devices to use WMM. Devices that do not support WMM cannot join the WLAN.

4. Disable the network.

Example

```
Device(config)# ap dot11 {5ghz | 24ghz} shutdown
```

5. Specify the aggregation method used for 802.11n packets.

Example

```
Device(config)# {ap | no ap} dot11 {5ghz | 24ghz} dot11n a-mpdu tx priority
{all | 0-7}
```

Aggregation is the process of grouping packet data frames together, rather than transmitting them separately. Two aggregation methods are available: Aggregated MAC Protocol Data Unit (A-MPDU) and Aggregated MAC Service Data Unit (A-MSDU). Both A-MPDU and A-MSDU are performed in the software.

You can specify the aggregation method for various types of traffic from the access point to the clients.

The list defines the priority levels (zero-7) assigned per traffic type.

- Zero–Best effort
- One–Background
- Two–Spare
- Three–Excellent effort
- Four–Controlled load
- Five–Video, less than 100-ms latency and jitter
- Six–Voice, less than 100-ms latency and jitter
- Seven–Network control

You can configure each priority level independently or use all the parameters to configure every priority level simultaneously. You can configure priority levels so that the traffic uses either A-MPDU transmission or A-MSDU transmission.

- When you use the **ap** command with other options, traffic at that priority level uses A-MPDU transmission.

- When you use the **no ap** command along with the other options, the traffic associated with that priority level uses A-MSDU transmission.

  Configure the priority levels to match the aggregation method used by the clients. By default, both A-MPDU and A-MSDU are enabled for priority levels zero to five; all other levels are disabled.

> **Note**
>
> Tracking the TX retries value by internal MSDUs instead of the whole PPDU in Cisco IOS XE 17.12 and later versions provides a more accurate count for the AP's retries. This adjustment has led to a notable increase in retry count values because multiple MSDUs are contained within a single PPDU. This change is intentional and will be implemented in all future versions of Cisco IOS XE.

6. Re-enable the network.

   Example

```
Device(config)# no ap dot11 {5ghz | 24ghz} shutdown
```

7. Configure the guard interval for the network.

   Example

```
Device(config)# ap dot11 {5ghz | 24ghz} dot11n guard-interval {any | long}
```

8. Configure the Reduced Interframe Space (RIFS) for the network.

   Example

```
Device(config)# ap dot11 {5ghz | 24ghz} dot11n rifs rx
```

9. Return to privileged EXEC mode. Alternatively, press **Ctrl-Z** to exit global configuration mode.

   Example

```
Device(config)# end
```

Configure 802.11h parameters (CLI)

Configure 802.11h parameters using commands.

Configure 802.11h parameters on a wireless AP using commands to manage compliance, interference, and transmit power settings.

Procedure

1. Disable the 802.11 network.

Example

```
Device(config)# ap dot11 5ghz shutdown
```

2. Disable the 802.11 6-GHz network.

Example

```
Device(config)# ap dot11 6ghz shutdown
```

3. Enable or disable the AP to announce when it is switching to a new channel.

Example

```
Device(config)# {ap | no ap} dot11 5ghz channelswitch mode 0
```

switch_mode: Enter zero or one to specify whether transmissions are restricted until the actual channel switch (***zero***) or are not restricted (***one***). The default value is disabled.

4. Enable or disable the AP to announce when it is switching to a new channel in quiet mode.

Example

```
Device(config)# ap dot11 5ghz channelswitch quiet
```

```
Device(config)# no ap dot11 6ghz channelswitch quiet
```

5. Configure the 802.11h power constraint value in dB.

Example

```
Device(config)# ap dot11 5ghz power-constraint 200
```

The valid range is from zero to 255. The default value is three.

6. Configure the 802.11 6-GHz power constraint value in dB.

Example

```
Device(config)# ap dot11 5ghz power-constraint 200
```

The valid range is from zero to 30. The default value is three.

7. Re-enable the 802.11a network.

Example

```
Device(config)# no ap dot11 5ghz shutdown
```

8. Re-enable the 802.11 6-GHz network.

Example

```
Device(config)# no ap dot11 6ghz shutdown
```

9. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

Verify configuration settings using band selection and 802.11 bands commands

Verify configuration settings using band selection and 802.11 bands commands.

Use these commands to verify band selection, 802.11 bands, and parameters on the device.

Table 93: Monitoring configuration settings using band selection and 802.11 band commands

Command	Purpose
show ap dot11 5ghz network	Displays 802.11a band network parameters, 802.11a operational rates, 802.11n MCS settings, and 802.11n status information.
show ap dot11 24ghz network	Displays 802.11b band network parameters, 802.11b/g operational rates, 802.11n MCS settings, and 802.11n status information.
show ap dot11 6ghz network	Displays 802.116-GHz band network parameters, 802.11b/g operational rates, 802.11n MCS settings, and 802.11n status information.
show wireless dot11h	Displays 802.11h configuration parameters.
show wireless band-select	Displays band-select configuration settings.

Example: view the configuration settings for the 6-GHz band

View the 6-GHz band example configuration settings.

```
Device# show ap dot11 6ghz network

802.11 6Ghz Network                       : Enabled
802.11 6Ghz Status:
  A-MPDU Tx:
    Priority 0                            : Enabled
    Priority 1                            : Enabled
    Priority 2                            : Enabled
    Priority 3                            : Enabled
    Priority 4                            : Enabled
    Priority 5                            : Enabled
    Priority 6                            : Disabled
    Priority 7                            : Disabled
  A-MSDU Tx:
    Priority 0                            : Enable
    Priority 1                            : Enable
    Priority 2                            : Enable
    Priority 3                            : Enable
    Priority 4                            : Enable
    Priority 5                            : Enable
```

```
    Priority 6                              : Disable
    Priority 7                              : Disable
802.11ax                                    : Enabled
  DynamicFrag                               : Enabled
  MultiBssid                                : Disabled
  Target Wakeup Time                        : Enabled
  Target Wakeup Time Broadcast              : Enabled
  BSS Color                                 : Disabled
  OBSS PD                                   : Disabled
  Non-SRG OBSS PD Maximum Threshold         : -62 dBm
  SRG OBSS PD                               : Disabled
  SRG OBSS PD Minimum Threshold             : -82 dBm
  SRG OBSS PD Maximum Threshold             : -62 dBm
802.11ax MCS Settings:
  MCS  7, Spatial Streams = 1               : Supported
  MCS  9, Spatial Streams = 1               : Disabled
  MCS 11, Spatial Streams = 1               : Supported
  MCS  7, Spatial Streams = 2               : Supported
  MCS  9, Spatial Streams = 2               : Disabled
  MCS 11, Spatial Streams = 2               : Supported
  MCS  7, Spatial Streams = 3               : Supported
  MCS  9, Spatial Streams = 3               : Disabled
  MCS 11, Spatial Streams = 3               : Supported
  MCS  7, Spatial Streams = 4               : Supported
  MCS  9, Spatial Streams = 4               : Disabled
  MCS 11, Spatial Streams = 4               : Supported
Beacon Interval                             : 95
CF Pollable mandatory                       : Disabled
CF Poll Request Mandatory                   : Disabled
CFP Period                                  : 4
CFP Maximum Duration                        : 60
Default Channel                             : 1
Default Tx Power Level                      : 1
DTPC Status                                 : Enabled
Fragmentation Threshold                     : 2335
RSSI Low Check                             : Disabled
RSSI Threshold                              : -127 dbm
TI Threshold                                :
Legacy Tx Beamforming setting               : Disabled
Traffic Stream Metrics Status               : Disabled
Expedited BW Request Status                 : Disabled
EDCA profile type check                     : default-wmm
Client Load Based EDCA Config               : Enabled
Call Admision Control (CAC) configuration
Voice AC
  Voice AC - Admission control (ACM)        : Disabled
  Voice Stream-Size                         : 84000
  Voice Max-Streams                         : 2
  Voice Max RF Bandwidth                    : 75
  Voice Reserved Roaming Bandwidth          : 6
  Voice Load-Based CAC mode                 : Enabled
  Voice tspec inactivity timeout            : Enabled
CAC SIP-Voice configuration
  SIP based CAC                             : Disabled
  SIP call bandwidth                        : 64
  SIP call bandwidth sample-size             : 20
Maximum Number of Clients per AP Radio      : 200
WiFi to Cellular RSSI Threshold             : -85 dbm
Client Network Preference                   : default
```

Example: view the configuration settings for the 5-GHz band

View the 5-GHz band example configuration settings.

```
Device# show ap dot11 5ghz network

802.11a Network : Enabled
11nSupport : Enabled
  802.11a Low Band : Enabled
  802.11a Mid Band : Enabled
  802.11a High Band : Enabled
802.11a Operational Rates
  802.11a 6M : Mandatory
  802.11a 9M : Supported
  802.11a 12M : Mandatory
  802.11a 18M : Supported
  802.11a 24M : Mandatory
  802.11a 36M : Supported
  802.11a 48M : Supported
  802.11a 54M : Supported
802.11n MCS Settings:
  MCS 0 : Supported
  MCS 1 : Supported
  MCS 2 : Supported
  MCS 3 : Supported
  MCS 4 : Supported
  MCS 5 : Supported
  MCS 6 : Supported
  MCS 7 : Supported
  MCS 8 : Supported
  MCS 9 : Supported
  MCS 10 : Supported
  MCS 11 : Supported
  MCS 12 : Supported
  MCS 13 : Supported
  MCS 14 : Supported
  MCS 15 : Supported
  MCS 16 : Supported
  MCS 17 : Supported
  MCS 18 : Supported
  MCS 19 : Supported
  MCS 20 : Supported
  MCS 21 : Supported
  MCS 22 : Supported
  MCS 23 : Supported
802.11n Status:
  A-MPDU Tx:
    Priority 0 : Enabled
    Priority 1 : Disabled
    Priority 2 : Disabled
    Priority 3 : Disabled
    Priority 4 : Enabled
    Priority 5 : Enabled
    Priority 6 : Disabled
    Priority 7 : Disabled
  A-MSDU Tx:
    Priority 0 : Enabled
    Priority 1 : Enabled
    Priority 2 : Enabled
    Priority 3 : Enabled
    Priority 4 : Enabled
    Priority 5 : Enabled
```

```
    Priority 6 : Disabled
    Priority 7 : Disabled
Guard Interval : Any
  Rifs Rx : Enabled
Beacon Interval : 100
CF Pollable mandatory : Disabled
CF Poll Request Mandatory : Disabled
CFP Period : 4
CFP Maximum Duration : 60
Default Channel : 36
Default Tx Power Level : 1
DTPC Status : Enabled
Fragmentation Threshold : 2346
Pico-Cell Status : Disabled
Pico-Cell-V2 Status : Disabled
TI Threshold : 0
Legacy Tx Beamforming setting : Disabled
Traffic Stream Metrics Status : Disabled
Expedited BW Request Status : Disabled
EDCA profile type check : default-wmm
Call Admision Control (CAC) configuration
Voice AC
  Voice AC - Admission control (ACM) : Disabled
  Voice Stream-Size : 84000
  Voice Max-Streams : 2
  Voice Max RF Bandwidth : 75
  Voice Reserved Roaming Bandwidth : 6
  Voice Load-Based CAC mode : Enabled
  Voice tspec inactivity timeout : Enabled
CAC SIP-Voice configuration
  SIP based CAC : Disabled
  SIP Codec Type : CODEC_TYPE_G711
  SIP call bandwidth : 64
  SIP call bandwidth sample-size : 20
Video AC
  Video AC - Admission control (ACM) : Disabled
  Video max RF bandwidth : Infinite
  Video reserved roaming bandwidth : 0
```

Example: view the configuration settings for the 2.4-GHz band

View the 2.4-GHz band example configuration settings.

```
Device# show ap dot11 24ghz network

802.11b Network : Enabled
11gSupport : Enabled
11nSupport : Enabled
  802.11b/g Operational Rates
  802.11b 1M : Mandatory
  802.11b 2M : Mandatory
  802.11b 5.5M : Mandatory
  802.11g 6M : Supported
  802.11g 9M : Supported
  802.11b 11M : Mandatory
  802.11g 12M : Supported
  802.11g 18M : Supported
  802.11g 24M : Supported
  802.11g 36M : Supported
  802.11g 48M : Supported
```

```
  802.11g 54M : Supported
802.11n MCS Settings:
  MCS 0 : Supported
  MCS 1 : Supported
  MCS 2 : Supported
  MCS 3 : Supported
  MCS 4 : Supported
  MCS 5 : Supported
  MCS 6 : Supported
  MCS 7 : Supported
  MCS 8 : Supported
  MCS 9 : Supported
  MCS 10 : Supported
  MCS 11 : Supported
  MCS 12 : Supported
  MCS 13 : Supported
  MCS 14 : Supported
  MCS 15 : Supported
  MCS 16 : Supported
  MCS 17 : Supported
  MCS 18 : Supported
  MCS 19 : Supported
  MCS 20 : Supported
  MCS 21 : Supported
  MCS 22 : Supported
  MCS 23 : Supported
802.11n Status:
  A-MPDU Tx:
    Priority 0 : Enabled
    Priority 1 : Disabled
    Priority 2 : Disabled
    Priority 3 : Disabled
    Priority 4 : Enabled
    Priority 5 : Enabled
    Priority 6 : Disabled
    Priority 7 : Disabled
  A-MSDU Tx:
    Priority 0 : Enabled
    Priority 1 : Enabled
    Priority 2 : Enabled
    Priority 3 : Enabled
    Priority 4 : Enabled
    Priority 5 : Enabled
  Priority 6 : Disabled
    Priority 7 : Disabled
  Guard Interval : Any
  Rifs Rx : Enabled
Beacon Interval : 100
CF Pollable Mandatory : Disabled
CF Poll Request Mandatory : Disabled
CFP Period : 4
CFP Maximum Duration : 60
Default Channel : 11
Default Tx Power Level : 1
DTPC Status : true
Call Admission Limit : 105
G711 CU Quantum : 15
ED Threshold : -50
Fragmentation Threshold : 2346
PBCC Mandatory : Disabled
Pico-Cell Status : Disabled
Pico-Cell-V2 Status : Disabled
```

```
RTS Threshold : 2347
Short Preamble Mandatory : Enabled
Short Retry Limit : 7
Legacy Tx Beamforming setting : Disabled
Traffic Stream Metrics Status : Disabled
Expedited BW Request Status : Disabled
EDCA profile type : default-wmm
Call Admision Control (CAC) configuration
Voice AC
  Voice AC - Admission control (ACM) : Disabled
  Voice Stream-Size : 84000
  Voice Max-Streams : 2
  Voice Max RF Bandwidth : 75
  Voice Reserved Roaming Bandwidth : 6
  Voice Load-Based CAC mode : Enabled
  Voice tspec inactivity timeout : Enabled
CAC SIP-Voice configuration
  SIP based CAC : Disabled
  SIP Codec Type : CODEC_TYPE_G711
  SIP call bandwidth : 64
  SIP call bandwidth sample-size : 20
Video AC
  Video AC - Admission control (ACM) : Disabled
  Video max RF bandwidth : Infinite
  Video reserved roaming bandwidth : 0
```

Example: view the status of 802.11h parameters

View the status of 802.11h parameters example.

```
Device# show wireless dot11
Power Constraint: 0
Channel Switch : Enabled
Channel Switch Mode : Quiet
Smart DFS : Enabled
```

Example: verify the band-selection settings

Verify the band-selection example settings.

This example displays a band-select configuration:

```
Device# show wireless band-select

Band Select Probe Response   : per WLAN enabling
Cycle Count                  : 2
Cycle Threshold (millisec)   : 200
Age Out Suppression (sec)    : 20
Age Out Dual Band (sec)      : 60
Client RSSI (dBm)            : -80
Client Mid RSSI (dBm)        : -80
```

This example displays an AP RF profile details:

```
Device# show ap rf-profile name vid detail
Description                          :
RF Profile Name                      : vid
Band                                 : 2.4 GHz
802.11n client only                  : Disabled
```

```
Transmit Power Threshold v1       : -70 dBm
Min Transmit Power                : -10 dBm
Max Transmit Power                : 30 dBm
Operational Rates
  802.11b 1M Rate                 : Mandatory
  802.11b 2M Rate                 : Mandatory
  802.11b 5.5M Rate               : Mandatory
  802.11b 11M Rate                : Mandatory
  802.11b 6M Rate                 : Supported
  802.11b 9M Rate                 : Supported
  802.11b 12M Rate                : Supported
  802.11b 18M Rate                : Supported
  802.11b 24M Rate                : Supported
  802.11b 36M Rate                : Supported
  802.11b 48M Rate                : Supported
  802.11b 54M Rate                : Supported
Max Clients                       : 200
Trap Threshold
  Clients                         : 12 clients
  Interference                    : 10%
  Noise                           : -80 dBm
  Utilization                     : 10%
Multicast Data Rate               : auto
Rx SOP Threshold                  : auto
Band Select
  Probe Response                  :  Disabled
  Cycle Count                     :  2 cycles
  Cycle Threshold                 :  200 milliseconds
  Expire Suppression              :  20 seconds
  Expire Dual Band                :  60 seconds
  Client RSSI                     :  -80 dBm
  Client Mid RSSI                 :  -80 dBm
High Speed Roam
  hsr mode                        :  Disabled
  hsr neighbor timeout            :  5
Load Balancing
  Window                          :  5 clients
  Denial                          :  3 count
Coverage Data
  Data                            : -62 dBm
  Voice                           : -80 dBm
  Minimum Client Level            : 12 clients
  Exception Level                 : 48%
DCA Channel List                  : 1,6,11
Unused Channel List               : 2,3,4,5,7,8,9,10
DCA Foreign AP Contribution       : Enabled
802.11n MCS Rates
  MCS  0                          : Enabled
  MCS  1                          : Enabled
  MCS  2                          : Enabled
  MCS  3                          : Enabled
  MCS  4                          : Enabled
  MCS  5                          : Enabled
  MCS  6                          : Enabled
  MCS  7                          : Enabled
  MCS  8                          : Enabled
  MCS  9                          : Enabled
  MCS 10                          : Enabled
  MCS 11                          : Enabled
  MCS 12                          : Enabled
  MCS 13                          : Enabled
  MCS 14                          : Enabled
```

```
  MCS 15                          : Enabled
  MCS 16                          : Enabled
  MCS 17                          : Enabled
  MCS 18                          : Enabled
  MCS 19                          : Enabled
  MCS 20                          : Enabled
  MCS 21                          : Enabled
  MCS 22                          : Enabled
  MCS 23                          : Enabled
  MCS 24                          : Enabled
  MCS 25                          : Enabled
  MCS 26                          : Enabled
  MCS 27                          : Enabled
  MCS 28                          : Enabled
  MCS 29                          : Enabled
  MCS 30                          : Enabled
  MCS 31                          : Enabled
State                             : Up
Client Network  Preference        : connectivity
```

Examples: band selection configuration

Band selection configuration examples.

This example shows how to set the probe cycle count and time threshold for a new scanning cycle period for band select:

```
Device# configure terminal
(config)# wireless client band-select cycle-count 3
(config)# wireless client band-select cycle-threshold 5000
(config)# end
```

This example shows how to set the suppression expiry time to the band select:

```
Device# configure terminal
(config)# wireless client band-select expire suppression 100
(config)# end
```

This example shows how to set the dual-band expiry time for the band select:

```
Device# configure terminal
(config)# wireless client band-select expire dual-band 100
(config)# end
```

This example shows how to set the client RSSI threshold for the band select:

```
Device# configure terminal
(config)# wireless client band-select client-rssi 40
(config)# end
```

This example shows how to configure band selection on specific WLANs:

```
Device# configure terminal
(config)# wlan wlan1 25 ssid12
(config-wlan)# band-select
(config)# end
```

Examples: 802.11 bands configuration

802.11 bands configuration examples.

This example shows how to configure 802.11 bands using beacon interval, fragmentation, and dynamic transmit power control:

```
Device# configure terminal
config)# ap dot11 5ghz shutdown
(config)# ap dot11 24ghz shutdown
(config)# ap dot11 5ghz beaconperiod 500
(config)# ap dot11 5ghz fragmentation 300
(config)# ap dot11 5ghz dtpc
(config)# wireless client association limit 50 interval 1000
(config)# ap dot11 5ghz rate 36 mandatory
(config)# no ap dot11 5ghz shutdown
(config)# no ap dot11 24ghz shutdown
(config)# ap dot11 24ghz dot11g
(config)# end
```

Examples: 802.11n configuration

802.11n configuration examples.

This example shows how to configure 802.11n parameters for 5-GHz band using aggregation method:

```
Device# configure terminal
(config)# ap dot11 5ghz dot11n
(config)# ap dot11 5ghz dot11n mcs tx 20
(config)# wlan wlan1 25 ssid12
(config-wlan)# wmm require
(config-wlan)# exit
(config)# ap dot11 5ghz shutdown
(config)# ap dot11 5ghz dot11n a-mpdu tx priority all
(config)# no ap dot11 5ghz shutdown
(config)# exit
```

This example shows how to configure the guard interval for 5-GHz band:

```
Device# configure terminal
(config)# ap dot11 5ghz dot11n
(config)# ap dot11 5ghz dot11n mcs tx 20
(config)# wlan wlan1 25 ssid12
(config-wlan)# wmm require
(config-wlan)# exit
(config)# no ap dot11 5ghz shutdown
(config)# ap dot11 5ghz dot11n guard-interval long
(config)# end
```

This example shows how to configure the RIFS for 5-GHz band:

```
Device# configure terminal
(config)# ap dot11 5ghz dot11n
(config)# ap dot11 5ghz dot11n mcs tx 20
(config)# wlan wlan1 25 ssid12
(config-wlan)# wmm require
(config-wlan)# exit
(config)# ap dot11 5ghz shutdown
(config)# ap dot11 5ghz dot11n rifs rx
(config)# end
```

Examples: 802.11h configuration

802.11h configuration examples.

This example shows how to configure the AP to announce when it is switching to a new channel using restriction transmission:

```
Device# configure terminal
(config)# ap dot11 5ghz shutdown
(config)# ap dot11 5ghz channelswitch mode 0
(config)# no ap dot11 5ghz shutdown
(config)# end
```

This example shows how to configure the 802.11h power constraint for 5-GHz band:

```
Device# configure terminal
(config)# ap dot11 5ghz shutdown
(config)# ap dot11 5ghz power-constraint 200
(config)# no ap dot11 5ghz shutdown
(config)# end
```

89 NBAR Protocol Discovery

Topics:

- NBAR protocol discovery

NBAR protocol discovery

Explains NBAR Protocol Discovery, detailing supported protocols, configuration steps for traffic monitoring through GUI and CLI, and methods for verifying protocol statistics to improve network traffic management and application visibility.

NBAR protocol discovery is a network traffic analysis feature that

- identifies and classifies application protocols passing through a network interface from Layer 4 through Layer 7
- collects protocol-specific traffic statistics for supported applications, and
- enables network administrators to monitor application performance and usage trends.

Network Based Application Recognition (NBAR) identifies which protocols and applications are currently running on the network. Use Protocol Discovery to find protocol traffic supported by NBAR and view related statistics.

NBAR also supports Cisco Application Visibility and Control (AVC). With AVC, NBAR helps you improve application performance with enhanced QoS and policing. It gives you more detailed visibility into your network.

Restrictions of NBAR

NBAR cannot accurately classify traffic when SaaS applications use end-to-end encryption, QUIC, or DoH, because encryption blocks traffic analysis.

- In these cases, encrypted traffic (including DoH and QUIC without SNI) limits NBAR, so it may not report the correct Protocol ID. As a result, you might experience issues with traffic classification.

Configure NBAR protocol discovery (CLI)

Configure NBAR protocol discovery using commands.

Follow this procedure to enable protocol discovery.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy nbar-proto-policy
   ```

3. Configure the wireless policy profile for central switching.

   Example

   ```
   Device(config-wireless-policy)# central switching
   ```

> **Note**
>
> NBAR protocol discovery is supported in local mode (central switching) and in FlexConnect (central switching) mode.

4. Enable application recognition on the wireless policy profile by activating the NBAR2 engine.

Example

```
Device(config-wireless-policy)# ip nbar protocol-discovery
```

Verify protocol discovery statistics

Verify protocol discovery statistics.

To view protocol discovery statistics, use the command:

```
Device# show ip nbar protocol-discovery wlan wlan-profile-name
wlan_profile_name (iif_id 0xF0400002)


Last clearing of "show ip nbar protocol-discovery" counters 00:07:12



Input                    Output
-----                    ------
Protocol                 Packet Count             Packet Count
Byte Count               Byte Count
5min Bit Rate (bps)      5min Bit Rate (bps)
5min Max Bit Rate (bps)  5min Max Bit Rate (bps)
------------------------ ------------------------ ------------------------
unknown                  22                       0
                                  4173                     0
                                  0                        0
                                 2000                     0
dhcp                         3                        2
                                 1166                     724
                                  0                        0
                                  0                        0
 ping                          2                        2
                                  204                      236
                                  0                        0
                                  0                        0
Total                         27                       4
                                 5543                     960
                                 0                        0
                                 2000                     0
```

To clear protocol discovery statistics, use the command:

```
Device# clear ip nbar protocol-discovery wlan wlan-profile-name
```

90 Conditional Debug, Radioactive Tracing, and Packet Tracing

Topics:

- Conditional debugging
- Packet tracing
- Wireless client debug bundle

Conditional debugging

Introduces conditional debugging, enabling selective activation of debug and logging outputs based on user-defined conditions such as interface, IP address, or MAC address to optimize resource usage and performance in large-scale networks.

A conditional debugging is a diagnostic feature that

- enables selective activation of debugging and logging for specific features based on conditions you define
- allows you to define custom conditions such as interface, IP address, or MAC address to narrow debug scope, and
- helps reduce system resource usage and performance impact in large, complex networks.

Use this feature in systems with many supported features.

Conditional debug allows granular debugging in networks operating at large scale with many features. It allows you to observe detailed debug output for specific instances within the system.

In contrast, the general debug command produces output without filtering feature objects being processed. The general debug command uses significant system resources and affects performance.

Radioactive tracing

Introduces radioactive tracing for enhanced diagnostic logging across system components, outlines conditional debugging techniques for FHS and mesh networking, and details relevant commands, supported components, and important considerations such as certificate validity and MAC address configuration.

A radioactive tracing is a diagnostic tracing mechanism that

- stitches together a chain of execution for operations of interest across the system, at an increased verbosity level
- enables conditional printing of detailed debug information across threads, processes and function calls, and
- troubleshoot system features, especially First-Hop Security (FHS) and mesh networking functions.

Note

- For more information on First Hop Security features, see **System Management** > **Wireless Multicast** > **Information About Wireless Multicast** > **Information About IPv6 Snooping**.
- The radioactive tracing filter does not work, if the certificate is not valid.
- To debug mesh features effectively, add both Ethernet and Radio MAC addresses as conditional MACs for RA tracing when you collect logs.
- To enable debugging for wireless IPs, use the **debug platform condition feature wireless ip** *ip-address* command.

Table 94: Components Supporting Radio Active Tracing

Components	Details
SISF or FHS	The first-hop security features, includes IPv6 Address Glean and IPv6 Device Tracking. For more information, see *Information About IPv6 Snooping*.

Components	Details
LISP	Locator or ID Separation Protocol.

Feature history for conditional debug and radioactive tracing

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 95: Feature history for conditional debug and radioactive tracing

Feature Name	Release Information	Feature Description
Conditional debug and radioactive tracing	Cisco IOS XE 17.11.1	Conditional debugging is a diagnostic feature that selectively activates debugging and logging based on defined conditions to optimize resource usage and performance in complex networks, while radioactive tracing is a diagnostic mechanism that stitches together execution chains with increased verbosity, enabling conditional detailed debug information across system components for troubleshooting specific features.

Conditional debugging and radioactive tracing

Introduces conditional debugging and radioactive tracing, detailing how to target specific execution contexts, trace code paths across multiple control flows, and use a unified debug CLI, as well as how to clear applied debug conditions.

A conditional debugging and radioactive tracing feature is a diagnostic mechanism that

- combines the precision of conditional debugging to target specific execution contexts
- leverages radioactive tracing to follow code paths across multiple control flows, and
- enables use of a single debug CLI to debug all execution contexts related to the condition.

> **Note**
>
> Use the **clear platform condition all** command to remove the debug conditions applied to the platform.

Location of tracefiles

Tracefile location details.

Tracefiles are generated for each process and saved in either the `/tmp/rp/trace` or `/tmp/fp/trace` directory. Each trace log is written to a file of 1 MB. The directory can hold up to a maximum of 25 such files for a given process.

When a tracefile in the `/tmp` directory reaches its 1MB limit or the configured size, it is rotated out to an archive location in the `/crashinfo` partition under the `tracelogs` directory.

- Process-name_Process-ID_running-counter.timestamp.gz

  Example: IOSRP_R0-0.bin_0.14239.20151101234827.gz

- Process-name_pmanlog_Process-ID_running-counter.timestamp.bin.gz

  Example: wncmgrd_R0-0.27958_1.20180902081532.bin.gz

The `/tmp` directory holds only a single tracefile for a given process. Once a tracefile reaches its size limit, the system moves it to `/crashinfo/tracelogs`. The archive directory stores up to 25 files and replaces the oldest file with each new file from `/tmp`. File size varies by process. Some processes use larger files that can reach 10 MB.

- The number of files stored in the `tracelogs` directory also depends on the process. For example, the WNCD process may use a limit of up to 400 files per instance, depending on the platform.
- You can verify these logs (per-process) using the **show platform software trace message** *process_name* **chassis active R0** command.

Configure conditional debugging (GUI)

Configure conditional debugging using GUI.

Enable targeted troubleshooting by configuring conditional debugging through the GUI, allowing you to capture and analyze logs for specific MAC or IP addresses.

Procedure

1. Choose **Troubleshooting** > **Radioactive Trace**. Click **Add**.
2. Enter the **MAC/IP Address**. The MAC address can be either in *xx:xx:xx:xx:xx:xx* , *xx-xx-xx-xx-xx-xx* , or *xxxx.xxxx.xxxx* format.
3. Click **Apply to Device**.
4. Click **Start** to start or **Stop** to stop the conditional debug.
5. Click **Generate** to create a radioactive trace log.
6. Click the radio button to set the time interval.
7. Click the **Download Logs** icon that is displayed next to the trace file name, to download the logs to your local folder.
8. Click the **View Logs** icon that is displayed next to the trace file name, to view the log files on the GUI page.
9. Click **Load More** to view more lines of the log file. Click **Apply to Device**.

Configure conditional debugging (CLI)

Configure conditional debugging using commands.

Enable targeted troubleshooting for wireless features by filtering debug output to specific APs, clients, or IP addresses using commands.

Procedure

1. Configure conditional debugging for a feature using the specified MAC address.

Example

```
Device# debug platform condition feature wireless mac b838.61a1.5433
```

> **Note**
>
> This is supported with AP or client MAC/IP and also on CMX IP address and mobility peer IP.

2. Start conditional debugging.

Example

```
Device# debug platform condition start
```

This starts radioactive tracing if there is a match on one of the conditions above.

> **Note**
>
> This is supported with AP or client MAC/IP and also on CMX IP address and mobility peer IP.

3. Display the current conditions set.

Example

```
Device# show platform condition
```

```
Device# show debugging
```

4. Stop conditional debugging.

Example

```
Device# debug platform condition stop
```

This will stop radioactive tracing.

> **Note**
>
> This is supported with AP or client MAC/IP and also on CMX IP address and mobility peer IP.

5. Display the logs from the latest wireless profile.

Example

```
Device# show logging profile wireless [ counter start last {20 minutes} |
filter mac {mac-address}  to-file bootflash:logs.txt]
```

> **Note**
>
> You can use either the **show logging profile wireless** command or **show logging process** command to collect the logs.

6. Display the logs collection specific to the process.

   Example

   ```
   Device# show logging process wncd to-file flash:wncd.txt
   ```

7. Clear all conditions.

   Example

   ```
   Device# clear platform condition all
   ```

What to do next

The command **request platform software trace filter-binary wireless** *mac-address* generates 3 flash files:

- collated_log_<.date..>
- mac_log <..date..>
- mac_database .. file

The *mac_log <..date..>* file is the most important because it contains messages related to the MAC address under investigation. The **show platform software trace filter-binary** command generates the same flash files and displays the mac_log on the screen.

Radioactive Tracing for L2 Multicast

L2 Multicast Radioactive Tracing details.

To identify a specific multicast receiver, specify the MAC address of the joiner or the receiver client, Group Multicast IP address and Snooping VLAN. Additionally, enable the trace level for the debug. The debug level will provide detailed traces and better visibility into the system.

debug platform condition feature multicast controlplane *igmp-mld-debug igmp-snooping level mld-snooping pim*

Recommended workflow for trace files

Trace files recommended workflow details.

The recommended workflow for trace files is listed below:

- To request the tracelogs for a specific time period, such as 1 day, use the command: Device# **show logging process wncd to-file flash:wncd.txt**.
- The system generates a text file of the tracelogs in the location /flash:
- Copy the file off the device so the tracelogs can be used to work offline. For more details on copying files, see section below.
- Delete the tracelog file (.txt) from the /flash: location to ensure enough space on the device for other operations.

Copy tracefiles off the box

Off the box copying tracefiles details.

This is an example of the tracefile:

```
Device# dir crashinfo:/tracelogs
Directory of crashinfo:/tracelogs/
50664 -rwx 760 Sep 22 2015 11:12:21 +00:00 plogd_F0-0.bin_0.gz
50603 -rwx 991 Sep 22 2015 11:12:08 +00:00
fed_pmanlog_F0-0.bin_0.9558.20150922111208.gz
50610 -rw- 11 Nov 2 2015 00:15:59 +00:00 timestamp
50611 -rwx 1443 Sep 22 2015 11:11:31 +00:00
auto_upgrade_client_sh_pmanlog_R0-.bin_0.3817.20150922111130.gz
50669 -rwx 589 Sep 30 2015 03:59:04 +00:00 cfgwr-8021_R0-0.bin_0.gz
50612 -rwx 1136 Sep 22 2015 11:11:46 +00:00
reflector_803_R0-0.bin_0.1312.20150922111116.gz
               50794 -rwx 4239 Nov 2 2015 00:04:32 +00:00
IOSRP_R0-0.bin_0.14239.20151101234827.gz
50615 -rwx 131072 Nov 2 2015 00:19:59 +00:00 linux_iosd_image_pmanlog_R0-0.bin_0
```

The trace files can be copied using one of the various options shown below:

```
Device# copy crashinfo:/tracelogs ?
  crashinfo: Copy to crashinfo: file system
  flash: Copy to flash: file system
  ftp: Copy to ftp: file system
  http: Copy to http: file system
  https: Copy to https: file system
  null: Copy to null: file system
  nvram: Copy to nvram: file system
  rcp: Copy to rcp: file system
  running-config Update (merge with) current system configuration
  scp: Copy to scp: file system
  startup-config Copy to startup configuration
  syslog: Copy to syslog: file system
  system: Copy to system: file system
  tftp: Copy to tftp: file system
  tmpsys: Copy to tmpsys: file system
```

The general syntax for copying onto a TFTP server is as follows:

```
Device# copy source: tftp:
Device# copy crashinfo:/tracelogs/IOSRP_R0-0.bin_0.14239.20151101234827.gz
tftp:
Address or name of remote host []? 2.2.2.2
Destination filename [IOSRP_R0-0.bin_0.14239.20151101234827.gz]?
```

> **Note**
>
> Clear the generated report and archive files from the switch to ensure that sufficient flash space remains available for tracelog and other uses.

Configuration examples for conditional debugging

Conditional debugging configuration examples.

This is an output example of the **show platform condition** command.

```
Device# show platform condition
Conditional Debug Global State: Stop
Conditions Direction
------------------------------------------------------------------------------------|--------
MAC Address 0024.D7C7.0054 N/A
Feature Condition Type Value
----------------------|----------------------|--------------------------------
```

This is an output example of the **show debug** command.

```
Device# show debug
IOSXE Conditional Debug Configs:
Conditional Debug Global State: Start
Conditions Direction
------------------------------------------------------------------------------------|--------
MAC Address 0024.D7C7.0054 N/A
Feature Condition Type Value
----------------------|----------------------|--------------------------------
Packet Infra debugs:
Ip Address Port
------------------------------------------------------|----------
```

Verify conditional debugging

Verify conditional debugging.

The table shown below lists the various commands that can be used to verify conditional debugging:

Command	Purpose
show platform condition	Displays the current conditions set.
show debug	Displays the current debug conditions set.
show platform software trace filter-binary	Displays logs merged from the latest tracefile.
request platform software trace filter-binary	Displays historical logs of merged tracefiles on the system.

Example: verify radioactive tracing log for SISF

Verify radioactive tracing log for SISF example.

This is an output example of the **show platform software trace message ios chassis active R0 | inc sisf** command.

```
Device# show platform software trace message ios chassis active R0 | inc sisf
2017/10/26 13:46:22.104 {IOSRP_R0-0}{1}: [parser]: [5437]:  UUID: 0, ra: 0
(note):  CMD: 'show platform software trace message ios switch active R0 | inc
 sisf' 13:46:22 UTC Thu Oct 26 2017
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):   FF8E802918 semaphore system unlocked
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):  Unlocking, count is now 0
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):   FF8E802918 semaphore system unlocked
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):  Unlocking, count is now 1
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Setting State to 2
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Start timer 0
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):  Gi1/0/5 vlan 10 aaaa.bbbb.cccc  Timer value/granularity for
0 :299998/1000
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Updated Mac Timer : 299998
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Before Timer :  350000
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Timer 0, default value is
350000
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):   Allocating timer wheel for 0
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):  Gi1/0/5 vlan 10 aaaa.bbbb.cccc No timer running
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):  Granularity for timer MAC_T1  is 1000
2017/10/26 13:46:10.667 {IOSRP_R0-0}{1}: [sisf]: [5437]:  UUID: 4800000000060,
 ra: 7 (debug):  Gi1/0/5 vlan 10 aaaa.bbbb.cccc Current State :MAC-STALE, Req
 Timer : MAC_T1  Current Timer MAC_T1
```

Packet tracing

Introduces packet tracing for Cisco Catalyst 9800 Series Wireless Controller for Cloud, detailing how to enable and use conditional debugging, apply per-AP and per-Client tracing, and explaining limitations of filtering with MAC or IP headers in CAPWAP-encapsulated packets.

A packet tracing is a network diagnostic tool that

- tracks the flow and processing of individual data packets through a network device
- provides detailed per-packet information based on user-defined conditions for troubleshooting purposes, and
- helps identify network issues such as misconfiguration, capacity overload, and software bugs.

The packet tracing feature explains how to perform data plane packet tracing for Cisco Catalyst 9800 Series Wireless Controller for Cloud software.

To trace packets on the controller:

1. Enable conditional debugging on selected packets or traffic to trace.
2. Enable packet tracing (per-AP or per-Client).

> **Note**
>
> Use per-AP conditional debugging with a MAC address filter when the AP and controllers are in the same VLAN. If they are in different VLANs, per-AP packet tracing with a MAC address does not capture packets because the MAC address varies.

Limitation of Conditional Debugging Packet Tracing

The MAC or IP filter applies only to the outer Ethernet or IP header. If a packet is CAPWAP encapsulated, the MAC or IP filter does not affect the inner 802.11 MAC or IP header.

Configure conditional debugging packet tracing (CLI)

Configure conditional debugging packet tracing using commands.

Enable and manage in-depth packet tracing for troubleshooting and diagnostics while minimizing resource usage using commands.

Procedure

1. Enable the privileged **EXEC** mode.

   Example

   ```
   Device# enable
   ```

   Enter your password, if prompted.

2. Configure packet tracing to capture the last set of packets.

   Example

   ```
   Device# debug platform packet-trace packet 8192 circular fia-trace data-size
    2048
   ```

   packet-count: Here, the valid range is from 16 to 8192.

   data-size: Here, the valid range is from 2048 to 16384 bytes.

3. Configure packet tracing for a copy of packet data.

   Example

   ```
   Device# debug platform packet-trace copy packet both size 2048
   ```

   packet-size: Here, the valid range is from 16 to 2048 bytes.

4. Enable conditional debugging for an interface, MAC, or IP filter.

Example

Enable conditional debugging for TenGigabitEthernet 0/0/0 and match packets whose source and destination MAC is 192.0.2.1:

```
Device# debug platform condition interface {interface-name TenGigabitEthernet
 | cpp} 0/0/0 {mac | ipv4 | match} 192.0.2.1 {both | ingress | egress}
```

An interface refers to any physical port, port channel, internal vlan, SVI, or wireless client.

5. Start conditional debugging packet tracing.

Example

```
Device# debug platform condition start
```

6. Stop conditional debugging packet tracing.

Example

```
Device# debug platform condition stop
```

7. Redirect all traced packets to bootflash.

Example

```
Device# show platform hardware chassis active qfp feature packet-trace packet
 all | redirect bootflash:packet_trace.txt
```

Convert the packet_trace.txt to pcap and download the pcap files. You can do so using this link:

http://wwwin-dharton-dev.cisco.com/pactrac2pcap.html

Configure conditional debugging packet tracing per AP (CLI)

Configure conditional debugging packet tracing per AP.

Enable conditional debugging to trace CAPWAP packets for a specific AP by filtering with MAC or IP address using commands.

Procedure

1. Enable privileged **EXEC** mode.

Example

```
Device# enable
```

Enter your password, if prompted.

2. Enable conditional debugging with MAC filter.

Example

```
Device# debug platform condition interface {interface-name TenGigabitEthernet
 | cpp} 0/0/0 {mac 192.0.2.1 | access-list acl-name | ipv4 | match} {both |
ingress | egress}
```

Here, the CLI matches the packets whose source or destination MAC address is 192.0.2.1

3. Enable conditional debugging with inline MAC ACL.

Example

```
Device# debug platform condition interface TenGigabitEthernet 0/0/0 match mac
 { H.H.H. | any | host} 192.0.2.1 {both | ingress | egress}
```

4. Enable conditional debugging with IP filter.

Example

```
Device# debug platform condition interface TenGigabitEthernet 0/0/0 ipv4
{192.0.2.1 | access-list ip-acl-name| both | egress | ingress} {both | egress
 | ingress}
```

0/0/0: is the GigabitEthernet interface number. The valid range is from 1 to 32.

Configure conditional debugging packet tracing per client (GUI)

Configure conditional debugging packet tracing per client using GUI.

Enable targeted packet tracing for individual clients to facilitate in-depth troubleshooting and analysis using GUI.

Procedure

1. Choose **Troubleshooting** > **Radioactive Trace**.
2. Click **Add**.
3. In the Add MAC/IP Address window, enter the **MAC/IP Address**.
4. Click **Apply to Device**.

Configure conditional debugging packet tracing per client (CLI)

Configure conditional debugging packet tracing using commands.

Enable conditional debugging to trace packets for a specific wireless client, helping to diagnose network issues more efficiently using commands.

Procedure

1. Enable privileged **EXEC** mode.

Example

```
Device# enable
```

Enter your password, if prompted.

2. Enable conditional debugging for a wireless client interface.

Example

```
Device# debug platform condition interface {intf-name | cpp 0xa0000001} { mac
 | ipv4 | match [ipv4 | ipv6 | mac]} protocol icmp host 192.0.2.1 host
192.0.2.1 {both | ingress | egress}
```

cpp-handle-index: The valid range is from 1 to 4294967295.

Verify conditional debugging packet tracing configuration

Verify conditional debugging packet tracing configuration.

To view the summary of the traced packet, use this command:

```
Device# show platform packet-trace summary
```

To view a specific traced packet, use this command:

```
Device# show platform packet-trace packet <packet number>
```

To view the wireless client interface handle, use this command:

```
Device# show platform hardware chassis active qfp feature wireless wlclient
cpp-client mac-address client-mac details
Device# show platform hardware chassis active qfp feature wireless wlclient
cpp-client mac-address 8825.93b0.b51f details
Client Details for client cpp_if_handle: 0x34
Name : WLCLIENT-IF-0x00a0000001
Mac Addr : 8825.93b0.b51f
pal_if_handle : 0xa0000001
Mobility State : LOCAL
Multicast Action : FORWARD
Auth State : RUN
```

Wireless client debug bundle

Introduces the wireless client debug bundle feature, detailing how it streamlines troubleshooting by collating debug logs, packet captures, show command outputs, and AP logs into a single file, and highlighting its release history and usage considerations.

A wireless client debug bundle is a troubleshooting feature that

- collates client-specific debug logs, packet captures, and relevant show command outputs
- packages them into a single compressed tar file for easier analysis, and
- streamlines client issue investigation by reducing the need to run multiple commands.

In earlier releases, logs were collected through various individual steps and commands. Now, the output of **show** commands related to clients is collected in a tar file through a single **debug** command. From Cisco IOS XE Cupertino 17.11.1, client debug bundle collates AP logs along with the existing controller bundle.

> **Note**
>
> The client debug bundle is not supported on HA with Stateful Switch Over (SSO).

> **Note**
>
> When you enable the **all** command option on the AP console, the command activates debug logging for all clients. This action can result in an excessive amount of logs being printed to the console.

Feature history for wireless client debug bundle

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 96: Feature History for client debug bundle

Feature Name	Release Information	Feature Description
Wireless client debug bundle	Cisco IOS XE 17.11.1	Client debug bundle includes AP logs along with the existing controller bundle, collected in a tar file through a single **debug** command.

Types of logs collected

Types of logs collected details.

A log type is a diagnostic data category that

- groups related log files generated during client device troubleshooting
- distinguishes between control plane packet captures, AP logs, and client-specific traces, and
- determines the format and file structure used to collect and organize system diagnostics.

The client debug bundle logs are collected in tar file format on bootflash by using a single **debug** command.

This example displays the file formats of logs collected from a client device with MAC address 8cXX.90XX.fdXX:

Example :

The final tar file that is generated is wireless_bundle_123456_UTC_Oct_20_2022.tar.

These files are extracted from the wireless_bundle_123456_UTC_Oct_20_2022.tar file:

- wireless_bundle_8cXX.90XX.fdXX.tar (client radioactive trace debug log)

- epc_135790_UTC_Oct_20_2022.pcap (packet capture in a control plane)
- ap_3802_cisco_client_bundle.17.11.0.61.20221020.135154.tgz (AP logs)

These files are extracted from wireless_bundle_8cXX.90XX.fdXX.tar client radioactive trace debug log file:

- show_tech_support_wireless_client_before_RA_start_8cXX.90XX.fdXX_134941_UTC_Oct_20_2022.txt
- ra_trace_8cXX.90XX.fdXX_135055_UTC_Oct_20_2022.log
- ra_trace_internal_8cXX.90XX.fdXX_135057_UTC_Oct_20_2022.log
- show_tech_support_wireless_client_after_RA_stop_8cXX.90XX.fdXX_135055_UTC_Oct_20_2022.txt

These files are extracted from ap_3802_cisco_client_bundle.17.11.0.61.20221020.135154.tgz AP log file:

- ap_3802_cisco_client_bundle.17.11.0.61.20221020.135154.messages
- ap_3802_cisco_client_bundle.17.11.0.61.20221020.135154.syslogs
- ap_3802_cisco_client_bundle.17.11.0.61.20221020.135154.tech_cdb_0
- ap_3802_cisco_client_bundle.17.11.0.61.20221020.135154.tech_cdb_1

Collect wireless client debug bundle (CLI)

Collect wireless client debug bundle using commands.

Collect wireless client debug information for specified clients for troubleshooting using commands.

Procedure

1. Add client MAC addresses for which debug logs are required.

   Example

   ```
   Device# debug wireless bundle client mac aaaa.bbbb.cccc
   ```

   You can add up to 32 client MAC addresses to the command. To delete the MAC addresses, run the **no** form of this command.

2. Start the collection of the client debug bundle for wireless clients.

   Example

   ```
   Device# debug wireless bundle client start
   ```

3. Optional: Enable the AP archive collection on a site tag.

   Example

   ```
   Device# debug wireless bundle client start ap-archive site-tag default-site-tag
    level { critical | debug | error | verbose}
   ```

   This specifies the AP archive levels as well.

4. Optional: Enable embedded packet capture (EPC) in a control plane.

Example

```
Device# debug wireless bundle client start epc
```

> **Note**
>
> If EPC is already enabled and active from a different source, you cannot start the debug bundle with EPC. To use EPC with the debug bundle, stop EPC if it is enabled from a different source, and then restart it with the debug bundle.

5. Optional: Configure the maximum time, in minutes, to trace the condition.

   Example

   ```
   Device# debug wireless bundle client start monitor-time 30
   ```

   The default time is 30 minutes.

6. Optional: Stop the collection of the debug bundle for wireless clients.

   Example

   ```
   Device# debug wireless bundle client stop-all collect {all | mac H.H.H}
   ```

7. Optional: Cancel the collection of the debug bundle for wireless clients.

   Example

   ```
   Device# debug wireless bundle client abort
   ```

91 Aggressive Client Load Balancing

Topics:

- Aggressive client load balancing

Aggressive client load balancing

Introduces aggressive client load balancing, detailing how wireless clients are distributed across lightweight APs, client association redirection from congested APs, and optimized network performance through AP utilization management.

Aggressive client load balancing is a wireless network feature that

- distributes wireless clients across lightweight APs managed by the same controller
- redirects client association attempts from overutilized APs to less busy APs using standardized response codes, and
- helps optimize overall network performance by preventing AP overload.

Guidelines of aggressive client load balancing

When a wireless client attempts to associate to a lightweight AP, the associated response packets are sent to a client with an 802.11 response packet including status code 17. This code 17 indicates that the corresponding AP is busy. The AP does not respond with 'success' if the AP threshold is not met. If the AP utilization threshold is exceeded and another less busy AP hears the client request, the AP responds with code 17 (AP busy).

If AP1 has more clients than AP2 during the load-balancing window, AP1 is considered busier. When a client attempts to associate to AP1, the client receives an 802.11 response packet with status code 17, indicating that the AP is busy, and the client attempts to associate to a different AP.

Configure the controller to deny client associations up to 10 times. If a client tries to connect 11 times, it will connect on the 11th attempt. Enable or disable load balancing on a particular WLAN. This option is helpful if you need to exclude time-sensitive voice clients from load balancing.

Note

For a FlexConnect AP, the association is locally handled. The controller makes load-balancing decisions. A FlexConnect AP sends an initial response to the client before knowing the result of the calculations in the controller. Load balancing does not work when the FlexConnect AP is in standalone mode.

In this mode, the FlexConnect AP does not send a (re)association response with status 17 for load balancing. Instead, it first sends a (re)association response with status zero (success), and then sends a deauthentication with reason five.

Restrictions of aggressive client load balancing

- This feature is not supported on the APs joined on default-site-tag.
- This feature is not supported on the APs across different named site-tags.
- This feature is supported only on the APs within a named-site-tag.

Enable aggressive client load balancing (GUI)

Enable aggressive client load balancing using GUI.

Ensure clients are evenly distributed across APs to maximize WLAN performance using GUI.

Procedure

1. Choose **Configuration** > **Wireless** > **WLANs** > **Wireless Networks**.

2. Select a **WLAN** to view the **Edit WLAN** window.
3. Click on the **Advanced** tab.
4. Check the **Load Balance** check box to enable the feature.
5. Click **Update & Apply to Device**.

Configure aggressive client load balancing (GUI)

Configure Aggressive Client Load Balancing using GUI.

Set the number of clients and denial count for aggressive load balancing to optimize wireless network performance using GUI.

Procedure

1. Choose **Configuration** > **Wireless** > **Advanced**.

   The **Load Balancing** window appears.
2. In the **Aggressive Load Balancing Window (clients)** field, enter the number of clients for the aggressive load balancing client window.
3. In the **Aggressive Load Balancing Denial Count** field, enter the load balancing denial count.
4. Click **Apply**.

Configure aggressive client load balancing (CLI)

Configure aggressive client load balancing using commands.

Configure aggressive client load balancing to optimize WLAN performance and distribute clients across APs using commands.

Procedure

1. Enter the global configuration mode and specify the WLAN name.

   Example

   ```
   Device# configure terminal
   ```

   ```
   Device(config)# wlan test-wlan
   ```
2. Disable the WLAN.

   Example

   ```
   Device(config-wlan)# shutdown
   ```
3. Configure a guest controller as mobility controller, in order to enable client load balance to a particular WLAN.

Example

```
Device(config-wlan)# load-balance
```

Configure the WLAN security settings as the WLAN requirements.

4. Enable WLAN.

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

5. Return to the privileged EXEC mode and enter the global configuration mode.

   Example

   ```
   Device(config)# end
   ```

   ```
   Device(config)# configure terminal
   ```

6. Configure the load balancing denial count.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} load-balancing denial 10
   ```

7. Configure the number of clients for the aggressive load balancing client window.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} load-balancing window 10
   ```

8. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-wlan)# end
   ```

9. Display a filtered section of the current configuration.

   Example

   ```
   Device# show running-config | section test-wlan
   ```

92 RF based Automatic AP Load Balancing

Topics:

- RF based automatic AP load balancing

RF based automatic AP load balancing

Introduces RF based automatic AP load balancing, detailing its operation using RRM neighbor report-based clustering, supported platforms and APs, prerequisites, use cases, feature history, and related commands for optimizing AP distribution and resource usage across WNCd instances.

RF based automatic AP load balancing is a wireless network load management feature that

- groups APs dynamically using RRM neighbor report-based AP grouping
- balances AP clusters across WNCd instances to optimize resource usage, and
- overrides site tag-based load balancing to ensure even AP distribution and prevent overload.

The RF based Automatic AP Load Balancing feature builds on the existing site tag-based load balancing feature. In the existing method, APs are balanced by assigning them to wireless network control daemons (WNCd) based on site tags. If the number of APs in a named site tag exceeds the capacity of a WNCd instance, uneven AP distribution may occur, which can result in high memory and CPU usage.

Limiting the number of APs in a site tag to 1000 using the **load** command may not prevent uneven distribution if the AP load limit is not correctly configured. In some cases, APs assigned to a site tag may not be colocated.

When you enable this feature, it forms AP clusters based on RSSI values from AP neighbor reports. The clusters are divided into sub-neighborhoods and smaller areas, which are then distributed evenly across WNCd processes. AP load balancing becomes active after you reboot the controller or perform an AP CAPWAP reset using the **ap neighborhood load-balance apply** command. When active, RF based Automatic AP Load Balancing overrides site tag-based load balancing.

Supported platforms

- Cisco Catalyst 9800-80 Wireless Controller
- Cisco Catalyst 9800-40 Wireless Controller
- Cisco Catalyst 9800 Wireless Controller for Cloud
- Catalyst 9800 Embedded Wireless Controller for a Cisco switch

Prerequisites for RF based automatic AP load balancing

Before enabling RF-based automatic AP load balancing, make sure your network is stable and all APs are fully deployed with enough time to discover all RF neighbors.

Use cases for RF based automatic AP load balancing

- You can use a single site tag for all deployed APs.
- If you attach more APs to a named-site tag than the available capacity of the WNCds, this feature provides better load balancing across WNCd instances.
- Use this feature in environments with many client intra-WNCd roaming scenarios. For instance, if you manage APs in two separate buildings on a campus, assign all APs for each building to a single WNCd rather than splitting them across multiple WNCds.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 97: Feature history for RF based automatic AP load balancing

Feature Name	Release Information	Feature Description
RF based automatic AP load balancing	Cisco IOS XE 17.12.1	The RF based Automatic AP Load Balancing feature uses Radio Resource Management (RRM) neighbor report-based AP grouping and load-balancing across WNCd instances. These commands are introduced: • apneighborhood load-balance • apneighborhood calendar-profile • wirelessload-balance ap method rf • showap neighborhood summary • showap neighborhood details • showap neighborhood • showap neighborhood mac details • showap neighborhood wncd

Restrictions and guidelines for RF based automatic AP load balancing

Restrictions for RF based automatic AP load balancing

- You cannot use the same calendar profile for AP neighborhood policy or AP profile.
- This feature is supported only on APs in local, fabric, and FlexConnect mode.
- You cannot run the feature when the overall load on the system is high.
- You cannot use the output of the **show wireless loadbalance tag affinity** command when the RF based automatic AP load balancing feature is enabled.

Guidelines for RF based Automatic AP Load Balancing

- For a new deployment, use the site tags and follow the current site tag recommendations to evenly distribute the APs, or use the **site tag load** command to automatically distribute the APs. Using site tags ensures that all APs with the same site tag are assigned to the same WNCd. This arrangement simplifies both troubleshooting and intra-WNCd roaming.
- If you cannot use site tags because grouping APs is not possible or you wish to avoid additional design work, select either the default site tag or any named site tag and enable the RF-based automatic AP load balancing feature.
- In an existing deployment, if high CPU usage results from an unbalanced system, use the auto RRM load balance system instead of redesigning site tags.
- If no CPU load issues exist, even though the system is unbalanced, maintain your current configuration.

Configure RF based automatic AP load balancing

Configure RF based automatic AP load balancing.

Achieve balanced AP distribution across wireless controllers using RF metrics through commands.

Before you begin

The enablement of the RF-based load-balancing algorithm involves two phases.

1. In the first phase, the running of the algorithm, the RF-based automatic AP load balancing feature can be scheduled to run when the calendar profile start time expires using the **ap neighborhood calendar-profile** command, or it can be started on-demand using the **ap neighborhood load-balance start** command. You can schedule the calendar profile start timer to run daily, weekly, or monthly.

   Note

   For more information on configuring Calendar profile, check Configure a calendar profile (GUI) on page 408 and Configure a calendar profile (CLI) on page 409.

2. To apply the algorithm, reload the controller or use the **ap neighborhood load-balance apply** command after the wireless load-balance ap method rf configuration is enabled.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP neighborhood calendar profile.

   Example

   ```
   Device(config)# ap neighborhood calendar-profile ap-calendar-profile
   ```

   Note

   After setting the calendar profile, you can choose to skip Step 4. If you want to perform load balancing immediately, complete Step 4.

3. Return to the privileged EXEC mode.

   Example

   ```
   Device(config)# exit
   ```

4. Optional: Start AP neighborhood load-balance algorithm calculation and WNCd allocation.

Example

```
Device# ap neighborhood load-balance start
```

5. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

6. Configure RF-based AP load balancing.

   Example

   ```
   Device(config)# wireless load-balance ap method rf
   ```

7. Return to the privileged EXEC mode.

   Example

   ```
   Device(config)# exit
   ```

8. Run on-demand RRM-based AP load balancing.

   Example

   ```
   Device# ap neighborhood load-balance apply
   ```

   This command rebalances the APs by performing a CAPWAP reset. If an AP is already assigned to the correct WNCd instance, it is not reset. You cannot execute this command if the RRM-based AP load balancing algorithm is running or if algorithm results are unavailable.

Disable RF based automatic AP load balancing (CLI)

Disable RF based Automatic AP Load Balancing.

Disable the RF-based automatic AP load balancing feature to prevent APs from using RF data to evenly distribute clients and APs using commands.

Before you begin

The APs may remain load balanced based on algorithm data, even after you disable all feature configurations and clear all algorithm outputs. To rebalance all APs using the default method of site tags, reload the controller or perform a CAPWAP reset on all APs.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Disable RF-based AP load balancing.

   Example

   ```
   Device(config)# no wireless load-balance ap method rf
   ```

3. Disable the AP neighborhood calendar profile.

   Example

   ```
   Device(config)# no ap neighborhood calendar-profile ap-calendar-profile
   ```

4. Return to the privileged EXEC mode.

   Example

   ```
   Device(config)# exit
   ```

5. Clear the AP neighborhood load-balance algorithm calculation and resource allocation.

   Example

   ```
   Device# ap neighborhood load-balance clear
   ```

Verify automatic WNCd load balancing

Verify automatic WNCd load balancing.

To verify the results of the RF-based algorithm and the result of the related load balancing, use these**show** commands.

To view the AP neighborhood summary, use this **show** command:

```
Device# show ap neighborhood summary
NH-ID = Neighborhood ID
AREA-ID = Area ID of neighborhood
Total number of neighborhood: 5
Total number of algorithm iterations: 1
Ideal AP capacity per WNCD instance: 210
Total number of neighborhood area: 14
NH-ID      AREA-ID         Process Name   Number of APs
-------------------------------------------------
0          0x00000000      WNCD_1         1
1          0x00000000      WNCD_0         2
2          0x00000000      WNCD_0         100
2          0x00000001      WNCD_0         100
2          0x00000002      WNCD_3         100
2          0x00000003      WNCD_4         50
3          0x00000000      WNCD_1         100
3          0x00000001      WNCD_1         100
3          0x00000002      WNCD_4         100
3          0x00000003      WNCD_4         50
4          0x00000000      WNCD_2         100
4          0x00000001      WNCD_2         100
4          0x00000002      WNCD_3         100
4          0x00000003      WNCD_3         50
```

To view the AP neighborhood details, use this **show** command:

```
Device# show ap neighborhood details
NH-ID = Neighborhood ID
AREA-ID = Area ID of neighborhood
Number of APs: 4
NH-ID   AREA-ID        WNCD instance    AP Name                 Ethernet MAC

-------------------------------------------------------------------------------
0       0x00000000     1               EDU_BR_01_00_28_3702    bc16.6509.bfcc

1       0x00000000     0               ci-glad-mdns-ap         0cd0.f894.567c

1       0x00000000     0               EDU_VW_9120_1_2         c4f7.d54c.f978

2       0x00000000     0               ewlc-hc-tsim-30-1       00b9.3000.02f0
```

To view the AP neighborhood information, use this **show** command:

```
Device# show ap neighborhood 0 details
NH-ID = Neighborhood ID
AREA-ID = Area ID of neighborhood
Number of APs: 1
NH-ID     AREA-ID          WNCD instance     AP Name              Ethernet MAC

-------------------------------------------------------------------------------
0         0x00000000       0               APA023.9FD8.EA22     a023.9fd8.ea22
```

To view the AP neighborhood details using its MAC address, use this **show** command:

```
Device# show ap neighborhood mac 0aa8.89f0.0001 details
NH-ID = Neighborhood ID
AREA-ID = Area ID of neighborhood
AP Name            Ethernet MAC    Radio MAC       NH-ID  AREA-ID     WNCD
instance
-------------------------------------------------------------------------------
AP6B8B4567-0001    0aa8.89f0.0001  0aa8.8900.0100  0      0x00000000  0
```

To view the WNCd information, use this **show** command:

```
Device# show ap neighborhood wncd 0 details
NH-ID = Neighborhood ID
AREA-ID = Area ID of neighborhood
Number of APs: 9
WNCD instance     NH-ID       AREA-ID          AP Name          Ethernet MAC

-------------------------------------------------------------------------------
0                 2           0x00000000       9130I-1          0c75.bdb5.ffc0

0                 2           0x00000000       9130E-2          3c41.0efe.46f0

0                 2           0x00000000       9120E-2          5ce1.7628.8bbc

0                 2           0x00000000       9130I-2          e44e.2d2e.59d4

0                 2           0x00000000       9120E-1          5ce1.7628.aa0c

0                 2           0x00000000       9120E-3          5ce1.7628.af04
```

```
0                2           0x00000000       3700I-2          b838.6159.dfa4

1                0           0x00000000       3800I-2          6cb2.ae2e.dfdc

2                1           0x00000000       4800-1           f4db.e643.fa72

NH-ID = Neighborhood ID
AREA-ID = Area ID of neighborhood
Number of APs: 5
WNCD instance    NH-ID   AREA-ID       AP Name             Ethernet MAC
-----------------------------------------------------------------------
0                12      0x00000000    AP6B8B4567-0001     0aa8.89f0.0001
0                12      0x00000000    AP6B8B4567-0004     0aa8.89f0.0004
0                12      0x00000000    AP6B8B4567-0007     0aa8.89f0.0007
0                12      0x00000000    AP6B8B4567-0010     0aa8.89f0.000a
0                12      0x00000000    AP6B8B4567-0013     0aa8.89f0.000d
```

93 Accounting Identity List

Topics:

- Configure accounting identity list (GUI)

Configure accounting identity list (GUI)

Configure accounting identity list using GUI.

Create a method list for accounting identity using the GUI to control and monitor user access to your network.

Procedure

1. Choose **Configuration** > **Security** > **AAA**.
2. In the **AAA Method List** tab, go to the **Accounting** section, and click **Add**.
3. In the **Quick Setup: AAA Accounting** window that is displayed, enter a name for your method list.
4. Select the type of authentication as identity, in the **Type** drop-down list.
5. Choose the server groups you want to use to authenticate access to your network, from the **Available Server Groups** list and click **>** icon to move them to the **Assigned Server Groups** list.
6. Click **Save & Apply to Device**.

Configure accounting identity list (CLI)

Configure accounting identity list using commands.

Enable RADIUS accounting with an identity list to log user actions and network usage events using commands.

When a user successfully executes an action, the RADIUS accounting server logs the changed attributes, user ID, remote host, date and time of execution, authorization level, description of the action performed, and values provided.

Follow the procedure given below to configure accounting identity list.

Before you begin

Configure the RADIUS server and AAA group server.

Procedure

Enable accounting to send a start-record accounting notice when a client is authorized and a stop-record at the end.

Example

```
Device(config)# aaa accounting identity user1 start-stop group aaa-test
```

Note

You can also use the default list, instead of a named list.

Configure client accounting (GUI)

Configure client accounting using GUI.

Enable client accounting on a specified policy profile through the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click the **Policy Profile Name**. In the **Edit Policy Profile** window, go to the **Advanced** tab.
3. From the **Accounting List** drop-down, select the appropriate accounting list for this policy profile. The policy profile uses the accounting type you selected before gaining access to the network.
4. Click **Save & Apply to Device**.

Configure client accounting (CLI)

Configure client accounting using commands.

Enable client accounting on a WLAN policy profile to track user activity and support auditing with an accounting server using commands.

Procedure

1. Configure WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy default-policy-profile
   ```

2. Disable the policy profile.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

3. Set the accounting list.

   Example

   ```
   Device(config-wireless-policy)# accounting-list user1
   ```

4. Enable the policy profile.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

94 Accounting Session ID

Topics:

- Accounting session ID

Accounting session ID

Introduces the accounting session ID for wireless client sessions, explains its generation and role in AAA, and outlines its feature history, including changes in support across Cisco IOS XE releases.

An accounting session ID is a wireless client session identifier that

- is unique for each client session
- identifies the accounting data for a wireless client in the AAA server, and
- is generated by the AAA module during session establishment.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 98: Feature history for Accounting session ID

Feature Name	Release Information	Feature Description
Accounting session ID	Cisco IOS XE 17.4.1	With this feature, accounting session ID is supported in the AAA access request, while authenticating wireless client using IEEE 802.1x method. From this release, the accounting session ID is sent as part of the access request too.
Accounting session ID	Cisco IOS XE 17.3.x	In the Cisco IOS XE release 17.3.x and earlier releases, the accounting session ID was sent only as part of the accounting request.

Configure an accounting session ID (CLI)

Configure an accounting session ID using commands.

Set up an accounting session ID on your device to track accounting sessions and include session identification attributes in RADIUS packets using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Send the RADIUS authentication attribute 44, in the access request packet.

Example

```
Device(config)# radius-server attribute wireless 44 include-in-access-req
```

3. Configure the accounting session identity of the AAA server.

Example

```
Device(config)# aaa accounting identity accounting-list-name start-stop group
 AAA_GROUP_1
```

4. Configure the WLAN policy profile.

Example

```
Device(config)# wireless profile policy default-policy-profile
accounting-list-name start-stop group AAA_GROUP_1
```

5. Configure the accounting list.

Example

```
Device(config-wireless-policy)# accounting-list accounting-list-name
```

> **Note**
>
> The accounting session ID is included in the account request only when the **radius-server attribute wireless 44 include-in-access-req** command is enabled and the accounting configuration is set under the wireless policy.

6. Add a description for the policy profile.

Example

```
Device(config-wireless-policy)# description accounting-description
```

7. Configure the VLAN name or ID.

Example

```
Device(config-wireless-policy)# vlan 40
```

8. Save the configuration, exit configuration mode and return to privileged EXEC mode.

Example

```
Device(config-wireless-policy)# no shutdown
```

Verify an account session ID

Verify an account session ID.

To verify if an account session ID is populated, use this command:

```
Device# show wireless pmk-cache

Number of PMK caches in total : 1
Type      Station            Entry Lifetime  VLAN Override        IP Override
         Accounting-Session-Id   Audit-Session-Id              Username
-----------------------------------------------------------------------------------------------------------
RSN       6c19.c0e6.a444      1768            NA
           0x00000006              052DA8C1000000104E634C77      cwa-user
```

To display the current Accounting Session ID, use this command:

```
Device# show wireless client mac-address <H.H.H> detail

Central NAT : DISABLED
Session Manager:
  Point of Attachment :CAPWAP_90000005
  IIF ID             : 0x90000005
  Authorized         : TRUE
  Session timeout    : 1800
  Common Session ID: 000000000000000B14E9130A
  Acct Session ID  : 0x0000000c
  Last Tried Aaa Server Details:
        Server IP : 192.0.2.1
  Auth Method Status List
        Method : Dot1x
                SM State         : AUTHENTICATED
                SM Bend State    : IDLE
  Local Policies:
        Service Template : wlan_svc_default-policy-profile (priority 254)
                VLAN             : 1
  Server Policies:
                Absolute-Timer   : 1800
  Resultant Policies:
                VLAN Name        : default
                VLAN             : 1
                Absolute-Timer   : 1800
```

95 Interim Accounting

Topics:

- Interim accounting

Interim accounting

Outlines interim accounting update mechanisms in RADIUS, including event-triggered and periodic updates, session parameter reporting, reasons to disable the feature, and feature history with release details for Cisco IOS XE Cupertino 17.9.1.

An interim accounting update is a RADIUS accounting message that

- reports changes to key session parameters such as DHCP, HTTP, IP, or roaming state
- is sent by the network access server (NAS) to the RADIUS server between the start-stop accounting messages, and
- can be triggered by specific events or at periodic intervals.

Note

The RADIUS accounting requests send data, such as VLAN ID, authentication methods, and so on, to a session.

RADIUS accounting covers these updates:

- Interim Updates: When RADIUS accounting covers DHCP_TLVs, HTTP_TLVs, IP, or ROAM (state machine) changes, the updates are referred to as interim updates or interim accounting.
- Periodic Updates: When RADIUS accounting is specified by a timer value, the updates are referred to as periodic updates.

Disable the interim updates for these reasons:

- These actions increase network congestion by sending unneeded traffic.
- They drive up error rates.
- They impact alarm thresholds and other metrics that are used for venue-authentication performance.

By default, the interim accounting feature is enabled with the **aaa accounting Identity default start-stop group radius** command.

Feature history for interim accounting

Disable interim accounting (CLI)

Disable interim accounting using commands.

Disable interim accounting on a WLAN policy profile using commands.

Before you begin

You must disable the Policy Profile before performing this procedure.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure WLAN policy profile and enter wireless policy configuration mode.

Example

```
Device(config)# wireless profile policy default-policy-profile
```

3. Disable interim accounting.

Example

```
Device(config-wireless-policy)# no accounting-interim
```

4. Return to the privileged EXEC mode.

Example

```
Device(config-wireless-policy)# end
```

Verify interim accounting

Verify interim accounting.

To verify the interim accounting updates, run this command:

```
Device# show wireless profile policy detailed default-policy-profile | s Interim
Interim Accounting Updates: DISABLED
```

96 Wireless Multicast

Topics:

- Wireless multicast
- IPv6 snooping
- IPv6 multicast over multicast
- Directed multicast service
- Restrictions for wireless broadcast, non-IP multicast and multicast VLAN
- Multicast filtering

Wrapper for Wireless Multicast chapter in vEWLC config guide.

Wireless multicast

Outlines wireless multicast communication methods and modes on controllers, including unicast, multicast, and FlexConnect operation, and details IGMP snooping, MGID assignment, and multicast traffic optimization for APs and clients.

A wireless multicast is a wireless network communication method that

- allows simultaneous transmission of data from a controller to multiple APs and clients
- operates in different modes (unicast and multicast) to accommodate network capabilities and efficiency, and
- leverages features such as IGMP snooping and MGID assignment to optimize multicast traffic delivery.

If the network supports packet multicasting, the multicast method that the controller uses can be configured. The controller performs multicast routing in two modes:

- Unicast mode: The controller unicasts every multicast packet to every AP associated with the controller. Although this mode is inefficient and generates significant extra traffic for the device and the network, it is required on networks that do not support multicast routing. This need arises if the APs are on different subnets than the device's wireless management interface.
- Multicast mode: The controller sends multicast packets to a CAPWAP multicast group. This method reduces the overhead on the controller processor by shifting the work of packet replication to the network. This approach is much more efficient than the unicast method.

The FlexConnect mode has two submodes: local switching and central switching. In local switching mode, data traffic is switched at the AP level and the controller does not process any multicast traffic. In central switching mode, multicast traffic reaches the controller. IGMP snooping, however, takes place at the AP.

When multicast mode is enabled and the controller receives a multicast packet from the wired LAN, it encapsulates the packet using CAPWAP and forwards it to the CAPWAP multicast group address. The controller always uses the management VLAN to send multicast packets. APs in the multicast group receive these packets and forward them to all BSSIDs mapped to the VLAN that delivers multicast traffic to clients.

The controller supports all capabilities of IGMP v1, including multicast listener discovery (MLD) v1 snooping. However, IGMP v2 and IGMP v3 support is limited. This feature tracks and delivers IPv6 multicast flows to clients that request them. For IPv6 multicast, you must enable global multicast mode.

Internet group management protocol (IGMP)

IGMP snooping is a network switch feature that

- listens to IGMP network traffic between hosts and routers
- maintains a map of which devices are subscribed to which multicast groups, and
- forwards multicast traffic only to the relevant devices, reducing unnecessary data transmission.

When this feature is enabled, the controller gathers IGMP reports from clients, processes them, creates unique multicast group IDs (MGIDs) based on the Layer 3 multicast address and VLAN number, and sends the IGMP reports to the IGMP querier. The controller then updates the AP MGID table on the corresponding AP with the client MAC address.

When the controller receives multicast traffic for a particular multicast group, it forwards the traffic to all APs. However, only those APs with active clients subscribed to that multicast group send multicast traffic on the specified WLAN. IP packets are forwarded with an MGID unique to both the ingress VLAN and the destination multicast group. Layer 2 multicast packets are forwarded with an MGID unique to the ingress VLAN.

An MGID is a 14-bit value placed in the 16-bit reserved field of wireless information in the CAPWAP header. The remaining two bits must be set to zero.

Multicast optimization

This section provides multicast optimization details.

Multicast optimization enables you to create a multicast VLAN that can be used for multicast traffic. One of the VLANs in the device can be configured as a multicast VLAN where multicast groups are registered. Clients can listen to a multicast stream on the multicast VLAN. The system generates the MGID based on both the multicast VLAN and the multicast IP addresses.

- If multiple clients on different VLANs of the same WLAN listen to a single multicast IP address, the system generates a single MGID. The device forwards all multicast streams from clients on this VLAN group through the multicast VLAN. This approach ensures that the upstream router has one entry for all VLANs in the group.
- Only one multicast stream reaches the VLAN group, even when clients are on different VLANs. Therefore, the device sends out just one multicast stream over the network.

Note

When VLAN groups are defined and use multicast communication, you must enable the multicast VLAN.

IPv6 global policies

This section provides IPv6 global policies details.

- IPv6 global policies provide storage and access policy database services. IPv6 ND inspection and IPv6 RA guard are features provided by IPv6 global policies that you can enable. When you configure ND inspection globally, the policy attributes are stored in the software policy database. You can then apply the policy to an interface. The software policy database updates its entry to record the interface that you applied the policy to.
- The controller enables IPv6 RA guard by default. If Stateless Address Auto-Configuration (SLAAC) is deployed in your network, forward route advertisements (RA) from the wired side to wireless clients.

Prerequisites for configuring wireless multicast

This section describes the prerequisites for configuring wireless multicast.

IP multicast routing must be enabled and the PIM version and PIM mode must be configured. The default routes should be available in the device. After performing these tasks, the device can forward multicast packets and populate its multicast routing table. To configure multicast mode, the network should be enabled for multicast.

- Multicast hosts, routers, and multilayer switches must have IGMP enabled to participate in IP multicasting.
- When enabling multicast mode on the controller, a CAPWAP multicast group address should also be configured. APs use IGMP to listen to the CAPWAP multicast group.
- You must be cautious when using IGMPv3 with switches enabled for IGMP snooping. The IGMPv3 messages are different from the messages used in IGMP Version 1 (IGMPv1) and Version 2 (IGMPv2). If your switch does not recognize IGMPv3 messages, the hosts do not receive traffic when IGMPv3 is used.
- IGMPv3 devices do not receive multicast traffic if IGMP snooping is disabled or if IGMPv2 is configured on the interface.
- It is recommended to enable IGMPv3 on all intermediate or Layer 3 network devices, especially on each subnet used by multicast devices, such as controller and AP subnets.

Restrictions on configuring wireless multicast

This section describes configuring wireless multicast restrictions.

These are the restrictions for configuring IP multicast forwarding:

- APs in monitor mode, sniffer mode, or rogue-detector mode do not join the CAPWAP multicast group address.
- Each controller should be configured with a distinct CAPWAP multicast group.
- Do not enable multicast routing on the management interface.
- Multicast with VLAN groups is supported only in APs operating in local mode.
- The uplink switch should route multicast traffic from wireless clients within a VLAN that is not configured for multicast.
- The wireless multicast to unicast (MCUC) mode is only supported in 9800-CL small template.
- Cisco IOS XE 17.9.5, 17.12.2, 17.14, and later versions support multicast traffic on an AAA-overridden VLAN.
- Multicast-unicast is not supported on large-scale controller appliances, including CW9800L, CW9800M, CW9800H1, and CW9800H2.
- Multicast is not supported over mobility tunnels.

Configuring wireless multicast

This section describes the various wireless multicast configuration tasks.

These sections provide information about the various wireless multicast configuration tasks:

Configure wireless multicast-MCMC mode (CLI)

Configure wireless multicast-MCMC mode using commands.

Enable wireless multicast in multicast-over-multicast (MCMC) mode on your device using commands.

Procedure

1. Enable multicast-over-multicast.

   Example

   ```
   Device(config)# wireless multicast 224.0.0.1
   ```

   Use the **no** form of this command to disable the feature.

2. Exit the configuration mode.

   Example

   ```
   Device(config)# end
   ```

Configure wireless multicast-MCUC mode (CLI)

Configure wireless multicast-MCUC mode using commands.

Enable multicast traffic for wireless clients in MCUC mode using commands.

> **Note**
>
> The wireless multicast to unicast (MCUC) mode is only supported in 9800-CL small template.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable multicast traffic for wireless clients.

   Example

   ```
   Device(config)# wireless multicast
   ```

3. Exit the configuration mode.

   Example

   ```
   Device(config)# end
   ```

Configure multicast listener discovery snooping (GUI)

Configure multicast listener discovery snooping using GUI.

Enable or disable multicast listener discovery (MLD) snooping on specific VLANs to efficiently manage multicast network traffic using commands.

Procedure

1. Choose **Configuration** > **Services** > **Multicast**.
2. Click **MLD Snooping**.
3. In the **MLD Snooping** section, click the toggle button to enable or disable MLD snooping.
4. Enter the **MLD Query Interval** in milliseconds. The valid value range is between 100 ms and 32767 ms. The default value is 1000 ms.
5. Move the required VLAN IDs from the **Disabled** section to the **Enabled** section. By default, this feature is disabled on all VLANs.

   You can also search for a VLAN ID using the search field. Click **Disable All** to move all the VLAN IDs from the **Enabled** list to the **Disabled** list. Click **Enable All** to move all the VLAN IDs from the **Disabled** list to the **Enabled** list.

6. Click **Apply to Device**.

Configure IPv6 MLD snooping (CLI)

Configure IPv6 MLD snooping using commands.

Enable MLD snooping on your device to optimize IPv6 multicast traffic traversal using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# ipv6 mld snooping
   ```

2. Enable MLD snooping.

   Example

   ```
   Device(config)# ipv6 mld snooping
   ```

Verify the multicast VLAN configuration

Verify the multicast VLAN configuration.

To view the multicast VLAN associated with a policy profile along with the VLAN assigned to that profile, use this command:

```
Device# show wireless profile policy detail default-policy-profile
Policy Profile Name           : default-policy-profile
Description                   : default policy profile
Status                        : ENABLED
VLAN                          : vlan-pool1
                Multicast VLAN : 84
Client count                  : 0
Passive Client                : DISABLED
```

To view the multicast VLAN associated with a client, use this command:

```
Device# show wireless client mac ac2b.6e4b.551e detail
Client MAC Address : ac2b.6e4b.551e
Client IPv4 Address : 192.0.2.20
…….
VLAN : 82
Access VLAN : 82
                Multicast VLAN: 84
```

IPv6 snooping

Outlines the operation of IPv6 snooping and IPv6 neighbor discovery inspection, detailing mechanisms for learning and securing IPv6 address-to-MAC bindings, traffic analysis, vulnerability mitigation, hardware capture rules, and traffic classification to enhance Layer 2 security and scalability.

IPv6 snooping or IPv6 neighbor discovery inspection is a Layer 2 security feature that

- combines several IPv6 first-hop security mechanisms, including IPv6 Address Glean and IPv6 Device Tracking
- learns and secures IPv6 address-to-MAC bindings using neighbor discovery messages, and
- mitigates vulnerabilities such as attacks on duplicate address detection (DAD), address resolution, device discovery, and the neighbor cache in the neighbor discovery process by analyzing and filtering ND protocol traffic based on trusted binding tables

IPv6 neighbor discovery (ND) inspection operates at Layer 2, or between Layer 2 and Layer 3, providing IPv6 features with security and scalability.

IPv6 ND inspection learns and secures bindings for stateless autoconfiguration addresses in Layer 2 neighbor tables and analyzes ND messages to build a trusted binding table. IPv6 ND messages that do not have valid bindings are dropped. An ND message is considered trustworthy if its IPv6-to-MAC mapping can be verified.

When IPv6 ND inspection is configured on a target which can include device ports, switch ports, Layer 2 interfaces, Layer 3 interfaces, or VLANs (depending on platform support), the device downloads capture instructions to the hardware. These instructions redirect ND protocol and Dynamic Host Configuration Protocol (DHCP) for IPv6 traffic to the switch integrated security features (SISF) infrastructure in the routing device. For ND traffic, messages such as NS, NA, RS, RA, and REDIRECT are directed to SISF. For DHCP, UDP messages sourced from port 546 or 547 are also redirected.

IPv6 ND inspection registers its 'capture rules' with the classifier, which aggregates all rules from all features on a given target and installs the corresponding ACL into the platform-dependent modules. When redirected traffic is received, the classifier calls all entry points from registered features for the relevant target, including the IPv6 ND inspection entry point. The IPv6 ND inspection entry point is called last. If another feature makes a decision, such as to drop the traffic, that decision overrides the IPv6 ND inspection action.

IPv6 device tracking

This topic describes the IPv6 device tracking details.

- IPv6 device tracking provides IPv6 host liveness tracking.
- This initiates a neighbor table to immediately update when an IPv6 host disappears.

IPv6 first-hop security binding table

IPv6 First-Hop Security Binding Table details.

An IPv6 first-hop security binding table is a security database that

- enables the binding table to recover in the event of a device reboot populating from information sources such as neighbor discovery (ND) snooping and DHCP gleaning
- stores entries that map IPv6 neighbors to validate the link-layer addresses (LLA), IPv6 or IPv4 addresses, prefix binding, and
- is used by IPv6 guard features to validate bindings and prevent spoofing or redirect attacks.

The recovery mechanism will block any data traffic sourced from an unknown source; that is, a source not already specified in the binding table and previously learned through ND or DHCP gleaning. When the destination guard cannot resolve a destination address, this feature recovers the missing binding table entries. When a failure occurs, a binding table entry is recovered by querying the DHCP server or the destination host, depending on the configuration.

Recovery protocols and prefix lists

This section provides recovery protocols and prefix lists details.

The IPv6 first-hop security binding table recovery mechanism allows you to configure a prefix list. This list is matched before recovery is attempted for both DHCP and NDP.

- If an address does not match the prefix list associated with the protocol, recovery of the binding table entry with that protocol is not attempted. The prefix list must match the prefixes that are valid for address assignment in the Layer 2 domain for the given protocol.
- By default, no prefix list is configured, so recovery is attempted for all addresses. The command to associate a prefix list to a protocol is **protocol { dhcp | ndp} [prefix-list prefix-list-name]**.

IPv6 address glean

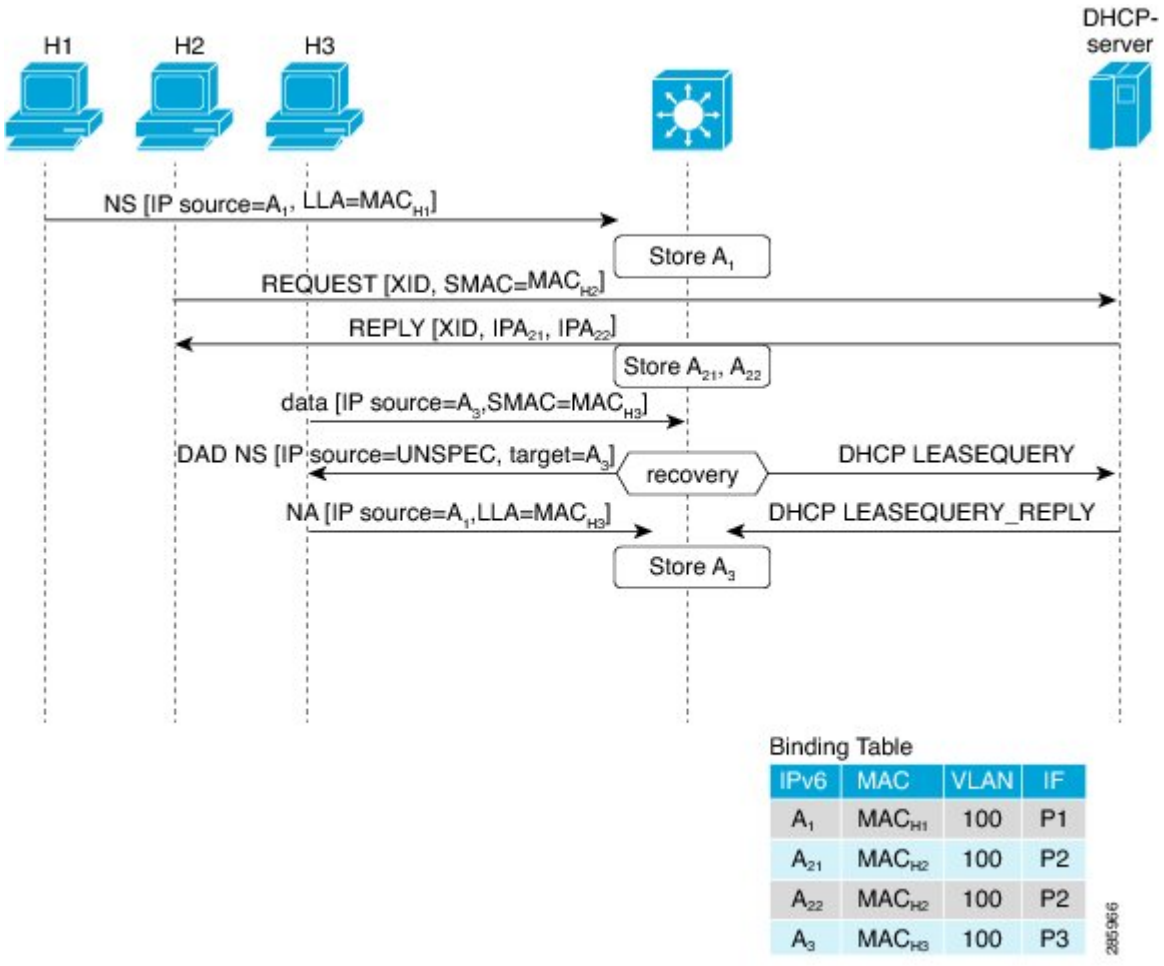
IPv6 address glean details.

An IPv6 address glean feature is an IPv6 security mechanism that

- inspects Neighbor Discovery (ND) and DHCP messages on a link to discover glean IPv6 addresses
- populates the binding table with gleaned addresses to enable accurate tracking, and
- enforces address ownership, limiting the number of addresses claimed by any node.

This figure shows how IPv6 address glean works.

Figure 8: IPv6 Address Glean



Restrictions for IPv6 snooping

This section describes the restrictions for IPv6 snooping.

The IPv6 snooping feature is not supported on EtherChannel ports.

IPv6 multicast over multicast

Outlines the configuration and mechanism of IPv6 multicast over multicast tunnels for wireless controllers, detailing AP participation, dual-protocol environments, tunnel types, mixed mode limitations, and multicast support across Cisco Catalyst 9800 platforms.

A multicast-over-multicast tunnel is a network transmission mechanism that

- enables a wireless controller to efficiently deliver IPv6 multicast traffic to multiple APs simultaneously
- ensures all APs join the same IPv6 multicast address for coordinated data flow, and
- supports deployment in environments where APs may connect via both IPv4 and IPv6 protocols.

When IPv6 multicast over multicast is configured, all APs join the IPv6 multicast address. The multicast traffic from the wireless controller to the APs then flows over the IPv6 multicast tunnel.

The IPv4 APs use a unicast IPv4 CAPWAP tunnel and join the IPv4 multicast group. The IPv6 APs use a unicast IPv6 CAPWAP tunnel and join the IPv6 multicast group.

> **Note**
>
> Cisco IOS XE Gibraltar 16.10.1 does not support mixed mode for Multicast over Unicast and Multicast over Multicast over IPv4 and IPv6.

Table 99: Multicast support per platform

Platform	Multicast support - multicast over unicast	Multicast support - multicast over multicast
Cisco Catalyst 9800-40 Wireless Controller	No	Yes
Cisco Catalyst 9800-80 Wireless Controller	No	Yes
Cisco Catalyst 9800 Wireless Controller for Cloud - Small Template	Yes	Yes
Cisco Catalyst 9800 Wireless Controller for Cloud - Medium Template	No	Yes
Cisco Catalyst 9800 Wireless Controller for Cloud - Large Template	No	Yes
Cisco Catalyst 9800-L Wireless Controller	Yes	Yes

Configure IPv6 multicast-over-multicast (GUI)

Configure IPv6 multicast-over-multicast using GUI.

Enable IPv6 multicast-over-multicast for APs to optimize multicast traffic delivery using the GUI.

Procedure

1. Choose **Configuration** > **Services** > **Multicast**.
2. From the **AP Capwap Multicast** drop-down list, select **Multicast**.
3. Enter the **AP Capwap IPv6 Multicast group Address**.
4. Click **Apply**.

Configure IPv6 multicast-over-multicast (CLI)

Configure IPv6 multicast-over-multicast using commands.

Enable IPv6 multicast-over-multicast functionality on your device using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure IPv6 multicast-over-multicast address.

   Example

   ```
   Device(config)# wireless multicast {ipv4 ipv4-address| ipv6 ipv6-address}
   ff45:1234::86
   ```

Verify IPv6 multicast-over-multicast

Verify IPv6 multicast-over-multicast.

To verify the IPv6 multicast-over-multicast configuration, use these commands:

```
Device# show wireless multicast
Multicast : Enabled
AP Capwap Multicast : Multicast
AP Capwap IPv4 Multicast group Address : 231.1.1.1
AP Capwap IPv6 Multicast group Address : ff45:1234::86
Wireless Broadcast : Disabled
Wireless Multicast non-ip-mcast : Disabled
```

```
Device# show running-configuration | inc multicast
show run | inc multicast:--
```

```
wireless multicast
wireless multicast ipv6 ff45:1234::86
wireless multicast 231.1.1.1
```

Verify the multicast connection between the controller and the AP

Verify the multicast connection between the controller and the AP.

The Cisco Catalyst 9800 Series Wireless Controller initiates a ping request. This request passes through the CAPWAP multicast tunnel to reach the CAPWAP multicast receiver, which is the AP. In response, the AP sends ping packets to the CAPWAP multicast group IP address and returns the response to the controller.

- View the statistics for transmitted and received traffic on the AP to analyze the data sent through the multicast tunnel.
- To verify multicast tunnel activity, enhance the statistics on the AP. The enhanced statistics show joins, leaves, and data packets transmitted and received through the multicast tunnel.

To confirm if the APs receive multicast to multicast (mom) traffic sent by the controller, use this command:

```
Device# show ap multicast mom
AP Name                      MOM-IP     TYPE MOM- STATUS
-----------------------------------------------------
SS-E-1                       IPv4             Up
SS-E-2                       IPv4             Up
9130E-r3-sw2-g1012           IPv4             Up
9115i-r3-sw2-te1-0-38        IPv4             Up
AP9120-r3-sw3-Gi1-0-46       IPv4             Up
ap3800i-r2-sw1-te2-0-2       IPv4             Up
```

Directed multicast service

Introduces directed multicast service, explaining how APs transmit multicast packets as unicast frames to improve delivery and battery life for clients, and lists supported Cisco Aironet AP models.

A directed multicast service is a wireless networking feature that

- enables APs to transmit multicast packets as unicast frames to clients
- allows clients to reliably receive multicast packets that were missed during sleep mode (for battery savings), and
- improves packet delivery rates and battery life by sending unicast frames at higher wireless link rates.

After receiving this request, an AP buffers multicast traffic for the client. When the client wakes up, the AP transmits the buffered traffic as a unicast frame. The AP can transmit the unicast frames to the client at a higher wireless link rate. Without DMS, the client has to wake up at each Delivery Traffic Indication Map (DTIM) interval to receive multicast traffic.

Configure directed multicast service (GUI)

Configure directed multicast service using GUI.

Enable directed multicast service on a specific WLAN to optimize multicast traffic delivery using the GUI.

Procedure

1. Choose **Configuration** > **Wireless** > **WLANs** > **Wireless Networks**.
2. Select a **WLAN** to view the **Edit WLAN** window.

3. Click on the**Advanced** tab.
4. Check the **Directed Multicast Service** check box to enable the feature.
5. Click **Update & Apply to Device**.

Configure directed multicast service (CLI)

Configure directed multicast service using commands.

Enable directed multicast service (DMS) on a WLAN to improve multicast efficiency for 802.11v-capable clients using commands.

Before you begin

- This feature is enabled on receiving a request from a client. Make sure to configure this feature under WLAN.
- It is supported only on 802.11v-capable clients, such as Apple iPad and Apple iPhone.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN profile and enter the WLAN profile configuration mode.

   Example

   ```
   Device(config)# wlan test5
   ```

3. Disable the WLAN profile.

   Example

   ```
   Device(config-wlan)# shutdown
   ```

4. Configure DMS processing per WLAN.

   Example

   ```
   Device(config-wlan)# dms
   ```

5. Enable the WLAN profile.

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

Verify the directed multicast service configuration

Verify the directed multicast service configuration.

To verify the status of the DMS configuration on the controller, use these commands below. The DMS status is displayed under *IEEE 802.11v Parameters*.

```
Device# show wlan id 5
WLAN Profile Name     : test
==============================================
Identifier                                    : 5
Network Name (SSID)                           : test
Status                                        : Disabled
Broadcast SSID                                : Enabled
Universal AP Admin                            : Disabled
Max Associated Clients per WLAN               : 0
Max Associated Clients per AP per WLAN        : 0
Max Associated Clients per AP Radio per WLAN  : 200
!
.
.
.
Assisted-Roaming
Neighbor List                               : Disabled
Prediction List                             : Disabled
Dual Band Support                           : Disabled
! DMS status is displayed below.
IEEE 802.11v parameters
Directed Multicast Service                  : Enabled
BSS Max Idle                                : Disabled
Protected Mode                              : Disabled
Traffic Filtering Service                   : Disabled
BSS Transition                              : Enabled
Disassociation Imminent                     : Disabled
Optimized Roaming Timer                     : 40
Timer                                       : 200
WNM Sleep Mode                              : Disabled
802.11ac MU-MIMO                            : Disabled
802.11ax parameters
OFDMA Downlink                              : unknown
OFDMA Uplink                                : unknown
MU-MIMO Downlink                            : unknown
MU-MIMO Uplink                              : unknown
BSS Color                                   : unknown
Partial BSS Color                           : unknown
BSS Color Code
```

To verify the status of the DMS configuration on the controller for clients, use this command:

```
Device# show wireless client mac-address 6c96.cff2.83a0 detail | inc 11v
11v BSS Transition : implemented
11v DMS Capable : Yes
```

To verify the DMS request and response statistics, use this command:

```
Device# show wireless stats client detail | inc DMS
Total DMS requests received in action frame      : 0
Total DMS responses sent in action frame         : 0
Total DMS requests received in Re-assoc Request  : 0
Total DMS responses sent in Re-assoc Response    : 0
```

To verify the DMS configuration Cisco Aironet 2700 and 3700 Series APs, use this command:

```
AP# show controllers dot11Radio 0/1 | begin Global DMS
Global DMS - requests:0 uc:0 drop:408
DMS enabled on WLAN(s): dms-open
 test-open
```

To verify the DMS configuration on the Cisco Aironet 2800, 3800, and 4800 Series APs, use this command:

```
AP# show multicast dms all
 vapid    client               dmsid        TClas
 0      1C:9E:46:7C:AF:C0     1    mask:0x55, version:4, proto:0x11, dscp:0x0,
 sport:0, dport:9, sip:0.0.0.0, dip:224.0.0.251
```

Restrictions for wireless broadcast, non-IP multicast and multicast VLAN

Outlines restrictions for wireless broadcast, non-IP multicast, and multicast VLAN, including limitations on VLAN groups, VLAN pool mappings, and forwarding of non-IPv4 multicast and broadcast traffic.

Restrictions

- Wireless broadcast does not support VLAN groups.
- If you map a VLAN pool to a WLAN profile, you cannot forward non-IPv4 multicast or broadcast traffic.
- Non-IPv4 multicasts and broadcasts are only available to clients on the VLAN you mapped to the WLAN. You cannot forward them on VLANs defined by AAA override.

Configure non-IP wireless multicast (CLI)

Configure non-IP wireless multicast using commands.

Enable non-IP multicast traffic on wireless VLANs using commands.

Before you begin

- The non-IP Multicast feature is disabled globally by default.
- To allow traffic to pass for non-IP multicast, you must enable global wireless multicast.
- This feature is not supported in Fabric or Flex deployments.

Procedure

1. Enable non-IP multicast in all the VLANs.

   Example

   ```
   Device(config)# wireless multicast non-ip
   ```

   By default, the non-IP multicast in all the VLANs is in disabled state. You must enable wireless multicast for traffic to pass. Use the **no** form of this command to disable non-IP multicast in all the VLANs.

2. Enable non-IP multicast per VLAN.

Example

```
Device(config)# wireless multicast non-ip vlan 5
```

By default, non-IP multicast per VLAN is disabled. Both wireless multicast and wireless multicast non-IP must be enabled to allow traffic to pass. Use the **no** form of this command to disable non-IP multicast per VLAN.

3. Exit the configuration mode.

Example

```
Device(config)# end
```

Configure wireless broadcast (GUI)

Configure wireless broadcast using GUI.

Enable broadcast packets for wireless clients on selected VLANs using the GUI.

Procedure

1. Choose **Configuration** > **Services** > **Multicast**.
2. On the Multicast page, enable **Wireless Broadcast** to allow packet transmission for wireless clients.

   The default value is disabled.
3. In the Disabled VLAN table, click the arrow next to the VLAN ID to change its status from **Disabled** to **Enabled**. This enables broadcast packets for the selected VLAN.

   The default value is disabled.
4. Save the configuration.

Configure wireless broadcast (CLI)

Configure wireless broadcast using commands.

Enable wireless clients to receive specific broadcast packets on the network using commands.

Before you begin

- This feature applies only to non-ARP and DHCP broadcast packets.
- By default, the feature is disabled globally.
- The feature is not supported in Fabric or FlexConnect deployments.

Procedure

1. Enable broadcast packets for wireless clients.

Example

```
Device(config)# wireless broadcast
```

By default, broadcast packets for wireless clients are disabled. When **wireless broadcast** is enabled, broadcast traffic is allowed for each VLAN. Use the **no** form of this command to disable broadcasting packets.

2. Enable broadcast packets for single VLAN.

Example

```
Device(config)# wireless broadcast vlan 3
```

By default, the Broadcast Packets for a Single VLAN feature is disabled. You must enable wireless broadcast to allow broadcasting. Use the **no** form of this command to disable broadcast traffic for each VLAN.

3. Exit the configuration mode.

Example

```
Device(config)# end
```

Configure multicast-over-multicast for AP multicast groups (CLI)

Configure multicast-over-multicast for AP multicast groups using commands.

Enable multicast-over-multicast to optimize client multicast traffic distribution across APs using commands.

Procedure

1. Configure an all-AP multicast group to send a single packet to all the APs.

Example

```
Device(config)# ap capwap multicast 239.4.4.4
```

2. Enable multicast-over-multicast for multicasting client multicast group traffic to all the APs through the underlying all-AP multicast group.

Example

```
Device(config)# wireless multicast 239.4.4.4
```

IP address: Multicast-over-multicast IP address.

3. Exit the configuration mode.

Example

```
Device(config)# end
```

Verify wireless multicast

This section provides verifying wireless multicast details.

- The table provides a comprehensive list of commands for verifying various aspects of wireless multicast, including its general status, IP multicast mode, and mDNS bridging state.

Command	Description
show wireless multicast	Displays the multicast status and IP multicast mode, and each VLAN's broadcast and non-IP multicast status. Also displays the Multicast Domain Name System (mDNS) bridging state.
show wireless multicast group summary	Displays all (Group and VLAN) lists and the corresponding MGID values.
show wireless multicast [**source** *source*] **group** *group* **vlan** *vlanid*	Displays details of the specified (S,G,V) and shows all the clients associated with and their MC2UC status.
show ip igmp snooping wireless mcast-spi-count	Displays statistics of the number of multicast SPIs per MGID sent between IOS and the Wireless Controller Module. feature-id []
	Displays statistics of the number of multicast SPIs per MGID sent internally between IOS and the Wireless Controller Module. feature-id []
show ip igmp snooping wireless mcast-ipc-count	Displays the number of multicast IPCs per MGID sent to the wireless controller module.
show ip igmp snooping wireless mgid	Displays the MGID mappings.
show ip igmp snooping igmpv2-tracking	Displays the client-to-SGV mappings and the SGV-to-client mappings.
show ip igmp snooping querier vlan *vlanid*	Displays the IGMP querier information for the specified VLAN.
show ip igmp snooping querier detail	Displays the detailed IGMP querier information of all the VLANs.
show ipv6 mld snooping querier vlan *vlanid*	Displays the MLD querier information for the specified VLAN.
show ipv6 mld snooping wireless mgid	Displays MGIDs for the IPv6 multicast group.

Multicast optimization

This topic describes multicast optimization details.

Previously, multicast was based on a combination of multicast addresses and VLANs, grouped together as a single entity called MGID. Using the VLAN group can lead to an increase in duplicate packets. When the VLAN group feature is enabled, each client listens to the multicast stream on a different VLAN. As a result, the device creates a separate MGID for each multicast address and VLAN. Consequently, the upstream router sends one copy per VLAN, resulting in a number of copies equal to the number of VLANs in the group. Since all clients use the same WLAN, multiple copies of the multicast packet are sent across the wireless network. The multicast optimization feature suppresses duplication of multicast streams on the wireless medium between the device and the APs.

- This feature enables you to create a multicast VLAN for multicast traffic. You can configure one of the VLANs in the device as a multicast VLAN, where multicast groups are registered. Clients can listen to a multicast stream on the multicast VLAN. The MGID is generated based on the multicast VLAN and multicast IP addresses.

- If multiple clients on different VLANs within the same WLAN listen to a single multicast IP address, the device generates a single MGID. The device ensures that all multicast streams from clients in the VLAN group are sent via the multicast VLAN. This approach allows the upstream router to use a single entry for all VLANs in the VLAN group. Only one multicast stream is sent to the VLAN group, even when clients are on different VLANs. As a result, the device sends only one multicast stream over the network.

Configure IP multicast VLAN for WLAN (GUI)

Configure IP multicast VLAN for WLAN using GUI.

Configure an IP multicast VLAN for a Wireless LAN to enable efficient multicast traffic delivery using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click **Add**.
3. In the **General** tab, enter the **Name** and **Description**.
4. Enable the **Central Switching** and **Central Association** toggle buttons.
5. In the **Access Policies** tab, under the **VLAN** settings, choose the vlans from the **VLAN/VLAN Group** drop-down list and enter the **Multicast VLAN**.
6. Click **Apply to Device**.

Configure IP multicast VLAN for WLAN (CLI)

Configure IP multicast VLAN for WLAN using commands.

Set up IP multicast VLAN for WLAN to enable both IPv4 and IPv6 multicast forwarding to APs using commands.

Before you begin

- This feature is not supported in Fabric or FlexConnect deployments.
- Multicast VLAN is used for both IPv4 and IPv6 multicast forwarding to APs.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy default-policy-profile
   ```

3. Configure central association for locally switched clients.

Example

```
Device(config-wireless-policy)# central association
```

4. Configure WLAN for central switching.

   Example

   ```
   Device(config-wireless-policy)# central switching
   ```

5. Optional: Add a description for the policy profile.

   Example

   ```
   Device(config-wireless-policy)# description test
   ```

6. Assign the profile policy to the VLAN.

   Example

   ```
   Device(config-wireless-policy)# vlan 32
   ```

7. Configure multicast for the VLAN.

   Example

   ```
   Device(config-wireless-policy)# multicast vlan 84
   ```

8. Enable the profile policy.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

Verify the multicast VLAN configuration

Verify the multicast VLAN configuration.

To view the multicast VLAN associated with a policy profile along with the VLAN assigned to that profile, use this command:

```
Device# show wireless profile policy detail default-policy-profile
Policy Profile Name           : default-policy-profile
Description                   : default policy profile
Status                        : ENABLED
VLAN                          : vlan-pool1
                Multicast VLAN : 84
Client count                  : 0
Passive Client                : DISABLED
```

To view the multicast VLAN associated with a client, use this command:

```
Device# show wireless client mac ac2b.6e4b.551e detail
Client MAC Address : ac2b.6e4b.551e
Client IPv4 Address : 192.0.2.20
……….
VLAN : 82
Access VLAN : 82
                Multicast VLAN: 84
```

Multicast filtering

Outlines multicast filtering configuration per WLAN, detailing how the feature selectively controls IGMP and MLD group membership requests for IPv4 and IPv6 clients, prevents unauthorized AP group access, and reviews release support history on Cisco IOS XE.

A multicast filter is a wireless network feature that

- selectively controls the acceptance or dropping of multicast group membership requests from wireless clients
- supports configuration per WLAN for both IPv4 (using IGMP) and IPv6 (using MLD), and
- prevents addition of unauthorized clients to AP multicast groups by filtering protocol-specific reports.

The multicast filtering feature is disabled by default. You can enable or disable the multicast filtering feature per WLAN, from the controller.

You can enable or disable the multicast filtering feature per WLAN from the controller. When you enable this feature, the APs drop the internet group management protocol (IGMP) join request from any client that is part of the WLAN and attempts to join a Layer 3 multicast group address. When you disable this feature, the APs honor the IGMP join request from clients that are part of the WLAN.

This table shows AP behavior for IPv4 and IPv6:

Table 100: Multicast filtering per WLAN

Multicast filtering feature status	IPv4	IPv6
Enabled	AP drops the internet group management protocol (IGMP) membership report from a client that is a part of a WLAN.	AP drops the multicast listener discovery (MLD) report with multicast group address scope value greater than three from a client that is a part of a WLAN.
Disabled	AP honors the IGMP membership report from the client that is a part of a WLAN.	AP honors the MLD report from the client that is a part of a WLAN

Feature history for multicast filtering

- This table provides release and related information for the feature explained in this module.
- This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 101: Feature history for multicast filtering

Feature name	Release information	Feature description
Multicast filtering	Cisco IOS XE 17.3.1	From this release, the multicast filtering feature is supported on Layer 3 for IPv6.
Multicast filtering	Cisco IOS XE 17.2.1	From this release, the multicast filtering feature is supported on Layer 3 for IPv4.

Supported L3 multicast report for filtering

This section provides supported L3 multicast report for filtering details.

APs drop IGMP and MLD join requests from clients on the WLAN for any Layer 3 multicast group address, based on these filtering options:

- IPv4: IGMP versions to be filtered:
  - V1 membership report (0x12)
  - V2 membership report (0x16)
  - V3 membership report (0x22)
- IPv6: ICMPv6 types to be filtered, except link-local multicast packets:
  - Multicast Listener report: MLD Version 1 (131)
  - Multicast Listener report: MLD Version 2 (143)

Note

Filtering supported types prevents creating or adding a client entry to the AP multicast group table.

Configure multicast filtering (CLI)

Use this procedure to configure multicast filtering using commands.

Enable multicast filtering to control the flow of multicast traffic on a specific WLAN using commands.

Perform the procedure given here to create a policy profile and then enable Multicast Filtering on a WLAN:

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy rr-xyz-policy-1
   ```

3. Configure a multicast filter.

   Example

   ```
   Device(config-wireless-policy)#multicast filter
   ```

   Use the **no** form of this command to disable the feature.

What to do next

1. Create a policy tag. For more information about creating policy tags, see *Configuring a Policy Tag (CLI)*.
2. Map the policy tag to an AP. For more information about mapping a policy tag to an AP, see *Attaching a Policy Tag and Site Tag to an AP (CLI)*.

Verify multicast filtering

This section provides verifying multicast filtering details.

To verify if multicast filtering is enabled, use the **show wireless profile policy detailed** *named-policy-profile* command:

```
Device# show wireless profile policy detailed named-policy-profile
Policy Profile Name                : named-policy-profile
Description                        :
Status                             : DISABLED
VLAN                               : 91
Multicast VLAN                     : 0
OSEN client VLAN                   :
Multicast Filter                   : ENABLED
```

97 Map-Server Per-Site Support

Topics:

- Map server per site support

Map server per site support

Explains how map servers per site enable flexible traffic segmentation and virtual network assignments for multiple sites.

A map server per site support is a network feature that

- selects the map server based on the client's subnet to enable per-site segmentation. The controller supports multiple sites and segregates each site's traffic
- allows both Enterprise and Guest sites to use dedicated or shared map servers, and
- enables flexible virtual network assignment using the Layer 2 Virtual Network Identifier (VNID) selection.

This list shows the map server selection order for AP query and client registration:

- Per-L3 VNID map server.
- Per site (ap-group) map server.
- Default or global map server.

Benefits

Some of the benefits of using the map server per site feature are:

- You can use a single large site with horizontal-scaling of the map server and border nodes.
- You can share the controller across multiple sites. Each site can have its own map server and virtual network (VNID), and traffic can still be segmented from each site.
- You can share the guest map server across multiple sites while keeping the enterprise map server separate.
- You can use the same SSID across different sites. Within a site, the SSIDs can belong to different virtual network domains.

Configure the default map server (GUI)

Configure the default map server using GUI.

Set up the default map server to manage control plane operations within the wireless fabric.

Procedure

1. Choose **Configuration** > **Wireless** > **Fabric**.
2. On the **Fabric** page, click the **Control Plane** tab.
3. In the **Control Plane Name** list, click **default-control-plane**.
4. In the **Edit Control Plane** window that is displayed, click **Add**.
5. Enter the IP address of the map server.
6. Set the **Password Type** as either **Unencrypted** or **AES**.
7. Enter the **Pre Shared Key**.
8. Click **Save**.
9. Click **Update & Apply to Device**.

Configure the default map server (CLI)

Configure the default map server using commands.

To configure a wireless fabric control-plane with a specified map server and pre-shared key for LISP control-plane redundancy and secure AP/client join processes.

Before you begin

- The global map server is the default map server that is used for both AP query (when an AP joins) as well as for client registration (when a client joins).
- We recommend that you configure map servers in pairs to ensure redundancy because the LISP control-plane does not support redundancy inherently.
- To share a map server set, create a map server group, which can be shared across site profiles, fabric profiles, Layer 2 and Layer3 VNID, as well with the default map server.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the control plane name.

   Example

   ```
   Device(config)# wireless fabric control-plane control-plane-name test-map
   ```

   If you do not provide a control plane name, the default-control-plane that is auto generated is used.

3. Configure the IP address and the key for the control plane.

   Example

   ```
   Device((config-wireless-cp)# ip address 10.12.13.14 key pre-shared-key secret
   ```

Configure a map server per site (GUI)

Configure a map server per site using GUI.

Configure the primary and backup map server controllers for a specific site using the GUI.

Before you begin
Ensure you have configured an AP Join Profile before setting up the primary and backup controllers.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.

2. On the **AP Join Profile** page, click the AP Join Profile name.
3. In the **Edit AP Join Profile** window, click the **CAPWAP** tab.
4. In the **High Availability** tab under **Backup Controller Configuration**, check the **Enable Fallback** check box.
5. Enter the primary and secondary controller names and IP addresses.
6. Click **Update & Apply to Device**.

Configure a map server per site (CLI)

Configure a map server per site using commands.

Set up a map server for each site or AP group.

Before you begin

You can configure map server for each site or each AP group. If a map server is not configured for each VNID or subnet, per-site map server is used for AP queries and client registration.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a site tag and enter the site tag configuration mode.

   Example

   ```
   Device(config)# wireless tag site site-tag test-site
   ```

3. Associate a fabric control plane name with a site tag.

   Example

   ```
   Device(config-wireless-site)# fabric control-plane map-server-name test-map
   ```

Create a map server for each VNID (GUI)

Create a map server for each VNID using GUI.

Configure map servers for Layer 2 VNIDs to enable secure fabric communication.

Procedure

1. Choose **Configuration** > **Wireless Plus** > **Fabric** > **Fabric Configuration**.
2. In the **Profiles** tab, click **Add** to add a new Fabric Profile.
3. In the **Add New Profile** window that is displayed, enter a name and description for the profile.

4. Specify the L2 VNID and SGT Tag details.
5. In the **Map Servers** section, specify the IP address and preshared key details for Server 1.
6. Optionally, you can specify the IP address and preshared key details for Server 2.
7. Click **Save & Apply to Device.**

Create a map server for each VNID (CLI)

Create a map server for each VNID using commands.

Follow the procedure given below to configure map server for each VNID in Layer 2 and Layer 3 or a map server for a client VNID.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a map server for each VNID in Layer 2 and Layer 3 or a map server for a client VNID. Use one of the following:

   Example

   ```
   Device(config)# wireless fabric name vnid-map test1 l2-vnid l2-vnid l3-vnid
   l3vnid 10 ip network-ip subnet-mask 10.8.6.2 255.255.255.236 control-plane
   control-plane-name cp1
   ```

   Example

   ```
   Device(config)# wireless fabric name vnid-map test1 l2-vnid l2-vnid
   control-plane control-plane-name cp1
   ```

Create a fabric profile and associate a tag and VNID (GUI)

Create a fabric profile and associate a tag and VNID using GUI.

Use these steps to create a fabric profile in the Cisco wireless GUI and associate it with a tag and VNID.

Procedure

1. Choose **Configuration** > **Wireless** > **Fabric**.
2. In the **Profiles** tab on **Fabric Configuration** page, click **Add** to add a new profile.
3. In the **Add New Profile** window that is displayed, enter a name and description for the profile.
4. Specify the L2 VNID and SGT Tag details.
5. Click **Save & Apply to Device**.

Create a fabric profile and associate a tag and VNID (CLI)

Create a fabric profile and associate a tag and VNID using commands.

Follow the procedure given below to create a fabric profile and associate the VNID to which the client belongs and the SGT tag to this profile.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a fabric profile.

   Example

   ```
   Device(config)# wireless profile fabric fabric-profile-name test-fabric
   ```

3. Configure an SGT tag.

   Example

   ```
   Device(config-wireless-fabric)# sgt-tag value 5
   ```

4. Configure a client Layer 2 VNID.

   Example

   ```
   Device(config-wireless-fabric)# client-l2-vnid vnid 10
   ```

Verify the map server configuration

Verify the map server configuration.

Use the following commands to verify the map server configuration:

```
Device# show wireless fabric summary

Fabric Status      : Enabled


Control-plane:
Name                            IP-address        Key
       Status
-------------------------------------------------------------------------------------
test-map                        10.12.13.14       test1
       Down


Fabric VNID Mapping:
```

```
  Name               L2-VNID        L3-VNID        IP Address
Subnet        Control plane name
------------------------------------------------------------------------------

  test1               12             10             10.6.8.9
255.255.255.236    test2
```

```
Device# show wireless fabric vnid mapping

Fabric VNID Mapping:
  Name               L2-VNID        L3-VNID        IP Address
Subnet      Control Plane Name
------------------------------------------------------------------------------

  fabric1             1              0              9.6.51.0
255.255.255.0          map-server-name
```

```
Device# show wireless profile fabric detailed profile-name

Profile-name      : fabric-ap
VNID              : 1
SGT               : 500
Type              : Guest

Control Plane Name       Control-Plane IP   Control-Plane Key
------------------------------------------------------------------------------

 Ent-map-server           5.4.3.2                guest_1
```

```
Device# show ap name ap-name config general

Fabric status                                  : Enabled
RLOC                                           : 2.2.2.2
Control Plane Name  : ent-map-server
```

```
Device# show wireless client mac mac-address detail

Fabric status : Enabled
RLOC                   : 2.2.2.2
Control Plane Name  : ent-map-server
```

```
Device# show wireless tag site detailed site-tag

Site Tag Name        : default-site-tag
Description          : default site tag
---------------------------------------
AP Profile           : default-ap-profile
Local-site           : Yes
Fabric-control-plane: Ent-map-server
```

98 Volume Metering

Topics:

- Volume metering

Describes how to configure AP reporting intervals and controller accounting updates so administrators can reduce latency in client usage reporting and forward interim accounting data to a RADIUS server.

Volume metering

Introduces Volume Metering principles and provides instructions for configuring volume metering using detailed tasks, guiding users through the setup and implementation process for monitoring network traffic volume.

A volume metering feature is an access point data reporting capability that

- allows you to set the interval for client accounting statistics updates from the access point (AP) to the network controller,
- supports configurable reporting frequencies from five to 90 seconds, and
- helps minimize delays in updating usage data for connected devices.

Additional information

By default, the access point sends client accounting updates to the controller every 90 seconds, which then forwards this information to the RADIUS server. With volume metering features, you can adjust the reporting interval from five seconds to 90 seconds.

This flexibility enables network administrators to better control the timeliness of usage data, ensuring more immediate reporting and improved network accounting accuracy.

Example

If you want to decrease the delay in receiving accounting data for a device, you can configure the reporting interval to 10 seconds so that the access point sends updates to the controller—and subsequently to the RADIUS server—much more frequently than the default 90 seconds.

This supports more accurate, real-time accounting of network usage.

Configure volume metering (CLI)

Use this procedure to configure volume metering on your device. This task helps you set the reporting intervals for 2.4 GHz and 5 GHz radios and configure accounting updates for client sessions.

Configure device parameters to control how often the system reports wireless usage data and sends accounting updates to the Remote Authentication Dial-In User Service (RADIUS) server.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an access point (AP) profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

3. Configure the 2.4 GHz (802.11ax) reporting interval.

Example

```
Device(config-ap-profile)# dot11 24ghz reporting-interval reporting-interval
```

4. Configure the 5 GHz (802.11ax) reporting interval.

   Example

   ```
   Device(config-ap-profile)# dot11 5ghz reporting-interval reporting-interval
   ```

5. Return to global configuration mode.

   Example

   ```
   Device(config-ap-profile)# exit
   ```

6. Set the time interval (in minutes) at which the controller sends interim accounting updates for the client to the RADIUS server.

   Example

   ```
   Device(config)# aaa accounting update periodic interval-in-minutes
   ```

7. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config)# exit
   ```

The device reports volume metering data at your specified intervals and sends accounting updates to the RADIUS server.

99 Enabling Syslog Messages in Access Points and Controller for Syslog Server

Topics:

- Syslog messages in access points and controllers
- AP logging to the in-memory buffer and flash

Explains how to configure syslog messaging on access points and controllers, including setup for syslog servers and logging to in-memory buffers and flash storage.

Syslog messages in access points and controllers

Introduces syslog messaging concepts for access points and controllers, detailing client state change support, configuration procedures via CLI and GUI, sample syslog formats, and syslog server setup instructions.

A syslog message in access points and controllers is a log communication that

- captures and records operational events generated by each device
- is routed to various destinations, including memory buffers, terminal sessions, device storage, or external syslog servers, and
- allows administrators to configure logging independently on access points and controllers to meet specific network requirements.

Configuring message logging in the IOS XE controller

System Message Logging in Cisco Catalyst 9800 Series Controllers is a platform-independent IOS and IOS XE feature. For more details, see:

- System Message Logging
- Configuration Logger Persistency chapter in *System Management Configuration Guide*
- Logging to Local Nonvolatile Storage chapter in *System Management Configuration Guide*
- Embedded Syslog Manager (ESM) chapter in *System Management Configuration Guide*
- Configuration Change Notification and Logging chapter in *System Management Configuration Guide*

Syslog support for client state change

This concept explains how syslog client state change events track critical transitions in wireless client connectivity, including the data captured and expected behavior when the feature is enabled.

A syslog client state change event is a network monitoring mechanism that

- records when a wireless client joins, obtains a new IP address, or disconnects from the network,
- captures details such as client IP addresses and access point (AP) names to support network monitoring, and
- provides actionable event logs for operational troubleshooting.

Feature details

A syslog event is generated in these situations:

- When a client enters the RUN state,
- when a client receives a new IP address (IPv4 or IPv6) in the RUN state, and
- when a client in the RUN state is deleted.

> **Note**
>
> If syslog support for client state change is enabled and an access point (AP) transitions from standalone to connected, usernames may temporarily appear as null in syslog messages and in client details for 802.1X clients associated with that AP.
>
> This behavior does not affect operations. The system updates usernames automatically after about 30 seconds.

For more information about the IOS XE Controller, see System Message Logging.

Example: Syslog Support for Client State Change

For example, when a client joins a wireless network and obtains a new IP address, a syslog message is generated to record the event, including details such as the client's IP address and the associated AP name.

Configure syslog support for client state change (CLI)

Learn how to configure syslog support for client state change using CLI commands.

Enable the system to log detailed client state changes for monitoring and troubleshooting.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable detailed syslogs for client events.

   Example

   ```
   Device(config)# wireless client syslog-detailed
   ```

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The device logs detailed syslog messages whenever a client state changes, aiding in network monitoring and troubleshooting.

Sample syslogs

Sample syslogs details.

802.11x authentication

This example shows a client IP update:

```
Oct  1 14:41:27.785 IST: %CLIENT_ORCH_LOG-7-CLIENT_IP_UPDATED:
Chassis 1 R0/0: wncd: Username (dev2), MAC: 0062.xxxx.0077,
IP fe80::262:aff:xxxx:77 101.6.2.119 2001:300:8:0:362:aff:xxxx:77
2001:300:8:0:762:aff:xxxx:77
2001:300:8:0:562:aff:xxxx:77 2001:300:8:0:962:aff:xxxx:77
2001:300:8:0:462:aff:xxxx:77
IP address updated, associated to AP (Asim_06-11) with SSID (dev_abcd_wlan_1)
```

This example shows a client RUN state:

```
Oct  1 14:41:27.779 IST: %CLIENT_ORCH_LOG-7-CLIENT_MOVED_TO_RUN_STATE:
Chassis 1 R0/0: wncd: Username (dev2), MAC: 0062.xxxx.006a, IP 101.xxxx.2.106
 associated to AP
(Asim_06-10) with SSID (dev_abcd_wlan_1)
```

Open authentication

This example shows a client IP update:

```
Sep 18 03:22:35.902: %CLIENT_ORCH_LOG-7-CLIENT_IP_UPDATED:
Chassis 1 R0/0: wncd: Username (null), MAC: 6014.xxxx.c5fb, IP 9.9.xxxx.252
fe80::643c:87c1:xxxx:c1c4 IP address updated,
associated to AP (AP2C5A.xxxx.159A) with SSID (test1)
```

This example shows a client RUN state:

```
Sep 18 03:22:35.257: %CLIENT_ORCH_LOG-7-CLIENT_MOVED_TO_RUN_STATE:
Chassis 1 R0/0: wncd: Username (null), MAC: 6014.xxxx.c5fb, IP 9.9.xxxx.252
associated to AP (AP2C5A.xxxx.159A) with SSID (test1)
```

This example shows a client delete state:

```
Sep 18 03:24:45.083: %CLIENT_ORCH_LOG-7-CLIENT_MOVED_TO_DELETE_STATE:
Chassis 1 R0/0: wncd: Username (null), MAC: 6014.xxxx.c5fb, IP
fe80::643c:xxxx:e316:c1c4 2001:300:42:0:643c:87c1:xxxx:c1c4
2001:300:42:0:xxxx:82ce:1ae4:5a32 9.9.xxxx.252 disconnected from AP
(AP2C5A.xxxx.159A) with SSID (test1)
```

Configure the syslog server for the controller (GUI)

Use this procedure to configure a syslog server for controller log management using the GUI.

Enable centralized log management by forwarding controller system logs to a designated syslog server through the controller GUI.

Procedure

1. Choose **Troubleshooting** > **Logs**.

2. Click **Manage Syslog Servers**.
3. In **Log Level Settings**, from the **Syslog** drop-down list, choose a severity level.
4. From the **Message Console** drop-down list, choose a logging level.
5. In **Message Buffer Configuration**, from the **Level** drop-down list, choose a server logging level.
6. In **Size (bytes)**, enter the buffer size. The value can range from 4,096 to 2,147,483,647.
7. In **IP Configuration** settings, click **Add**.
8. From the **Server Type** drop-down list, choose **IPv4**, **IPv6**, or **FQDN**.
9. If you choose **IPv4** or **IPv6**, enter the **Server Address**. If you choose **FQDN**, enter the **Host Name**. Then choose the IP type and the appropriate **VRF Name** from the drop-down lists.

   To delete a syslog server, click **x** next to the appropriate server entry under the **Remove** column.

   > **Note**
   >
   > Spaces are not allowed in the host name.

10. Click **Apply to Device**.

    > **Note**
    >
    > When you click **Apply to Device**, the system applies the changes. If you click **Cancel**, the system discards the changes.

After you apply the configuration, log messages are forwarded to the specified syslog server.

Configure a syslog server for a controller (CLI)

Use this procedure to configure a syslog server for an embedded-wireless controller using CLI commands to enable centralized logging and monitoring for auditing and troubleshooting.

Configure the controller to send system and event messages to a remote syslog server for monitoring.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the syslog server IP address and configure its parameters.

   Example

   ```
   Device(config)# logging hostname ipv6
   ```

3. Enable the facility parameter for syslog messages.

   Example

   ```
   Device(config)# logging facility {auth | cron | daemon | kern | local0 |
   local1 | local2 | local3 | local4 | local5 | local6 | local7 | lpr | mail |
   news | sys10 | sys11 | sys12 | sys13 | sys14 | sys9 | syslog | user | uucp}
   ```

   Example

   ```
   Device(config)# logging facility syslog
   ```

   You can enable these facility parameters for syslog messages:

   - **auth** Authorization system.
   - **cron** Cron facility.
   - **daemon** System daemons.
   - **kern** Kernel.
   - **local0** to **local7** Local use.
   - **lpr** Line printer system.
   - **mail** Mail system.
   - **news** USENET news.
   - **sys10** to **sys14** and **sys9** System use.
   - **syslog** Syslog itself.
   - **user** User process.
   - **uucp** Unix-to-Unix copy system.

4. Set the logging level for the syslog server.

   Example

   ```
   Device(config)# logging trap {severity-level | alerts | critical | debugging
    | emergencies | errors | informational | notifications | warnings}
   ```

   Example

   ```
   Device(config)# logging trap 2
   ```

   severity-level Refers to the logging severity level. The valid range is from zero to seven.

   These are the syslog server logging levels:

   - **emergencies** Signifies severity 0. The system is not usable.
   - **alerts** Signifies severity 1. Immediate action is required.
   - **critical** Signifies severity 2. Critical conditions.
   - **errors** Signifies severity 3. Error conditions.

- **warnings** Signifies severity 4. Warning conditions.
- **notifications** Signifies severity 5. Normal but significant conditions.
- **informational** Signifies severity 6. Informational messages.
- **debugging** Signifies severity 7. Debugging messages.

> **Note**
>
> Select a syslog level to view the supported levels. Selecting a level also enables all lower levels.
>
> If you enable the *critical* syslog level, the system also enables lower levels–*alerts* and *emergencies*.

5. Enter privileged EXEC mode.

Example

```
Device(config)# end
```

Alternatively, press **Ctrl-Z** to exit global configuration mode.

The controller now forwards syslog messages to the specified syslog server as configured.

AP logging to the in-memory buffer and flash

Describes access point logging capabilities to in-memory buffers and flash storage, covering terminal logging, syslog server configuration for AP profiles, GUI-based syslog settings, and verification commands.

AP always log messages to an in-memory buffer. Once the buffer reaches 40 KB, its contents are automatically written to flash memory, and a new buffer is created.

Administrators can manage and view these logs using AP commands.

- **show logging** command to display the contents of the in-memory logging buffer
- **show flash syslogs** command to list all log files stored in flash, along with other diagnostic files
- **<filename>** command to display the contents of an individual log file stored in flash
- **copy syslogs <filename>** command to transfer a specific syslog file to an external server. To see available options for this command, use **copy syslogs <filename>**

AP logging to terminal

This topic explains how access points support real-time logging of messages to an active SSH terminal session and the use of the terminal monitor command for enabling or disabling this feature.

Access points support real-time logging of messages to an active SSH terminal session, which can be enabled or disabled using the **terminal monitor** and **terminal monitor disable** commands.

Administrators can enable real-time logging to an SSH terminal session using the **terminal monitor** command. To disable real-time logging to the session, use the **terminal monitor disable** command. In addition to SSH terminal sessions, APs send a subset of log messages to the serial console, providing another method for real-time monitoring.

Configure AP logging to a syslog server

This topic explains how to configure access point (AP) syslog logging, including setting the syslog host, filtering messages by severity and facility, enabling secure transmission, and viewing current syslog settings.

Use the **syslog** command in the AP join profile to configure the destination IP address for syslog messages and manage which messages are sent based on severity and facility levels.

Configure the syslog host

- Use the **syslog host <IP address>** command to specify the destination IP address for syslog messages.
- By default, the syslog host is **255.255.255.255** , the IPv4 limited broadcast address.
- Configure IP helper addresses on the AP subnet router to forward broadcasts to one or more syslog servers.
- To reset the syslog host to **255.255.255.255** , use either the **default syslog host** or **no syslog host** command.
- To prevent the AP from sending syslog messages entirely, use **syslog host 0.0.0.0** .
- If a subnet contains more than 20 access points, do not log to the broadcast address. This prevents flooding the broadcast domain with log messages. Configure a specific syslog destination IP address.
- If you do not use the AP syslog feature, set the syslog host to **0.0.0.0** by entering the **syslog host 0.0.0.0** command.

Filter messages by severity

- Use the **syslog level <levelname>** command to filter messages based on severity.
- By default, the severity level is **informational** (**severity=6**), which means the system sends all messages except debugging logs to the server.

Filter messages by facility

- Use the **syslog facility <facilityname>** command to filter messages based on the facility code (facility name).
- By default, the facility is **kernel** (**kern** , **code=0**), so the system sends only kernel-related messages.
- To send messages from all facilities, configure the facility as **local7** .
- The configured facility name is included in the facility field of transmitted syslog messages.

> **Note**
>
> Most AP log messages use the **kern** facility, while terminal access logs such as SSH and console access use the **auth** facility.

Secured syslog transmission

- Use the **syslog secured** command to enable Transport Layer Security (TLS), as defined in RFC 5425, to transmit syslog messages securely instead of using UDP.

- TLS-based syslog transmission is supported starting with software versions 17.9.6 and 17.12.1 (released in June 2023 and December 2023, respectively).

View syslog settings

To display the current syslog settings for an AP, use the **show capwap client configuration** command.

Configure a syslog server for an AP profile (CLI)

Learn how to configure a syslog server for an AP profile using CLI commands.

Configure a syslog server for an AP profile.

This ensures access points send system log messages to a specified remote server for central monitoring and troubleshooting.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile. Then, enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile
   ```

3. Configure the facility parameter used by syslog messages.

   Example

   ```
   Device(config-ap-profile)# syslog facility
   ```

4. Enter the Syslog server IP address and parameters to define where the AP profile sends log messages.

   Example

   ```
   Device(config-ap-profile)# syslog host ip-address
   ```

5. Set the syslog server logging level to determine which messages the AP profile sends.

   Example

   ```
   Device(config-ap-profile)# syslog level {alerts | critical | debugging |
   emergencies | errors | informational | notifications | warnings}
   ```

   Example

   ```
   Device(config-ap-profile)# syslog level
   ```

   The syslog server supports these logging levels:

- **emergencies** –Signifies severity 0. Indicates that the system is not usable.
- **alerts** –Signifies severity 1. Indicates that an immediate action is required.
- **critical** –Signifies severity 2. Indicates a critical condition.
- **errors** –Signifies severity 3. Indicates an error condition.
- **warnings** –Signifies severity 4. Indicates a warning condition.
- **notifications** –Signifies severity 5. Indicates normal, but significant, conditions.
- **informational** –Signifies severity 6. Indicates informational messages.
- **debugging** –Signifies severity 7. Indicates debugging messages.

> **Note**
>
> Select a syslog level to enable it and all lower levels.
>
> If you enable the *critical* syslog level, *alerts* and *emergencies* are also enabled.

6. Return to privileged EXEC mode.

Example

```
Device(config-ap-profile)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The syslog server is configured for the specified AP profile. APs that use this profile will send log messages with the selected severity levels to the designated remote server.

Configure the AP syslog settings (GUI)

Learn how to configure syslog settings on access points (APs) using the GUI to control log destination, facility, and security options.

Set up syslog parameters for access points. This enables monitoring and centralized logging of events.

Procedure

---

1. Choose **Configuration**, then **Tags and Profiles**, then **AP Join**.
2. Select the APs from the AP list.

   The **Edit AP Join Profile** window is displayed.
3. Click the **Management** tab.
4. Select the **Device** tab.
5. In the **System Log** section:
   a) From the **Facility Value** drop-down list, select a value.
   b) Enter the IP address in the **Host IPv4 or IPv6 Address** field.

c) From the **Log Trap Value** drop-down list, select a value.
d) To enable **Secured**, check the box. To disable **Secured**, clear the box.

6. Click **Update and Apply to Device**

The selected APs are updated with the new syslog configuration, enabling centralized logging according to your specified parameters.

Verify syslog server configurations

Verify Syslog Server Configurations.

Verify global syslog server settings for all APs

To view the global syslog server settings for all access points that joins the controller, use this command:

```
Device# show ap config general
Cisco AP Name : APA0F8.4984.5E48
================================================

Cisco AP Identifier : a0f8.4985.d360
Country Code : IN
Regulatory Domain Allowed by Country : 802.11bg:-A 802.11a:-DN
AP Country Code : IN - India
AP Regulatory Domain
Slot 0 : -A
Slot 1 : -D
MAC Address : a0f8.4984.5e48
IP Address Configuration : DHCP
IP Address : 9.4.172.111
IP Netmask : 255.255.255.0
Gateway IP Address : 9.4.172.1
Fallback IP Address Being Used :
Domain :
Name Server :
CAPWAP Path MTU : 1485
Telnet State : Disabled
SSH State : Disabled
Jumbo MTU Status : Disabled
Cisco AP Location : default location
Site Tag Name : ST1
RF Tag Name : default-rf-tag
Policy Tag Name : PT3
AP join Profile : default-ap-profile
Primary Cisco Controller Name : WLC2
Primary Cisco Controller IP Address : 9.4.172.31
Secondary Cisco Controller Name : Not Configured
Secondary Cisco Controller IP Address : 0.0.0.0
Tertiary Cisco Controller Name : Not Configured
Tertiary Cisco Controller IP Address : 0.0.0.0
Administrative State : Enabled
Operation State : Registered
AP Certificate type : Manufacturer Installed Certificate
AP Mode : Local
AP VLAN tagging state : Disabled
AP VLAN tag : 0
CAPWAP Preferred mode : Not Configured
AP Submode : Not Configured
Office Extend Mode : Disabled
Remote AP Debug : Disabled
Logging Trap Severity Level : notification
```

```
Software Version : 16.10.1.24
Boot Version : 1.1.2.4
Mini IOS Version : 0.0.0.0
Stats Reporting Period : 180
LED State : Enabled
PoE Pre-Standard Switch : Disabled
PoE Power Injector MAC Address : Disabled
Power Type/Mode : PoE/Full Power (normal mode)
Number of Slots : 3
AP Model : AIR-AP1852I-D-K9
IOS Version : 16.10.1.24
Reset Button : Disabled
AP Serial Number : KWC212904UB
Management Frame Protection Validation : Disabled
AP User Mode : Automatic
AP User Name : Not Configured
AP 802.1X User Mode : Global
AP 802.1X User Name : Not Configured
Cisco AP System Logging Host : 9.4.172.116
AP Up Time : 11 days 1 hour 15 minutes 52 seconds
AP CAPWAP Up Time : 6 days 3 hours 11 minutes 6 seconds
Join Date and Time : 09/05/2018 04:18:52
Join Taken Time : 3 minutes 1 second
Join Priority : 1
Ethernet Port Duplex : Auto
Ethernet Port Speed : Auto
AP Link Latency : Disable
AP Lag Configuration Status : Disabled
AP Lag Operational Status : Disabled
Lag Support for AP : Yes
Rogue Detection : Enabled
Rogue Containment auto-rate : Disabled
Rogue Containment of standalone FlexConnect APs : Disabled
Rogue Detection Report Interval : 10
Rogue AP minimum RSSI : -90
Rogue AP minimum transient time : 0
AP TCP MSS Adjust : Enabled
AP TCP MSS Size : 1250
AP IPv6 TCP MSS Adjust : Enabled
AP IPv6 TCP MSS Size : 1250
Hyperlocation Admin Status : Disabled
Retransmit count : 5
Retransmit interval : 3
Fabric status : Disabled
FIPS status : Disabled
WLANCC status : Disabled
USB Module Type : USB Module
USB Module State : Enabled
USB Operational State : Disabled
USB Override : Disabled
Lawful-Interception Admin status : Disabled
Lawful-Interception Oper status : Disabled
```

Verify syslog server settings for a specific AP

To view the syslog server settings for a specific access point, use this command:

```
Device# show ap name <ap-name> config general
show ap name APA0F8.4984.5E48 config general
Cisco AP Name : APA0F8.4984.5E48
================================================
```

```
Cisco AP Identifier : a0f8.4985.d360
Country Code : IN
Regulatory Domain Allowed by Country : 802.11bg:-A 802.11a:-DN
AP Country Code : IN - India
AP Regulatory Domain
Slot 0 : -A
Slot 1 : -D
MAC Address : a0f8.4984.5e48
IP Address Configuration : DHCP
IP Address : 9.4.172.111
IP Netmask : 255.255.255.0
Gateway IP Address : 9.4.172.1
Fallback IP Address Being Used :
Domain :
Name Server :
CAPWAP Path MTU : 1485
Telnet State : Disabled
SSH State : Disabled
Jumbo MTU Status : Disabled
Cisco AP Location : default location
Site Tag Name : ST1
RF Tag Name : default-rf-tag
Policy Tag Name : PT3
AP join Profile : default-ap-profile
Primary Cisco Controller Name : WLC2
Primary Cisco Controller IP Address : 9.4.172.31
Secondary Cisco Controller Name : Not Configured
Secondary Cisco Controller IP Address : 0.0.0.0
Tertiary Cisco Controller Name : Not Configured
Tertiary Cisco Controller IP Address : 0.0.0.0
Administrative State : Enabled
Operation State : Registered
AP Certificate type : Manufacturer Installed Certificate
AP Mode : Local
AP VLAN tagging state : Disabled
AP VLAN tag : 0
CAPWAP Preferred mode : Not Configured
AP Submode : Not Configured
Office Extend Mode : Disabled
Remote AP Debug : Disabled
Logging Trap Severity Level : notification
Software Version : 16.10.1.24
Boot Version : 1.1.2.4
Mini IOS Version : 0.0.0.0
Stats Reporting Period : 180
LED State : Enabled
PoE Pre-Standard Switch : Disabled
PoE Power Injector MAC Address : Disabled
Power Type/Mode : PoE/Full Power (normal mode)
Number of Slots : 3
AP Model : AIR-AP1852I-D-K9
IOS Version : 16.10.1.24
Reset Button : Disabled
AP Serial Number : KWC212904UB
Management Frame Protection Validation : Disabled
AP User Mode : Automatic
AP User Name : Not Configured
AP 802.1X User Mode : Global
AP 802.1X User Name : Not Configured
Cisco AP System Logging Host : 9.4.172.116
AP Up Time : 11 days 1 hour 15 minutes 52 seconds
```

```
AP CAPWAP Up Time : 6 days 3 hours 11 minutes 6 seconds
Join Date and Time : 09/05/2018 04:18:52
Join Taken Time : 3 minutes 1 second
Join Priority : 1
Ethernet Port Duplex : Auto
Ethernet Port Speed : Auto
AP Link Latency : Disable
AP Lag Configuration Status : Disabled
AP Lag Operational Status : Disabled
Lag Support for AP : Yes
Rogue Detection : Enabled
Rogue Containment auto-rate : Disabled
Rogue Containment of standalone FlexConnect APs : Disabled
Rogue Detection Report Interval : 10
Rogue AP minimum RSSI : -90
Rogue AP minimum transient time : 0
AP TCP MSS Adjust : Enabled
AP TCP MSS Size : 1250
AP IPv6 TCP MSS Adjust : Enabled
AP IPv6 TCP MSS Size : 1250
Hyperlocation Admin Status : Disabled
Retransmit count : 5
Retransmit interval : 3
Fabric status : Disabled
FIPS status : Disabled
WLANCC status : Disabled
USB Module Type : USB Module
USB Module State : Enabled
USB Operational State : Disabled
USB Override : Disabled
Lawful-Interception Admin status : Disabled
Lawful-Interception Oper status : Disabled
```

100 Login Banner

Topics:

- Login banners

This chapter explains how to configure customizable login banners on the controller via GUI or CLI, including delimiter rules, examples, verification, and saving steps, so administrators can display legal notices or security warnings before user authentication.

Login banners

This concept explains the purpose, configuration requirements, and known limitations of login banners in controller systems.

A login banner is a user authentication feature that

- displays a customizable warning or informational message when you attempt to access the controller
- uses a delimiting character to mark the start and end of the banner text, and
- helps communicate security policies or legal notices to users before granting access.

Additional reference information

To configure a login banner, you must specify a delimiting character that signals the system to treat the enclosed text as the banner message. After entering the delimiting character, enter the message text, and then repeat the same character at the end.

The delimiter can be any single character from the extended ASCII set, but that character cannot appear inside the banner message once selected as the delimiter.

For example, to create a login banner with the delimiter `#`, enter:

```
banner login #Unauthorized access is prohibited#
```

Note

When HTTP authentication is configured using TACACS+ or RADIUS, the banner message does not display on the Web UI. (Addressed CSCvr26813: C9800 - Banner not seen on GUI with TACACS authentication.)

Configure a login banner (GUI)

Use this procedure to configure a login banner using the web-based GUI. This enables you to display a customized message to users prior to device login.

Set up a custom login banner to communicate important information or legal notices to users before they access the device.

Procedure

1. Choose **Administration** > **Device** .
2. In the **General** tab, in the **Banner** field, enter a name for the device and a message.
3. Click **Apply** .

The customized login banner is displayed to users before they authenticate to the device.

Configure a login banner (CLI)

Learn how to configure a login banner to display a custom message when users log in to a device.

Set a login banner to inform users about security policies or authorized access.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   If prompted, enter your password.

2. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Specify the login message.

   Example

   ```
   Device(config)# banner login c message c
   ```

   Example

   ```
   Device(config)# banner login $
   Access for authorized users only.
   Please enter your username and password.
   $
   ```

   - c: Enter the delimiting character of your choice (for example, the pound sign), then press the **Return** key. The delimiting character marks the beginning and end of the banner text. Any characters after the ending delimiter are discarded.
   - Message: Enter a login message up to 255 characters. The delimiting character cannot be used in the message.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

5. Verify your entries.

   Example

   ```
   Device# show running-config
   ```

6. Optional: Save your entries to the configuration file.

   Example

   ```
   Device# copy running-config startup-config
   ```

All users see the login banner when they attempt to log in.

101 Wi-Fi Alliance Agile Multiband

Topics:

- Wi-Fi alliance agile multiband

Introduces the WiFi Alliance Agile Multiband standard and explains its role in optimizing wireless network performance across multiple frequency bands.

Wi-Fi alliance agile multiband

Introduces WiFi Alliance Agile Multiband technology, detailing topology components, supported MBO components, limitations, WLAN configuration procedures, and verification methods for optimizing multiband wireless networks.

A Wi-Fi alliance agile multiband feature is a Wi-Fi interoperability certification that

- enables better use of Wi-Fi network resources by leveraging information from both networks and clients,
- certifies features defined by the IEEE 802.11k, 802.11v, and 802.11u amendments and Wi-Fi Alliance specifications, and
- improves Wi-Fi client roaming decisions and the overall network user experience.

- MBO certified clients: Devices that comply with Wi-Fi alliance testing for agile multiband interoperability.
- IEEE 802.11k, 802.11v, and 802.11u amendments: Standards that provide networks and clients with mechanisms to exchange information about network topology, quality, and roaming policies.
- RF neighbors: Other radio devices in the vicinity that could affect Wi-Fi performance, as detected and reported by access points (APs) or clients.

Additional reference information

The Wi-Fi alliance agile multiband (MBO) feature enables better use of Wi-Fi network resources. Wi-Fi networks and client devices share information to improve roaming decisions and overall Wi-Fi performance.

> **Note**
>
> This feature applies only to MBO certified clients.

This feature certifies the interoperability of features defined by the IEEE 802.11k, 802.11v, and 802.11u amendments, as well as Wi-Fi Alliance specifications. These technologies are used to exchange access point (AP) information such as band and channel preferences, link quality, and status information between an AP and a client device.

MBO focuses on the following:

- Interactions between wireless clients and APs,
- Exchange of information between APs and clients about the wireless medium (such as RF neighbors), and
- Allowing clients to work with APs to make intelligent connection decisions and improve quality of service.

Example of MBO in Action

For example, when a client device supports MBO, it can receive information from multiple APs about available channels and link quality. This information enables the client to select the best AP for optimal performance and seamless roaming.

Components of a Wi-Fi Agile Multiband topology

This section lists the main components and roles within a Wi-Fi Agile Multiband wireless infrastructure network.

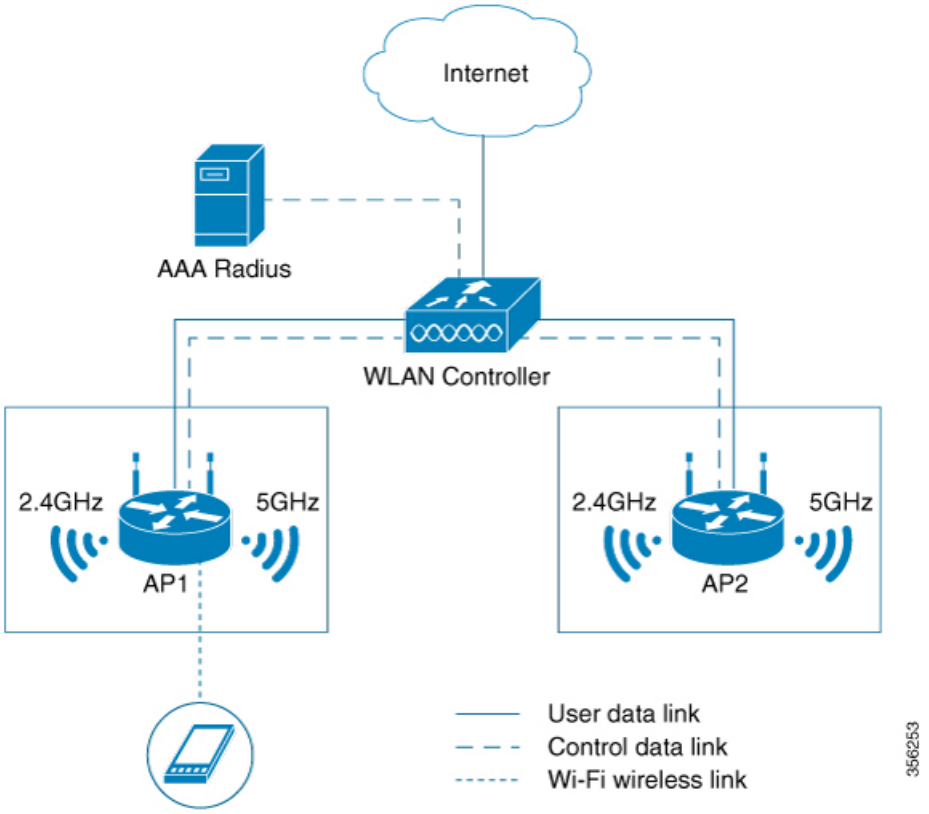
This topic describes the system topology and components involved in a Wi-Fi Agile Multiband wireless infrastructure network.

A Wi-Fi Agile Multiband wireless infrastructure network includes these components:

- Access Point (AP): A Wi-Fi Agile Multiband wireless infrastructure network contains one or more Wi-Fi Agile Multiband APs.
- WLAN Controller: A Wi-Fi Agile Multiband wireless infrastructure network contains zero or more WLAN controllers that provide centralized management and other features to the interconnected APs.
- Client Station (STA): A Wi-Fi Agile Multiband wireless infrastructure network contains zero or more Client Stations (STAs). Client Stations support only one WLAN.
- RADIUS Server: A Wi-Fi Agile Multiband wireless infrastructure network contains zero or more RADIUS Servers that provide Authentication, Authorization, and Accounting (AAA) services.

The topology may vary according to deployment requirements.

Figure 9: Wi-Fi agile multiband wireless infrastructure network



Supported MBO components

This section lists the wireless features, requirements, and behaviors supported and not supported by MBO (Multi Band Operation) components on Cisco access points.

This section provides a comprehensive summary of the features, standards, requirements, and behaviors supported by Multi Band Operation (MBO) components on Cisco access points. It enables you to determine compatibility and operational prerequisites for MBO functionality.

MBO AP capability

802.11ax access points add a new information element to the Beacon, Probe Response, Association Response, and Re Association Response frames to indicate MBO support.

> **Note**
>
> The new information element indicates that Cisco access points are not aware of cellular data status.

The MBO AP capability is enabled when an SSID is configured on the access point.

802.11k/v/r support

To support MBO, access points must support 802.11k, 802.11v, and 802.11r standard-based technologies. Each technology has specific requirements:

- 802.11k – The access point sends a list of preferred neighboring access points to the client upon request. It also sends a beacon request when a beacon report is required.
- 802.11v – The access point steers the client to a less congested access point using BSS transition. The target access point must not be in the non-preferred or non-operable channel list sent by the MBO client during the association or WNM notification request.
- 802.11r – MBO-related 802.11r capabilities are not supported.

802.11u ANQP or GAS support

For MBO, 802.11ax access points must support 802.11u ANQP or GAS.

These are the prerequisites:

- The access point responds to the ANQP request for a neighbor report ANQP element.
- Before authentication, the network must provide Layer 2 transport between the mobile device and the server to support advertisement protocol frames.

MBO beacon request

When an access point sends a beacon request to the client, an MBO-compliant client responds with a beacon report.

MBO associate disallowed IE

When Cisco access points cannot accommodate new clients, they include an **Associate Disallowed IE** in the Beacon, Probe Response, or (Re) Association Response.

Limitations of MBO

Understand the limitations while using MBO. Use these guidelines when evaluating access point compatibility with MBO features.

Only 802.11ax access points are supported.

Configure MBO on a WLAN (CLI)

Learn how to configure MBO on a WLAN using CLI commands.

Enable Mobility Optimization (MBO) on a WLAN using command-line interface commands to improve client management and network performance.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a WLAN and enter the WLAN configuration mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id ssid
   ```

   Example

   ```
   Device(config)# wlan wlan-demo 1 ssid-demo
   ```

   Note

   If you use WPA2 WLAN while configuring MBO for WLAN, you need to enable PMF in your configuration.

3. Configure MBO support on WLAN.

   Example

   ```
   Device(config-wlan)# mbo
   ```

   Note

   Use the **no mbo** command to disable MBO configuration.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wlan)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The WLAN is configured with MBO support. Wireless clients can now benefit from enhanced roaming and optimization features.

Verify MBO configuration

Verify MBO Configuration.

To view the MBO configuration, use this command:

```
Device# show wlan id 1
WLAN Profile Name     : wlan-demo
=================================================
Identifier                                       : 1
Description                                      :
Network Name (SSID)                              : ssid-demo
Status                                           : Disabled
Broadcast SSID                                   : Enabled
802.11ax parameters
    OFDMA Downlink                               : Enabled
    OFDMA Uplink                                 : Enabled
    MU-MIMO Downlink                             : Enabled
    MU-MIMO Uplink                               : Enabled
    BSS Color                                    : Enabled
    Partial BSS Color                            : Enabled
    BSS Color Code                               : 0
    BSS Target Wake Up Time                      : Enabled
    BSS Target Wake Up Time Broadcast Support    : Enabled
mDNS Gateway Status                              : Bridge
WIFI Alliance Agile Multiband      : Enabled
```

To view the non-operational or non-preferred channels, use this command:

```
Device# show wireless client mac-address 3413.e8b5.f252 detail
Client MAC Address : 3413.e8b5.f252
Client IPv4 Address : 192.165.1.53
Client IPv6 Addresses : fe80::98bb:ea89:f016:3332
Client Username: N/A
AP MAC Address : 00ee.ab18.d920
AP Name: ssap-pp
AP slot : 1
Client State : Associated
Policy Profile : prof
Flex Profile : N/A
Wireless LAN Id: 1
WLAN Profile Name: mbo_1
Wireless LAN Network Name (SSID): mbo_1
BSSID : 00ee.ab18.d92f
Connected For : 25 seconds
Protocol : 802.11ax - 5 GHz
Channel : 36
Client IIF-ID : 0xa0000001
Association Id : 1
Authentication Algorithm : Open System
Session Timeout : 1800 sec (Remaining time: 1779 sec)
Session Warning Time : Timer not running
Input Policy Name  : None
Input Policy State : None
Input Policy Source : None
Output Policy Name  : None
Output Policy State : None
Output Policy Source : None
WMM Support : Enabled
U-APSD Support : Enabled
  U-APSD value : 0
```

```
  APSD ACs    : BK, BE, VI, VO
Fastlane Support : Disabled
Client Active State : Active
Power Save : OFF
Current Rate : 1.5
Supported Rates : 9.0,18.0,36.0,48.0,54.0
Mobility:
  Move Count                  : 0
  Mobility Role               : Local
  Mobility Roam Type          : None
  Mobility Complete Timestamp : 05/15/2019 16:03:34 IST
Client Join Time:
  Join Time Of Client : 05/15/2019 16:03:34 IST
Policy Manager State: Run
Last Policy Manager State : IP Learn Complete
Client Entry Create Time : 26 seconds
Policy Type : N/A
Encryption Cipher : None
User Personal Network    : Disabled
Encrypted Traffic Analytics : No
Protected Management Frame - 802.11w : No
EAP Type : Not Applicable
VLAN : default
Multicast VLAN : 0
WFD capable : No
Managed WFD capable : No
Cross Connection capable : No
Support Concurrent Operation : No
Session Manager:
  Point of Attachment : capwap_90400001
  IIF ID             : 0x90400001
  Authorized         : TRUE
  Session timeout    : 1800
  Common Session ID: 000000000000000BB92939C5
  Acct Session ID  : 0x00000000
  Last Tried Aaa Server Details:
  Server IP :
  Auth Method Status List
  Method : None
  Local Policies:
  Service Template : wlan_svc_prof_local (priority 254)
  VLAN             : 165
  Absolute-Timer   : 1800
  Server Policies:
  Resultant Policies:
  VLAN Name         : VLAN0165
  VLAN             : 165
  Absolute-Timer   : 1800
DNS Snooped IPv4 Addresses : None
DNS Snooped IPv6 Addresses : None
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 0
Fast BSS Transition Details :
  Reassociation Timeout : 0
11v BSS Transition : Implemented
11v DMS Capable : No
QoS Map Capable : Yes
```

```
Non-Preferred Channels : 40
Non-Operable Channels : 56
FlexConnect Data Switching : N/A
FlexConnect Dhcp Status : N/A
FlexConnect Authentication : N/A
FlexConnect Central Association : N/A
Client Statistics:
  Number of Bytes Received : 0
  Number of Bytes Sent : 0
  Number of Packets Received : 0
  Number of Packets Sent : 0
  Number of Policy Errors : 0
  Radio Signal Strength Indicator : -34 dBm
  Signal to Noise Ratio : 56 dB
Fabric status : Disabled
Client Scan Reports
Assisted Roaming Neighbor List
Nearby AP Statistics:
EoGRE : No/Simple client
```

102 SNMP Traps

Topics:

- SNMP trap configuration

This chapter explains how to configure and verify SNMP trap notifications for wireless components (APs, clients, mesh, RF, rogue, mobility, and RRM) using GUI and CLI, including trapflags and host settings.

SNMP trap configuration

Explains SNMP wireless trap functionality and outlines configuration procedures for enabling traps via GUI and CLI, including access points, wireless client, mesh, RF, rogue, mobility, RRM, and general traps, with instructions for verification.

An SNMP trap is an alert message sent from SNMP-enabled devices to an SNMP manager. It does not require acknowledgment from the receiver and does not confirm receipt at the sender.

SNMP trap configuration details

To configure the controller to send SNMP notifications, you must enter at least one **snmp-server host** command. If you do not enter an **snmp-server host** command, the system does not send any notifications.

- To enable multiple hosts, specify a separate **snmp-server host** command for each host.
- You can specify multiple notification types in the command for each host.
- If you issue multiple **snmp-server host** commands for the same host and notification type (trap or inform), each new command overwrites the previous one. The system considers only the most recent **snmp-server host** command.
- For example, entering an **snmp-server host** inform command for a host and then entering another for the same host causes the second command to replace the first.

To specify which SNMP notifications are sent globally, use the **snmp-server enable traps wireless <TrapName>** command. For a host to receive wireless notifications, at least one **snmp-server enable traps wireless <TrapName>** command and the **snmp-server host** command for that host must be enabled.

- Some notification types cannot be controlled with the **snmp-server enable** command.
- Some notification types are enabled by default. For example, a few Access Point (AP)-related traps such as **crash**, **register**, and **noradiocards** are enabled by default.

> **Note**
>
> On physical Cisco Catalyst 9800 Series Wireless Controller appliances, such as Cisco Catalyst 9800-L, 9800-40, and 9800-80, the SNMP trap rate is limited to 4 traps per second. On virtual Catalyst 9800 appliances, the SNMP trap rate supports up to 300 traps per second.

Configure SNMP Traps (GUI)

Configure SNMP traps by using the GUI.

Enable and customize Simple Network Management Protocol (SNMP) trap notifications for wireless events by using the GUI.

Procedure

1. Choose **Administration** > **Management** > **SNMP**.

   The SNMP page appears. By default, SNMP mode is disabled.

   Click the **SNMP Mode** toggle button to enable or disable SNMP.

2. Choose the **Wireless Traps** tab.

By default, only the **Access Point** trap is enabled; all other SNMP wireless traps are disabled.

To enable all wireless traps, click **Enable All**.

3. Select wireless SNMP traps. Select the wireless SNMP trap to enable. Click the **Select All** check box to enable all trap flags in the trap. To enable all trap flags in the **Mesh** trap section, check the **Select All** check box in the section. Clear the **Select All** check box to deselect all trap flags.

   Note

   In the **Access Point** trap, the **Crash**, **No Radio Cards**, and **Register** trap flags are enabled by default. Select the **Broken Antenna** trap flag to detect a broken antenna. Select the **AP Stats** trap flag to enable traps for access point statistics.

4. Click **Apply**.

By configuring SNMP trap notifications, network management systems receive wireless event alerts based on your selections.

Enable AP Traps (CLI)

Enable AP traps with CLI commands.

Configure Simple Network Management Protocol (SNMP) traps on your device to monitor wireless access point (AP) events.

This configuration helps you monitor AP status and troubleshoot issues.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable wireless SNMP traps for APs.

   Example

   ```
   Device# snmp-server enable traps wireless AP
   ```

3. Enable or disable AP-related trap flags.

   Example

   ```
   Device# trapflags ap authorization
   ```

   Example

   ```
   Device# trapflags ap authorization broken-antenna crash interfaceup
   ipaddrfallback mfp mode noradiocards register
   ```

The **crash**, **noradiocards**, and **register** trap flags are enabled by default.

Your device sends SNMP traps for wireless AP events.

This configuration helps you monitor AP status and identify issues.

Enable wireless client traps (CLI)

Enable wireless client traps by using CLI commands.

Configure Simple Network Management Protocol (SNMP) traps on your device to monitor wireless client events. This setup helps you manage and troubleshoot your network.

Procedure

---

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable wireless client traps.

   Example

   ```
   Device# snmp-server enable traps wireless bsnMobileStation
   ```

3. Enable or disable 802.11ax-related trap flags for wireless clients.

   Example

   ```
   Device# trapflags client dot11 assocfail
   ```

   Example

   ```
   trapflags client dot11 assocfail associate authenticate authfail deauthenticate
    disassociate
   ```

4. Enable trap flags that are excluded for clients.

   Example

   ```
   Device# trapflags client excluded
   ```

Your device generates SNMP traps for wireless client events. Your network monitoring tools receive and process these notifications.

Enable Mesh Traps (CLI)

Enable mesh traps with CLI commands.

You can configure a device to send wireless-mesh Simple Network Management Protocol (SNMP) traps and control mesh trap notifications.

Before you begin

To enable mesh traps with CLI commands, complete each step in order.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable wireless-mesh traps.

   Example

   ```
   Device# snmp-server enable traps wireless MESH
   ```

3. Enable or disable mesh trap flags.

   Example

   ```
   Device# trapflags mesh abate-snr
   ```

   Example

   ```
   trapflags mesh abate-snr authentication-failure child-moved excessive-children
    excessive-hopcount onset-snr parent-change
   ```

The device sends wireless-mesh SNMP traps according to the mesh trap flags you selected.

Enable RF Traps (CLI)

Enable RF traps using commands.

You can configure your device to send radio frequency (RF) traps for wireless networking events using CLI commands.

Before you begin

To enable radio frequency (RF) traps using CLI commands, complete the steps below.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable wireless RF-related traps.

Example

```
Device# snmp-server enable traps wireless bsnAutoRF
```

3. Enable or disable sending traps related to radio resource management (RRM) parameter updates.

Example

```
Device# trapflags rrm-params channels tx-power
```

4. Enable or disable traps related to the radio resource management (RRM) profile.

Example

```
Device# trapflags rrm-profile coverage
```

Example

```
trapflags rrm-profile coverage interference load noise
```

The device sends RF-related SNMP traps to support network monitoring of wireless events.

Enable rogue, mobility, RRM, and general traps (CLI)

Enable rogue, mobility, RRM, and general traps using commands.

Configure SNMP traps for wireless rogue devices, mobility events, Radio Resource Management (RRM), and general controller events on your device using the CLI.

Before you begin

Use these steps to enable rogue, mobility, RRM, and general traps.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enable traps for wireless rogue.

Example

```
Device# snmp-server enable traps wireless rogue
```

3. Enable rogue AP detection trapflag.

Example

```
Device# trapflags rogue-ap
```

4. Enable rogue client detection trapflag.

   Example

   ```
   Device# trapflags rogue-client
   ```

5. Enable traps for wireless mobility.

   Example

   ```
   Device# snmp-server enable traps wireless wireless_mobility
   ```

6. Enable anchor trapflags.

   Example

   ```
   Device# trapflags anchor
   ```

7. Enable traps for wireless RRM.

   Example

   ```
   Device# snmp-server enable traps wireless RRM
   ```

8. Enable or disable the RRM parameter related traps, when the RF manager group changes.

   Example

   ```
   Device# trapflags rrm-params group
   ```

9. Enable general controller traps.

   Example

   ```
   Device# snmp-server enable traps wireless bsnGeneral
   ```

SNMP trap notifications for rogue, mobility, RRM, and general controller events are enabled on the device and will be sent to your configured SNMP management system.

Verify SNMP wireless traps

Verify SNMP Wireless Traps.

To verify the various SNMP traps enabled, use this command:

```
Device# show run | sec trapflag
trapflags ap crash
trapflags ap noradiocards
trapflags ap register
trapflags rogue-client
```

103 Disabling Clients with Random MAC Address

Topics:

- Random MAC addresses

This chapter explains how to configure a Cisco wireless controller to block clients using locally administered (random) MAC addresses: CLI steps to enable/disable the feature, verification commands, and expected outputs.

Random MAC addresses

This concept explains how wireless clients use random (locally administered) MAC addresses, why they improve privacy, and how Cisco controllers control their use within networks.

A random MAC address is a locally administered hardware address that

- is generated by a device (instead of assigned by the manufacturer),
- helps protect user privacy by making device tracking more difficult, and
- can be permitted or denied on wireless networks through controller settings.

Random MAC address denial

Wireless clients traditionally use manufacturer-assigned, globally unique MAC addresses (burn-in addresses) for network association. Devices may also use locally administered, random MAC addresses for Wi-Fi operations to improve privacy. Network administrators can deny access to these clients using this feature.

Beginning with Cisco IOS XE 17.5.1, access controllers can block clients with random MAC addresses using the **local-admin-mac deny** feature. This feature is disabled by default.

> **Note**
>
> This feature is not supported on Cisco Wave 1 access points.

Example: Denying Random MAC Address Clients

When the **local-admin-mac deny** knob is enabled on the controller, a client attempting to join the network with a random MAC address is rejected, preventing unauthorized or untraceable devices from accessing the wireless network.

For example, a phone may use a new random MAC address whenever it joins a public Wi-Fi network, which prevents tracking by Wi-Fi infrastructure or third parties.

Configure random MAC address denial (CLI)

Use this procedure to prevent clients with randomized MAC addresses from connecting to a wireless network by configuring random MAC address denial in the CLI.

Prevent unauthorized or random MAC devices from associating with your WLAN by enabling the random MAC address denial feature on your wireless controller through the CLI.

To stop the entry of clients with random MAC addresses from joining a wireless network, enable the random MAC address deny knob by following these steps.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile.

Example

```
Device(config)# wlan wlan-profile-name <1-4096> SSID-network-name
```

3. Shut down the WLAN.

Example

```
Device(config-wlan)# shutdown
```

4. Enable the random MAC address deny knob.

Example

```
Device(config-wlan)# local-admin-mac deny
```

Use the **no** form of this command to disable the feature.

5. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

6. Save the configuration, exit the configuration mode, and return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

Devices using randomized MAC addresses are prevented from joining the specified wireless SSID.

Verify denial of clients with a random MAC address

Verify Denial of Clients with a Random MAC Address.

To verify the denial of a client with a random MAC address, run the **show wlan name** *wlan-profile-name* **| begin locally** command:

```
Device# show wlan name laa | begin locally
Locally Administered Address Configuration
Deny LAA clients                          : Enabled
```

To verify if a client address is a random MAC address, run the **show wireless client mac-address** *MAC-address* **detail** command:

```
Device# show wireless client mac-address 72xx.38xx.2axx detail
Client MAC Address : 72xx.38xx.2axx
Client MAC Type    : Locally Administered Address
Client IPv4 Address      : 192.0.2.1
Client IPv6 Addresses    : 2001:DB8::71xx:27xx:a7xx:efxx
Client Username          : 72xx.38xx.2axx
```

To verify how many random MAC clients are present in the system, run the **show wireless stats client detail** command:

```
Device# show wireless stats client detail
Client Summary
------------------------------
Current Clients : 1
Excluded Clients: 0
Disabled Clients: 0
Foreign Clients : 0
Anchor Clients  : 0
Local Clients   : 1
Idle Clients    : 0
Locally Administered MAC Clients: 1
```

To display the statistics of a specific client, run the **show wlan id** *<1-4096>* **client stats** command:

```
Device# show wlan id 8 client stats
Wlan Profile Name: wlan-profile, Wlan Id: 8
Current client state statistics:
------------------------------------------------------------------------------

  Authenticating         : 0
  Mobility               : 0
  IP Learn               : 0
  WebAuth Pending        : 0
  Run                    : 1
Locally Administered MAC Clients         : 1
```

Note

Run the **show configuration wlan wlan-name** command on an AP, to view the status of the locally administered address (LAA) on the WLAN.

104 Dataplane Packet Logging

Topics:

- Dataplane packet logs

This chapter explains how to enable, configure, verify, and clear packet logging for wireless clients, including global and filtered trace buffers, CLI commands for ingress and punt-inject paths, and examples for troubleshooting DHCP, ARP, and CAPWAP issues.

Dataplane packet logs

This concept explains how dataplane packet logs capture and display packet information for onboarding and debugging wireless clients. It describes logging categories, features, and the types of network packets collected for troubleshooting.

A dataplane packet log is a diagnostic feature that

- captures connectivity and protocol packets (such as DHCP, Neighbor Discovery, and ARP) exchanged between wireless clients and the controller,
- helps you debug client onboarding and network communication issues, and
- offers unconditional (global) and selective (filtered) packet logging modes.

Expanded explanation

While onboarding wireless clients, you might encounter problems such as client IP address allocation, Address Resolution Protocol (ARP) resolution, and other connectivity issues. All of these issues require debugging.

For rapid troubleshooting, the controller logs Dynamic Host Configuration Protocol (DHCP), Neighbor Discovery, and ARP packets exchanged with wireless clients by default.

Packet logging categories

You can use packet-logging categories to capture connectivity information for wireless clients.

- **Global Trace Log**: Captures client connectivity information and is enabled by default.
- **Filtered Trace Log**: Lets you log packets to a filtered buffer. You enable filters using **debug** commands. Filters capture only specific packet types or packets associated with particular client MAC addresses.

Features of dataplane packet logs

- You can add or remove filters to customize the packet types captured. By default, DHCP, Neighbor Discovery, and ARP packets are always included.
- You can use display filters to select a subset of logged packets to review.
- You can use packet-logging data to view detailed debugging information such as client MAC address, client IP address, VLAN, interface, packet type, and time delta.

Dataplane packet log examples

- For onboarding issues, enable filtered trace logs with specific filters to capture only DHCP packets from a problematic client.
- Use display filters to narrow the logs to packets exchanged on a particular VLAN or interface.

Enable or disable debug level (CLI)

Use this procedure to enable or disable the debug level for wireless datapath trace-buffer using CLI commands.

Adjust the debug level for tracing wireless datapath operations on the device.

Procedure

1. Enable privileged EXEC mode.

Example

```
Device> enable
```

Enter your password, if prompted.

2. Enable the debug level to trace global and filtered logic.

Example

```
Device# [no] debug platform hardware chassis active qfp feature wireless
datapath trace-buffer debug-level {all | error | info | trace | warning}
```

Example

```
Device# debug platform hardware chassis active qfp feature wireless datapath
 trace-buffer debug-level all
```

Use the **no** form of this command to disable the feature.

The specified debug level is enabled or disabled for the wireless datapath trace-buffer.

Enable packet logging in the global and filtered buffers in the ingress-path (CLI)

Use this task to enable packet logging in the global and filtered buffers in the ingress-path using CLI commands.

Enable detailed packet logging in the ingress-path of your device to troubleshoot and analyze network traffic.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device> enable
```

Enter your password if prompted.

2. Enable the quantum flow processor for the global-trace buffer in the ingress-path.

Example

```
Device# [no] debug platform hardware chassis active qfp feature wireless
datapath trace-buffer ingress global-trace
```

Use the **no** form of this command to disable the feature.

3. Enable Control and Provisioning of Wireless Access Points (CAPWAP) packet logging to the filtered-trace buffer in the ingress-path.

Example

```
Device# [no] debug platform hardware chassis active qfp feature wireless
datapath trace-buffer ingress filtered-trace capwap ipv4 209.165.200.224/27
```

Use the **no** form of this command to disable the feature.

4. Enable logging of wireless client packet information to the filtered-trace buffer in the ingress-path.

Example

```
Device# [no] debug platform hardware chassis active qfp feature wireless
datapath trace-buffer ingress filtered-trace wlclient {ipv6-nd | ipv6-ra |
mac-address H.H.H}
```

Example

```
Device# [no] debug platform hardware chassis active qfp feature wireless
datapath trace-buffer ingress filtered-trace wlclient ipv6-nd
```

Use the **no** form of this command to disable the feature.

The device captures and logs packet information for analysis in the global-trace and filtered-trace buffers of the ingress-path.

Enable packet logging in global and filtered buffers for the punt-inject path (CLI)

Learn how to enable packet logging in the global and filtered buffers for the punt-inject path using CLI commands.

Enable detailed packet logging to monitor traffic for troubleshooting on specific data paths.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device> enable
```

Enter your password if prompted.

2. Enable the quantum flow processor in the global trace buffer for the punt-inject path.

Example

```
Device# [no] debug platform hardware chassis active qfp feature wireless
datapath trace-buffer punt-inject global-trace
```

Use the **no** form of this command to disable the feature.

3. Enable CAPWAP packet logging to the filtered trace buffer for the punt-inject path.

Example

```
Device# [no] debug platform hardware chassis active qfp feature wireless
datapath trace-buffer punt-inject filtered-trace capwap {ipv4 A.B.C.D | ipv6
 X:X:X:X::X | keepalive}
```

Example

```
Device# [no] debug platform hardware chassis active qfp feature wireless
datapath trace-buffer punt-inject filtered-trace capwap ipv4 209.165.200.224/27
```

Use the **no** form of this command to disable the feature.

4. Enable wireless client packet logging to the filtered trace buffer for the punt-inject path.

Example

```
Device# [no] debug platform hardware chassis active qfp feature wireless
datapath trace-buffer punt-inject filtered-trace wlclient {ipv6-nd | ipv6-ra
 | mac-address H.H.H}
```

Example

```
Device# [no] debug platform hardware chassis active qfp feature wireless
datapath trace-buffer punt-inject filtered-trace wlclient ipv6-nd
```

Use the **no** form of this command to disable the feature.

You have enabled packet logging for the global and filtered trace buffers in the punt-inject path.

Verify dataplane packet logging

Verify Dataplane Packet Logging.

To show trace buffer-configured conditions, use this command:

```
Device# show platform hardware chassis active qfp feature wireless trace-buffer
 ingress conditions
LogTrace Event: Enabled
Trace wlclient-MACs:
    8c85.90ee.ca92
allow_all_AP_kalives: enabled
AP_kalive cnt=1, AP_kalive6 cnt=0
  IP0: 49.1.0.73
```

To view all the log entries in the filtered trace buffer, use this command:

```
Device# show platform hardware chassis active qfp feature wireless trace-buffer
 ingress filtered-trace all
Trace wlclient-MACs: 8c85.90ee.ca92
  Trace-Buffer for Ingress: Enabled
  Total allocated global-log buffer: 16384
  Total allocated filtered-log v=buffers: 4096

<0 KEEP_ALIVE: CAPWAP peer=49.1.0.73 udp=5256 local=49.1.1.2 udp=5247 vlan=49,
 dt=213207 c=0
<1 KEEP_ALIVE: CAPWAP peer=49.1.0.73 udp=5256 local=49.1.1.2 udp=5247 vlan=49,
 dt=213236 c=0
<2 KEEP_ALIVE: CAPWAP peer=49.1.0.73 udp=5256 local=49.1.1.2 udp=5247 vlan=49,
 dt=213264 c=0
<3 KEEP_ALIVE: CAPWAP peer=49.1.0.73 udp=5256 local=49.1.1.2 udp=5247 vlan=49,
 dt=213293 c=0
<4 KEEP_ALIVE: CAPWAP peer=49.1.0.73 udp=5256 local=49.1.1.2 udp=5247 vlan=49,
```

```
 dt=213321 c=0
<5 KEEP_ALIVE: CAPWAP peer=49.1.0.73 udp=5256 local=49.1.1.2 udp=5247 vlan=49,
 dt=213350 c=0
```

To view the number of entries based on a count, use this command:

```
Device# show platform hardware chassis active qfp feature wireless trace-buffer
 ingress filtered-trace 3
Trace wlclient-MACs: 8c85.90ee.ca92
  Trace-Buffer for Ingress: Enabled
  Total allocated global-log buffer: 16384
  Total allocated filtered-log v=buffers: 4096

<18 KEEP_ALIVE: CAPWAP peer=49.1.0.73 udp=5256 local=49.1.1.2 udp=5247 vlan=49,
 dt=213720 c=0
<19 KEEP_ALIVE: CAPWAP peer=49.1.0.73 udp=5256 local=49.1.1.2 udp=5247 vlan=49,
 dt=213748 c=0
<20 KEEP_ALIVE: CAPWAP peer=49.1.0.73 udp=5256 local=49.1.1.2 udp=5247 vlan=49,
 dt=213777 c=0

>> 3 entries displayed
   21 entries found in filtered-log buffer
   21 entries ever collected for filtered-log buffer
```

Clear logs and conditions in global and filtered trace buffers

Clear Logs and Conditions in Global and Filtered Trace Buffers.

To clear conditions and logs in the global and filtered trace buffers, use these commands:

```
Device# clear platform hardware chassis active qfp feature wireless trace-buffer
 ingress all
Trace, clear all trace configuration & buffer.

Device# clear platform hardware chassis active qfp feature wireless trace-buffer
 ingress conditions
Trace, clear trace configuration

Device# clear platform hardware chassis active qfp feature wireless trace-buffer
 ingress filtered-trace
Trace, clear trace Q

Device# clear platform hardware chassis active qfp feature wireless trace-buffer
 ingress global-trace
Trace, clear trace global Q
```

105 Streaming Telemetry

Topics:

- Streaming telemetry
- Client event and SSID telemetry filters

This chapter explains configuring and troubleshooting native streaming telemetry for Cisco wireless controllers and APs, including gather points, transports (gNMI/gRPC), subscription/session modes, scale limits, on-change support, zero-trust setup, and SSID/client filtering.

Streaming telemetry

This concept explains how telemetry streams provide a modern, structured approach for monitoring and troubleshooting network health by transmitting configuration and operational data from the controller to remote management stations.

A telemetry stream is a network monitoring method that:

- streams configuration and operational data from controllers in real time
- transmits data in a structured format to remote management stations
- enables efficient monitoring and troubleshooting of Wi-Fi and system health
- Telemetry feed: a specific subscription within a telemetry stream that delivers the selected data elements.
- Collector: the remote management station that receives telemetry data for analysis.

Streaming telemetry capabilities and usage

You can use streaming telemetry for scalable and efficient monitoring of access points and clients through native Cisco wireless models.

- You can enhance telemetry support to monitor up to 1,000 access points and 15,000 clients.
- Use a single collector setup to subscribe to the required XPaths.
- Use a telemetry feed to subscribe to data elements and monitor access points and clients.
- You receive data through the native Cisco wireless models.

Gather points

This concept explains how gather points serve as the top-level XPaths that define the smallest unit of data export in telemetry, enabling efficient and comprehensive data subscriptions.

A gather point is a data grouping mechanism that

- represents a top-level XPath in the telemetry data model
- acts as the smallest unit of data exported by a target device, and
- ensures subscriptions receive all data leaves defined beneath that point.

Supported gather points and subscription intervals

This table lists supported gather points and the minimum subscription intervals for each:

Table 102: Supported gather points and subscription intervals

Supported gather point	Subscription interval
wireless-access-point-oper:access-point-oper-data/ethernet-mac-wtp-mac-map	>=15 mins
/wireless-access-point-oper:access-point-oper-data/capwap-data	>=15 mins
/wireless-access-point-oper:access-point-oper-data/cdp-cache-data/	>=15 mins
/wireless-access-point-oper:access-point-oper-data/radio-oper-stats	>=60 secs
/wireless-access-point-oper:access-point-oper-data/radio-oper-data	>=180 secs

Supported gather point	Subscription interval
/wireless-access-point-oper:access-point-oper-data/oper-data	>=180 secs
/wireless-rrm-oper:rrm-oper-data/rrm-measurement	>=180 secs
/wireless-client-oper:client-oper-data/dot11-oper-data	>=180 secs
/wireless-client-oper:client-oper-data/common-oper-data	>=15 mins
/wireless-client-oper:client-oper-data/policy-data	>=60 secs
/wireless-client-oper:client-oper-data/sisf-db-mac/ipv4-binding/ip-key/ip-addr	>=15 mins
/wireless-client-oper:client-oper-data/traffic-stats	>=180 secs
/lldp-ios-xe-oper:lldp-entries/lldp-state-details	>=60 secs
/device-hardware-xe-oper:device-hardware-data/device-hardware	>=15 mins
/wireless-mobility-oper:mobility-oper-data/mobility-node-data/ulink-status	>=60 secs
/process-cpu-ios-xe-oper:cpu-usage/cpu-utilization/one-minute	>=60 secs
/platform-sw-ios-xe-oper:cisco-platform-software/control-processes	>=60 secs
/environment-ios-xe-oper:environment-sensors/environment-sensor	>=60 secs
/lldp-ios-xe-oper:lldp-entries/lldp-intf-details	>=60 secs
/interfaces-ios-xe-oper:interfaces/interface	>=60 secs
/platform-ios-xe-oper:components/component	>=60 secs
/mdt-oper-v2:mdt-oper-v2-data	>=60 secs
/wireless-access-point-oper:access-point-oper-data/radio-oper-data/radio-band-info	>=180 secs

Example of gather point subscription

When you subscribe to an XPath such as */access-point-operdata/radio-oper-data/vap-oper-config/ssid*, which is part of the gather point */access-point-operdata/radio-oper-data/vap-oper-config*, the reply will include all attributes that are part of the gather point, such as AP-VAP-ID, SSID, and WLAN ID.

Subscription

This concept explains how subscriptions connect one or more gather points with destinations, enabling cadence-based data streaming using Multicast Default Transport (MDT).

A subscription is a configuration that

- binds one or more gather points and destinations
- streams data for each gather point using MDT, and
- operates at a configured cadence (frequency).

Transport

This concept explains the transport protocols available for publisher-to-receiver connections and how they determine the method and encoding of data transmission in the system.

A transport is a protocol that:

- enables a connection between a publisher and a receiver
- determines how data is transmitted
- operates independently from the management protocol used for configured subscriptions
- The choice of transport protocol affects compatibility and the data formats exchanged between systems.
- The supported transport protocols are gNMI (gRPC Network Management Interface) and gRPC (Google Remote Procedure Call).
- The gNMI transport protocol supports JSON encoding of data.
- The gRPC transport protocol supports key value Google Protocol Buffers (kvGPB: key value Google Protocol Buffers) encoding.

Scale considerations

This topic explains the scale numbers that are applicable to the native model for an XPath set.

This table provides the scale numbers that are applicable to the native model for an XPath set.

Table 103: Scaling considerations to the native model

Attribute	Scale
AP	4000
Client	15000
SSID per AP	6
BSSID per AP	12
Neighbors per AP	60 (30x2)
Number of physical neighbor APs	49
Number of neighbor records	60000 records

Session

This concept explains what sessions are in network telemetry and describes their core attributes and available initiation modes.

A session is a telemetry connection that

- enables data exchange between the controller and the receiver
- may be initiated by the user, and
- supports two modes: gNMI Dial-In and gRPC Dial-Out.

Session initiation modes

You can initiate a session between the controller and the receiver using these modes:

- gNMI Dial-In Mode
- gRPC Dial-Out Mode

gNMI dial-in modes

This concept explains how gNMI dial-in modes enable a telemetry receiver to initiate and manage dynamic subscriptions to a controller for real-time telemetry data streaming.

A gNMI dial-in mode is a telemetry subscription mechanism that:

- enables a Model-Driven Telemetry (MDT) receiver to connect directly to the controller
- allows the receiver to dynamically subscribe to one or more gather points or subscriptions, and
- streams telemetry data from the controller to the receiver over the established session until the subscription is canceled or the session is terminated

Reference Information for gNMI Dial-In Mode

In gNMI dial-in mode, the MDT receiver (client) initiates a connection ("dials in") to the controller (server). After connecting, the receiver can select and change the telemetry data it wants to receive by subscribing to specific gather points (subscriptions).

The controller streams the selected telemetry data to the receiver over the same session. Subscriptions in dial-in mode are temporary. They end when the receiver cancels the subscription or disconnects. This differs from persistent, controller-initiated subscriptions.

Example of gNMI Dial-In Mode

A network operations system uses gNMI dial-in mode to gather real-time interface statistics from a controller. The system connects as a client, subscribes to the desired telemetry data, and continues receiving updates until it cancels the subscription and terminates the session.

gRPC dial-out modes

This concept explains gRPC dial-out modes. It describes how the controller acts as a client to stream telemetry data to receivers by using persistent subscriptions.

A gRPC dial-out mode is a telemetry delivery method that:

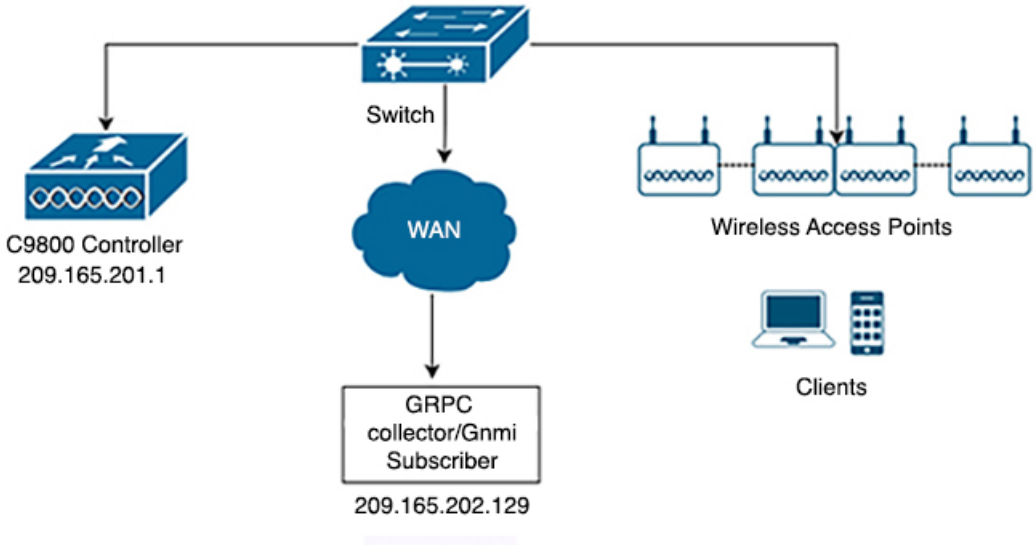
- enables the controller to act as a client and initiate outbound connections to receivers
- binds gather points and destinations into subscriptions that continually deliver telemetry data
- streams data to receivers and continually maintains sessions for each configured destination

Additional reference information

In dial-out mode, the controller contacts the receiver, which acts as a server. Configure gather points and destinations and link them into one or more subscriptions.

The controller establishes and maintains sessions with each destination in the subscription to ensure continuous data streams. Subscriptions in this mode remain persistent.

Figure 10: Telemetry Session



Note

If you remove a VRF (Virtual Routing and Forwarding) definition and link it to a telemetry source interface, configure these commands with the name server settings to ensure DNS resolution works correctly:

```
ip domain lookup vrf <vrf name> source-interface <>
```

```
ip name-server vrf <vrf name> <domain server ip>
```

Configure telemetry on controllers

This process shows you how to configure telemetry on a controller. You enable the gRPC Network Management Interface (gNXI) in secure and insecure modes. You also verify subscriptions and manage configured subscriptions.

Summary

You enable gNXI in different modes. You verify the status of telemetry subscriptions. You manage existing subscriptions on the controller.

Workflow

You can configure telemetry on a controller by completing these steps.

1. Enable gNXI in insecure mode.
2. Enable gNXI in secure mode.
3. Verify the status of the subscription.
4. Manage configured subscriptions.

Result

The controller is configured for telemetry. You can validate and manage the desired subscriptions.

This configuration supports real-time monitoring and reporting.

Enable the gRPC Network Operations Interface (gNXI) in Insecure Mode (CLI)

Enable the gNXI server in insecure mode using CLI commands.

Enable the gNXI server in insecure mode. Use this mode for testing or non-production environments that do not require certificate-based security.

The gRPC Network Operations Interface (gNXI) is a network management protocol. In insecure mode, the server does not use certificate-based authentication or encrypted communication.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Start the gNXI process.

   Example

   ```
   Device(config)# gnxi
   ```

4. Enable the gNXI server in insecure mode.

   Example

   ```
   Device(config)# gnxi server
   ```

5. Set the gNXI port.

   Example

   ```
   Device(config)# gnxi port-number
   ```

   The default insecure gNXI port is 9339.

6. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

7. View the status of the gNXI server.

Example

```
Device# show gnxi state
```

The gNXI server operates in insecure mode. It listens for incoming connections on the specified port.

Verify the status using the **show gnxi state** command.

This example shows sample output from the **show gnxi state** command:

```
Device# show gnxi state
State Status
--------------------------------
Enabled Up
```

Enable gNXI in Secure Mode (CLI)

Enable gNXI in secure mode using CLI commands. This configuration provides secure communication and authentication between your device and gNXI clients.

Enable gNXI in secure mode so your device securely interfaces with gNXI clients. This setup uses certificate-based authentication and encrypted communication.

gRPC Network Operations Interface (gNXI) is a modern protocol for network management. Secure mode encrypts and authenticates all gNXI communication.

Use this task if you deploy devices in environments where security is critical or compliance mandates certificate-based authentication for management protocols.

Procedure

---

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Start the gNXI process.

   Example

   ```
   Device(config)# gnxi
   ```

4. Enable the gNXI server in secure mode.

Example

```
Device(config)# gnxi secure-server
```

5. Specify the trustpoint and certificate set that gNXI uses to authenticate clients.

   Example

   ```
   Device(config)# gnxi secure-trustpoint trustpoint-name
   ```

6. Optional: Enable client certificate authentication.

   Example

   ```
   Device(config)# gnxi secure-client-auth
   ```

   gNXI authenticates client certificates against the root certificate you configure.

7. Optional: Set the gNXI port number.

   Example

   ```
   Device(config)# gnxi secure-port
   ```

   If you do not enable secure mode, gNXI uses port 9,339 by default.

8. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

9. Display the gNXI server's status to confirm your configuration.

   Example

   ```
   Device# show gnxi state
   ```

The device enables gNXI in secure mode using the configured trustpoint and port. The device allows secure, certificate-authenticated connections from gNXI clients.

This is sample output from the **show gnxi state** command:

```
Device# show gnxi state
State Status
--------------------------------
Enabled Up
```

Verify the status of a telemetry subscription on a controller

This topic explains how to verify the status of a telemetry subscription on a Cisco Catalyst 9800 Series Wireless Controller using various show commands.

To verify the status of a subscription, use these commands:

```
Device# show telemetry ietf subscription all
Device# show telemetry ietf subscription 101
Device# show telemetry ietf subscription 101 detail
Device# show telemetry ietf subscription 101 receiver
Device# show telemetry internal connection
Device# show telemetry internal subscription all stats
Device# show telemetry receiver all
Device# show telemetry receiver name <receivers-name>
Device# show telemetry connection all
```

Manage configured subscriptions on a controller (CLI)

Learn how to manage configured subscriptions on a Cisco Catalyst 9800 Series Wireless Controller using the CLI.

Configure telemetry subscriptions to collect data from supported YANG models and stream data to remote receivers on a controller.

Use the **show platform software ndbman switch** { *switch-number* | **active** | **standby** } **models** command to display the list of YANG models that support on-change subscription.

> **Note**
>
> Currently, you can only use the gRPC protocol for managing configured subscriptions.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password, if prompted.

2. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Create a telemetry subscription and enter telemetry-subscription mode.

   Example

   ```
   Device(config)#  telemetry ietf subscription id
   ```

4. Specify the Key-value Google Protocol Buffers (kvGPB) encoding.

Example

```
Device(config-mdt-subs)# encoding encode-kvgpb
```

5. Specify the XPath filter for the subscription.

Example

```
Device(config-mdt-subs)# filter xpath path
```

6. Configure the source IP address on the telemetry subscription interface.

Example

```
Device(config-mdt-subs)# source-address { A.B.C.D | X:X:X:X::X }
```

7. Configure a stream for the subscription.

Example

```
Device(config-mdt-subs)# stream yang-push path
```

8. Configure a periodic update policy for the subscription.

Example

```
Device(config-mdt-subs)# update-policy periodic period
```

9. Configure a periodic update policy for the subscription.

Example

```
Device(config-mdt-subs)# receiver ip address ip-address receiver-port  protocol
  protocol profile name
```

Example

```
Device(config-mdt-subs)# receiver ip address 209.165.201.1 protocol grpc-tcp
```

10. Exit telemetry-subscription configuration mode and return to privileged EXEC mode.

Example

```
Device(config-mdt-subs)# end
```

The telemetry subscription is configured, and the controller pushes YANG model data to the defined receiver using the gRPC protocol.

Configure zero trust telemetry

This process describes how to configure zero trust telemetry on a Cisco Catalyst 9800 Series Wireless Controller, including defining a protocol, naming a receiver, and setting up a telemetry subscription.

Summary

Zero trust telemetry provides enhanced monitoring and security by collecting and transmitting telemetry data from the controller.

Workflow

The stages describe the steps to configure zero trust telemetry on the controller.

1. Define a protocol for telemetry communication on the controller.
2. Define a named receiver to receive telemetry data.
3. Configure the telemetry subscription to enable data collection and transmission.

Define a protocol (CLI)

Use this procedure to configure a telemetry protocol profile for secure telemetry communication using gRPC.

Securely configure telemetry data export protocols by defining protocol profiles with appropriate trustpoints.

Before you begin

Define crypto trustpoints (CAforMDTserver and IDforWLCclient) and certificates before the telemetry configuration.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the protocol gRPC profile and enter gRPC profile name.

   Example

   ```
   Device(config)# telemetry protocol grpc profile profile-name
   ```

3. Add the server CA trustpoint.

   Example

   ```
   Device(config-mdt-protocol-grpc-profile)# ca-trustpoint CAforMDTserver
   ```

4. Add the client ID trustpoint.

   Example

   ```
   Device(config-mdt-protocol-grpc-profile)# id-trustpoint wlc-id-trustpoint
   ```

The telemetry protocol profile is configured with the specified trustpoints.

Define a named receiver (CLI)

Use this procedure to define a named receiver for telemetry configuration.

Define a named receiver and associate an FQDN and TLS profile for telemetry data delivery.

This procedure defines:

- FQDN DNS name
- Crypto protocol definition

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the receiver name.

   Example

   ```
   Device(config)# telemetry receiver protocol receiver-name
   ```

3. Add FQDN DNS name of receiver.

   Example

   ```
   Device(config-mdt-protocol-receiver)# host name FQDN-receiver
   ```

4. Define the gRPC TLS profile named mtlsyang.

   Example

   ```
   Device(config-mdt-protocol-receiver)# protocol grpc-tls profile profile-name
   ```

The device is now configured with a named telemetry receiver with the specified parameters.

Configure a telemetry subscription (CLI)

Use this procedure to configure a telemetry subscription on the device using the CLI. This enables streaming of selected network data for monitoring by external collectors.

Set up telemetry subscriptions to send device data to external receivers for analysis and monitoring.

This procedure configures:

- Xpath
- Named receiver
- Protocol

Procedure

1. Enable privileged EXEC mode. Enter your password, if prompted.

Example

```
Device# enable
```

2. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Create a telemetry subscription and enter telemetry-subscription mode.

   Example

   ```
   Device(config)# telemetry ietf subscription id
   ```

4. Specify the key-value Google Protocol Buffers (kvGPB) encoding.

   Example

   ```
   Device(config-mdt-subs)# encoding encode-kvgpb
   ```

5. Specify the XPath filter for the subscription.

   Example

   ```
   Device(config-mdt-subs)# filter xpath path
   ```

6. Configure the source IP address on the telemetry subscription interface.

   Example

   ```
   Device(config-mdt-subs)# source-address ip-address  A.B.C.D | X:X:X:X::X
   ```

7. Configure a stream for the subscription.

   Example

   ```
   Device(config-mdt-subs)# stream yang-push
   ```

8. Configure a periodic update policy for the subscription.

   Example

   ```
   Device(config-mdt-subs)# update-policy periodic period
   ```

9. Configure type protocol for receiver.

   Example

   ```
   Device(config-mdt-subs)# receiver-type protocol
   ```

10. Specify the receiver name.

Example

```
Device(config-mdt-subs)# receiver name collector
```

11. Exit telemetry-subscription configuration mode and return to privileged EXEC mode.

Example

```
Device(config-mdt-subs)# end
```

The device is configured to stream selected data to the named receiver using the specified encoding and update policy.

On-Change Telemetry Support

On-Change Telemetry Support details.

From Cisco IOS XE Cupertino 17.7.1 onwards, on-change telemetry support is provided to a subset of XPaths.

Supported XPaths for on-change subscription

This topic explains the supported XPaths for on-change subscription, including those introduced in various Cisco IOS XE YANG models and their corresponding gather points, EXEC CLIs, and telemetry feeds.

The tables list the supported XPaths for on-change subscription. Each section provides the relevant gather points, EXEC CLIs, or telemetry feeds associated with Cisco IOS XE YANG models.

Supported gather points and XPaths

This table lists the supported XPaths for on-change subscription.

Gather points	XPaths
/access-point-oper-data/radio-operdata/	/access-point-oper-data/radio-operdata/phy-ht-cfg/cfg-data/curr-freq /access-point-oper-data/radio-operdata/phy-ht-cfg/cfg-data/chan-width /access-point-oper-data/radio-oper-data/current-band-id

Gather points	XPaths
/access-point-oper-data/capwap-data	/access-point-oper-data/capwap-data/name
	/access-point-oper-data/capwapdata/device-detail/wtp-version/sw-ver/version
	/access-point-oper-data/capwap-data/device-detail/wtp-version/sw-ver/release
	/access-point-oper-data/capwapdata/device-detail/wtp-version/sw-ver/maint
	/access-point-oper-data/capwapdata/device-detail/wtp-version/sw-ver/build
	/access-point-oper-data/capwap-data/ap-state/ap-operation-state
	/access-point-oper-data/capwapdata/device-detail/static-info/board-data/wtp-serial-num
/access-point-oper-data/oper-data	/access-point-oper-data/oper-data/ap-ip-data/ap-ip-addr
	/access-point-oper-dat/oper-data/ap-pow/power-type

Supported gather points and XPaths (Cisco-IOS-XE-wireless-ap-global-oper-transform.yang)

Gather oints	XPaths
/ap-global-oper-data/ap-join-stats/wtp-mac	/ap-global-oper-data/ap-join-stats/ap-join-info/ap-ethernet-mac
	/ap-global-oper-data/ap-join-stats/ap-join-info/ap-name
	/ap-global-oper-data/ap-join-stats/ap-join-info/ap-ip-addr
	/ap-global-oper-data/ap-join-stats/ap-join-info/is-joined
	/ap-global-oper-data/ap-join-stats/ap-join-info/last-error-type
	/ap-global-oper-data/ap-join-stats/ap-disconnect-reason

Supported gather points and XPaths (Cisco-IOS-XE-aaa-oper.yang)

Gather Points	Xpaths
/aaa-data/aaa-radius-stats/	/aaa-data/aaa-radius-stats/radsec-pkt-cnt-idletime
	/aaa-data/aaa-radius-stats/radsec-send-hs-start-cnt
	/aaa-data/aaa-radius-stats/radsec-hs-success-cnt
	/aaa-data/aaa-radius-stats/radsec-total-tx-pkt-cnt
	/aaa-data/aaa-radius-stats/radsec-total-rx-pkt-cnt
	/aaa-data/aaa-radius-stats/radsec-total-conn-rst-cnt
	/aaa-data/aaa-radius-stats/radsec-conn-rst-cnt-idle
	/aaa-data/aaa-radius-stats/radsec-conn-rst-cnt-noresp
	/aaa-data/aaa-radius-stats/radsec-conn-rst-cnt-malpkt
	/aaa-data/aaa-radius-stats/radsec-conn-rst-cnt-err
	/aaa-data/aaa-radius-stats/radsec-conn-rst-cnt-peer
	/aaa-data/aaa-radius-stats/num-aaa-lib-inst
	/aaa-data/aaa-radius-stats/server-detail

Gather Points	Xpaths
/aaa-data/aaa-radius-global-stats	/aaa-data/aaa-radius-global-stats/access-rejects
	/aaa-data/aaa-radius-global-stats/access-accepts
	/aaa-data/aaa-radius-global-stats/authen-responses-seen
	/aaa-data/aaa-radius-global-stats/authen-with-response
	/aaa-data/aaa-radius-global-stats/authen-without-response
	/aaa-data/aaa-radius-global-stats/authen-avg-response-delay
	/aaa-data/aaa-radius-global-stats/authen-max-response-delay
	/aaa-data/aaa-radius-global-stats/authen-timeouts
	/aaa-data/aaa-radius-global-stats/authen-duplicate-id
	/aaa-data/aaa-radius-global-stats/authen-bad-authenticators
	/aaa-data/aaa-radius-global-stats/acct-responses-seen
	/aaa-data/aaa-radius-global-stats/acct-with-response
	/aaa-data/aaa-radius-global-stats/acct-without-response
	/aaa-data/aaa-radius-global-stats/acct-avg-response-delay
	/aaa-data/aaa-radius-global-stats/acct-max-response-delay
	/aaa-data/aaa-radius-global-stats/acct-timeouts
	/aaa-data/aaa-radius-global-stats/acct-duplicate-id
	/aaa-data/aaa-radius-global-stats/acct-bad-authenticators
	/aaa-data/aaa-radius-global-stats/stats-time

Supported EXEC CLIs and XPaths (Cisco-IOS-XE-wireless-mesh-rpc.yang)

EXEC CLI	XPath
ap name *<ap-name>* [no] mesh ethernet [0|1|2|3] mode trunk vlan allowed *<vlan-id>*	/set-rad-mesh-ethernet-trunk-allowed-vlan
ap name *<ap-name>* [no] mesh ethernet [0|1|2|3] mode trunk vlan native	/set-rad-mesh-ethernet-trunk-native-vlan
ap name *<ap-name>* mesh linktest *<dst AP MAC> <data rate> <packets/sec> <packet size> <duration>*	/exec-linktest-ap
ap name *<ap-name>* [no] mesh ethernet [0|1|2|3] mode access *<vlan-id>*	/set-rad-mesh-ethernet-access-vlan
ap name *<ap-name>* [no] mesh block-child	/set-rad-mesh-block-child
ap name *<ap-name>* [no] mesh vlan-trunking	/set-rad-mesh-trunking

EXEC CLI	XPath
ap name *<ap-name>* [no] mesh daisy-chaining strict-rap	/set-rad-mesh-daisy-chain-strict-rap
ap name *<ap-name>* [no] mesh daisy-chaining	/set-rad-mesh-daisy-chain-mode
ap name *<ap-name>* [no] mesh parent preferred	/set-rad-mesh-preferred-parent-ap
ap name *<ap-name>* mesh backhaul rate dot11ac mcs *<mcs-index> ss <1-4>*	/set-rad-mesh-bhaul-tx-rate
ap name *<ap-name>* mesh backhaul radio dot11 5ghz [slot *<slot-id>*]	/set-rad-mesh-bhaul-radio
ap name *<ap-name>* mesh security psk provisioning delete	/set-rad-mesh-security-psk-provisioning-delete
ap name *<ap-name>* mesh vlan-trunking native *<vlan-id>*	/set-rad-mesh-trunking-vlan

Supported EXEC CLIs and XPaths (Cisco-IOS-XE-aaa-oper.yang)

EXEC CLIs	XPaths
show radius statistic	/aaa-data/aaa-radius-global-stats/

Troubleshooting Telemetry Support

This section lists recommended commands, checks, and analysis steps for collecting telemetry diagnostics from Cisco Catalyst 9800 Series Wireless Controllers, with an emphasis on gRPC telemetry troubleshooting and supporting TAC case resolution.

This document outlines a set of commands and procedures to help you systematically gather data and diagnose telemetry-related issues.

Key factors to consider before troubleshooting

- Provide a clear problem description.
- Identify any recent changes in the network.
- Determine the previous known good state.
- Assess the overall impact of the problem.

Timestamp logging recommendation

Run all the **show** commands with **show clock** or **terminal exec prompt timestamp** once to log timestamps automatically.

General commands for all telemetry cases

For every issue, run these commands:

1. `Device# terminal length 0`
2. `Device# show clock`
3. `Device# show tech-support wireless`
4. `Device# request platform software trace archive last 1`

Basic checks for telemetry function

1. Verify that the requisite processes (particularly pubd) are running using this commands:

```
show platform software yang-management process
```

2. Capture and validate the telemetry-specific configuration using this command:

```
show running-config | section telemetry
```

3. Check the validity of any subscriptions using this command:

```
show telemetry ietf subscription all
```

4. Check the validity of any named receivers using this command:

```
show telemetry receiver all
```

5. Verify the telemetry subscription states using this command:

```
show telemetry internal subscription all stats
```

Checking telemetry connectivity

1. Check the state of the subscription receiver using this command:

```
show telemetry ietf subscription <id> receiver
```

2. Check the state of telemetry connections using the following command:

```
show telemetry connection all
```

3. Check which subscriptions use a particular connection using the following command:

```
show telemetry connection <index> subscription
```

Capture debug logs for telemetry

1. Enable these debug options:

```
set platform software trace mdt-pubd chassis active r0 mdt-ctrl debug
set platform software trace mdt-pubd chassis active r0 pubd debug
set platform software trace mdt-pubd chassis active r0 green-be debug
set platform software trace mdt-pubd chassis active r0 green-fe debug
```

```
set platform software trace mdt-pubd chassis active r0 dbal debug
set platform software trace mdt-pubd chassis active r0 tdllib debug
set platform software trace ios chassis active r0 green-be debug
set platform software trace ios chassis active r0 dbal debug
set platform software trace ios chassis active r0 tdllib debug
```

2. Recreate the problem.
3. Collect debug logs:

```
request platform software trace archive last <days>
```

4. Disable debugging using these commands:

```
set platform software trace mdt-pubd chassis active r0 mdt-ctrl notice
set platform software trace mdt-pubd chassis active r0 pubd notice
set platform software trace mdt-pubd chassis active r0 green-be notice
set platform software trace mdt-pubd chassis active r0 green-fe notice
set platform software trace mdt-pubd chassis active r0 dbal notice
set platform software trace mdt-pubd chassis active r0 tdllib notice
set platform software trace ios chassis active r0 green-be notice
set platform software trace ios chassis active r0 dbal notice
set platform software trace ios chassis active r0 tdllib notice
```

General telemetry diagnostics

To capture general telemetry diagnostics, use this command:

```
show telemetry internal diagnostics
```

Generate a process core

Generate a core using these commands:

1. `show clock`
2. `configure terminal`
3. `service internal`
4. `end`
5. `request platform software process core mdt-pubd chassis active r0`

Disable logging

Disable the logging using these commands:

1. `configure terminal`
2. `no service internal`
3. `end`

Capture CPU and memory data for telemetry processes

To capture CPU memory details use the following commands:

- `show processes cpu platform sorted | i pubd`
- `show processes memory platform sorted | s pubd`

Client event and SSID telemetry filters

This concept explains client event and SSID telemetry filters, which enable selective filtering of telemetry data for specific SSIDs to address scaling challenges in environments with many transient guest clients.

A client event and SSID telemetry filter is a feature that

- filters telemetry data for specific SSIDs to separate transient guest client data from clients requiring assurance,
- reduces scaling challenges by minimizing high-frequency telemetry data from transient clients, and
- enables configuration and management through controller and access point profiles.

Feature History for client event and SSID telemetry filters

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 104: Feature History

Release	Feature	Feature information
Cisco IOS XE Dublin 17.10.1	Cisco Catalyst Center Client Event and SSID Telemetry Filter	This feature filters out telemetry data for a configured SSID on the controller and AP.

Example: Filtering telemetry data for a guest SSID

For instance, in a large airport, the administrator configures the filter for the "Guest-WiFi" SSID. The controllers and APs then exclude telemetry data from transient guest clients, ensuring that only relevant data from corporate clients is processed and stored.

How the Client Event and SSID Telemetry Filter Works

This process explains how the client event and SSID telemetry filter operates in large enterprise environments. It outlines the key components, the filtering stages, and how the filter reduces unnecessary telemetry data processing and storage.

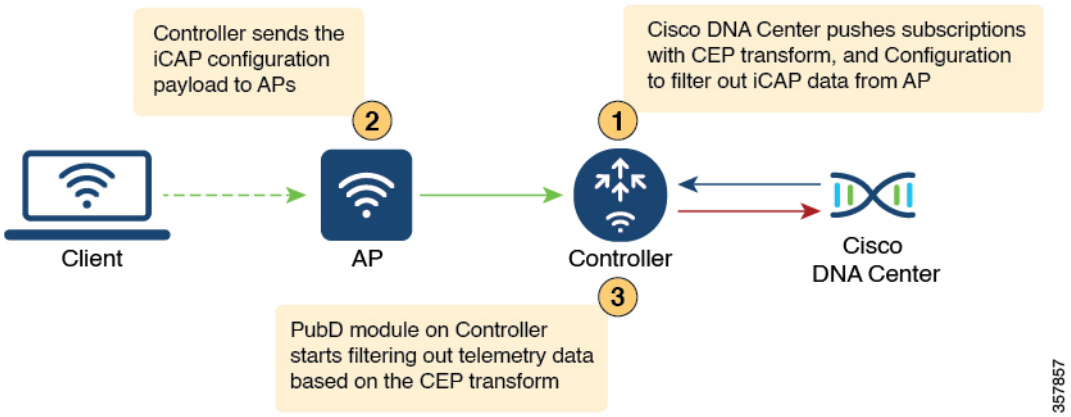
In environments such as airports and shopping malls, large numbers of transient guest clients can overwhelm telemetry systems. The filter allows administrators to configure which SSIDs' telemetry data is excluded from processing and storage.

- Telemetry data for configured SSIDs is filtered at both the controller and access point (AP) levels.
- The Complex Event Processing (CEP) transform is configured with the SSID to be filtered, and the Publishing Daemon (PubD) module in the controller enforces the filter.
- Filtered data types include client events, client statistics, AP or RF statistics, partial PCAP, and anomaly detection.

To debug filtering at PubD, run the following commands on the controller:

- `set platform software trace mdt-pubd chassis active r0 pubd debug`
- `set platform software trace mdt-pubd chassis active r0 mdt-xfrm debug`

Figure 11: High-Level End-to-End System Flow for Cisco Catalyst Center Client Event and SSID Telemetry Filter



Note

The Cisco Catalyst Center automation pushes the transforms. Enable or disable filtering for a specific SSID in the controller GUI.

Summary

This process enables administrators to selectively filter telemetry data for specified SSIDs in large-scale wireless environments. Configuring filters at both the controller and access point levels prevents unnecessary client and SSID telemetry data from being processed or stored. This improves system performance and scalability.

Workflow

1. Configure the CEP transform with the SSID to be filtered in Cisco Catalyst Center.
2. The controller applies the filter and pushes the configuration to the APs.
3. The AP programs the **aptrace** module to drop packets and events for the filtered SSID.

Result

Telemetry data for configured SSIDs is excluded from processing and storage at both the controller and AP levels. This helps maintain system performance and prevents overload, especially in large environments with many transient clients.

Restrictions for Cisco Catalyst Center client event and SSID telemetry filter

Use these guidelines when configuring Cisco Catalyst Center Client Event and SSID Telemetry Filter to ensure proper application and avoid unsupported configurations.

- CLI configuration is applicable for WLAN and not SSID. The Cisco Catalyst Center automation covers one-to-one mapping of WLAN to SSID.
- Controller does not send any notification to Cisco Catalyst Center at the beginning or at the end of filtering.
- Controller GUI configuration is not supported.

Configure Client Event and SSID Telemetry Filters

This process explains how to configure client event and SSID telemetry filters in Cisco DNA Center by creating WLANs, mapping them to policy profiles, and associating filters through the AP Join Profile.

Summary

Configuring client event and SSID telemetry filters in Cisco DNA Center consists of several stages. These stages enable targeted monitoring and reporting for wireless networks.

Workflow

Follow these stages to configure client event and SSID telemetry filters:

1. Create WLANs for client devices.
2. Map each WLAN to a policy profile to define network and security policies.
3. Create and assign a filter for the WLAN in the AP Join Profile to collect event and telemetry data.

Result

After completing these stages, client event and SSID telemetry filters are applied. You can then monitor and report on wireless network activity in Cisco DNA Center.

Enable iCAP filter on AP (CLI)

Learn how to enable iCAP filtering on APs using CLI commands.

Enable iCAP filtering in AP profiles to manage data telemetry and enhance security for WLAN deployments.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile
   ```

3. Add a description for the AP profile.

Example

```
Device(config-ap-profile)# description ap-profile-name
```

4. Enable iCAP filtering in APs.

Example

```
Device(config-ap-profile)# icap subscription client exclude telemetry-data
wlan wlan-profile-name
```

---

iCAP filtering is enabled for the specified AP profile; the designated telemetry data will be excluded according to your configuration.

Disable client telemetry data for a WLAN (YANG)

Disable Client Telemetry Data for a WLAN (YANG).

To disable the client telemetry data for a WLAN, use this RPC model:

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
message-id="urn:uuid:0a77124f-c563-469d-bd21-cc625a9691cc">
<nc:edit-config>
<nc:target>
<nc:running/>
</nc:target>
<nc:config>
<site-cfg-data xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-site-cfg">
<ap-cfg-profiles>
<ap-cfg-profile>
<profile-name nc:operation="merge">default-ap-profile</profile-name>
<icap-client-exclude-cfgs>
<icap-cient-exclude-cfg nc:operation="merge">
<wlan-profile nc:operation="merge">tel</wlan-profile>
</icap-client-exclude-cfg>
</cap-client-exclude-cfgs>
</ap-cfg-profile>
</ap-cfg-profiles>
</site-cfg-data>
</nc:config>
</nc:edit-config>
</nc:rpc>
```

For more information on YANG models, see the *Cisco IOS XE Programmability Configuration Guide* and YANG Data Models on Github at https://github.com/YangModels/yang/tree/master/vendor/cisco/xe.

You can contact the Developer Support Community for NETCONF/YANG features using this link:

https://developer.cisco.com/

Verify client telemetry data for WLAN

Verify client telemetry data for a WLAN.

To verify the client telemetry data for a WLAN, use this command:

```
Device# show running-config | section profile
ap profile default-ap-profile
 capwap retransmit count 8
 capwap timers primary-discovery-timeout 3000
 country IN
 description "default ap profile"
 icap subscription client exclude telemetry-data wlan guest
```

106 Application Performance Monitoring

Topics:

- APM
- Configure APM

APM

Introduces the core concepts of application performance monitoring and details specific restrictions that apply to its implementation in Cisco wireless deployments.

Application performance monitoring features are assurance solutions that

- collect and export assurance-related metrics for each application when flows are sent through specific AP interfaces
- aggregate these metrics using Cisco Catalyst 9800 Series controllers, and
- export the aggregated data to Cisco Catalyst Center for analysis.

Supported platforms

- Cisco Catalyst 9800 Series Controllers: 9800-80, 9800-40, 9800-L, and 9800-CL
- Cisco Catalyst 9100 Series APs in FlexConnect and fabric mode
- Cisco Catalyst 9300 Series and 9400 Series switches in fabric mode

> **Note**
>
> FNFv9 flows transit through the C9800 controller before being sent to Cisco Catalyst Center.

Supported monitors

- A general assurance monitor collects quantitative metrics for TCP and UDP flows, and qualitative metrics for TCP flows.
- A media monitor computes both qualitative and quantitative metrics for real-time protocol (RTP) flows.

Voice applications, such as Microsoft Teams and SIP, use RTP monitors. Other applications use TCP and UDP monitors.

A flow monitor can be attached to

- an interface that monitors all the flows from the attachment point.
- a wireless profile policy (the wireless profile policy that is associated with a WLAN or SSID) that monitors all the traffic passing through it.

Restrictions

Application Performance Monitoring restrictions.

- In FlexConnect and fabric mode, enable Application Performance Monitoring (APM) on a maximum of four WLANs. If APM is enabled on more than four WLANs, AP CAPWAP flaps might occur when APM is enabled or disabled, or when the AP mode is changed.
- Local flow exporter is not supported.
- These commands are not supported:
  - **show avc wlan application top**
  - **show avc client top application**

- You cannot configure Application Performance Monitoring and Application Visibility and Control basic on a single policy profile. You can configure each feature on separate policy profiles.
- During Control and Provisioning of Wireless Access Points (CAPWAP) restart, the AP moves to standby mode, and the nitro engine is disabled. When CAPWAP is up and the nitro engine is enabled, an attempt is made to classify the flows. Since there is not enough information to classify the applications, they are marked as unknown. When the AP rejoins CAPWAP, client traffic gets marked or classified correctly.
- When a client roams while an application has an active session, the specific session traffic is marked as unknown. The client has to start a new session to mark or classify the traffic correctly.

Configure APM

Guides the configuration of application performance monitoring by demonstrating how to create flow monitors, wireless WLAN policy profiles, policy tags, and attach policy profiles to APs, with steps for verifying the APM setup.

Create a flow monitor

Create a Flow Monitor using commands.

Configure flow monitors to collect wireless AVC assurance metrics.

Use these steps to set up flow monitors on a Cisco device for wireless assurance monitoring.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a flow monitor.

   Example

   ```
   Device(config)# flow monitor avc_assurance
   ```

3. Add a description to the flow monitor using the **description***description* command.

   Example

   ```
   Device(config-flow-monitor)# description assurance monitor ID is 90
   ```

4. Specify the IPv4 assurance metrics for wireless.

   Example

   ```
   Device(config-flow-monitor)# record wireless avc ipv4 assurance
   ```

5. Return to global configuration mode.

Example

```
Device(config-flow-monitor)# exit
```

6. Create a flow monitor.

Example

```
Device(config)# flow monitor avc_assurance_rtp
```

7. Add a description to the flow monitor using the **description***description* command.

Example

```
Device(config-flow-monitor)# description assurance-rtp monitor ID is 94
```

8. Specify the IPv4 assurance RTP metrics for wireless.

Example

```
Device(config-flow-monitor)# record wireless avc ipv4 assurance-rtp
```

9. Return to privileged EXEC mode.

Example

```
Device(config-flow-monitor)# end
```

The flow monitors for wireless AVC assurance and assurance RTP metrics are successfully created on the device.

Create a wireless WLAN profile policy

Create a wireless WLAN profile policy using commands.

Define and configure a WLAN policy profile using CLI commands for wireless network management.

Create a WLAN policy profile when you need to control traffic handling, enable advanced monitoring, or specify flow monitors for your wireless network on Cisco devices.

- Ingress flow monitors track incoming (received) traffic and help pinpoint what enters your wireless network.
- Egress flow monitors monitor outgoing (sent) traffic. This allows you to know what leaves your network.
- Multiple monitors (such as `avc_assurance` and `avc_assurance_rtp`) can be assigned to analyze different kinds of traffic or applications (for example, general application traffic and real-time media).

Before you begin

Know the profile name and flow monitor names you plan to use.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the WLAN policy profile and enter wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy policy-name
   ```

3. Disable the policy profile.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

4. Disable central switching.

   Example

   ```
   Device(config-wireless-policy)# no central switching
   ```

5. Specify the name of the IPv4 ingress flow monitor for general application traffic using the **ipv4 flow monitor** *monitor-name***input** command.

   Example

   ```
   Device(config-wireless-policy)# ipv4 flow monitor avc_assurance input
   ```

6. Specify the name of the IPv4 ingress flow monitor for real-time media traffic (RTP).

   Example

   ```
   Device(config-wireless-policy)# ipv4 flow monitor avc_assurance_rtp input
   ```

7. Specify the name of the IPv4 egress flow monitor for general application traffic.

   Example

   ```
   Device(config-wireless-policy)# ipv4 flow monitor avc_assurance output
   ```

8. Specify the name of the IPv4 egress flow monitor for real-time media traffic (RTP).

   Example

   ```
   Device(config-wireless-policy)# ipv4 flow monitor avc_assurance_rtp output
   ```

9. Enable the policy profile.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

10. Return to privileged EXEC mode.

Example

```
Device(config-wireless-policy)# end
```

The WLAN policy profile is created and configured with the specified settings. Traffic monitoring and switching behavior are now defined by the profile.

Create a policy tag

Create a Policy Tag using commands.

Define and attach a policy tag to a WLAN using the CLI.

Policy tags associate WLANs with specific policies on a Cisco wireless controller. When you create a policy tag and link it to a WLAN, you can apply policy profiles.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a policy tag and enter policy tag configuration mode.

Example

```
Device(config-policy-tag)# wireless tag policy policy-tag-name
```

3. Attach the policy tag to a WLAN.

Example

```
Device(config-policy-tag)# wlan wlan-avc  policy policy
```

4. Return to privileged EXEC mode.

Example

```
Device(config-policy-tag)# end
```

The policy tag is created and associated with the specified WLAN and policy.

Attach the policy profile to an AP

Attach the Policy Profile to an AP using commands.

Assign a policy profile to a specific AP to enforce wireless network policies.

Use this procedure when you need to apply or change the policy profile for an AP in your wireless network using CLI commands.

Procedure

1. Enter AP configuration mode.

   Example

   ```
   Device(config)# ap ap-ether-mac
   ```

   Example

   ```
   Device(config)# ap 9412.1212.1201
   ```

2. Specify the policy tag that is to be attached to the AP.

   Example

   ```
   Device(config-ap-tag)# policy-tag policy-tag
   ```

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-tag)# end
   ```

The AP is now configured with the specified policy profile.

Verify APM

Verify Application Performance Monitoring.

Use the following commands to verify application performance monitoring configuration.

To check application performance monitoring statistics, use these commands:

```
Device# show flow exporter statistics

Flow Exporter apm_exp:
  Packet send statistics (last cleared 4w1d ago):
    Successfully sent:         2082                  (216624 bytes)
!Packet sent count sent from controller to Cisco Cisco Catalyst Center
    Reason not given:          1099                  (114296 bytes)

  Client send statistics:
    Client: Flow Monitor avc
      Records added:           0
      Bytes added:             0
```

```
Device# show flow monitor assurance cache
```

```
Cache type:                                 Normal (Platform cache)
  Cache size:                                 200000
  Current entries:                                 0
  High Watermark:                                  1
!Controller flow monitor cache statistics

  Flows added:                                     6
  Flows aged:                                      6
    - Active timeout      (    10 secs)          6
```

To check status of application performance monitoring, use these command

```
Device# show avc status

VAP FNF-STATUS AVC-QOS-STATUS SD AVC-STATUS APM-STATUS
!APM-STATUS contains IPv4, IPv6 assurance and assurance-rtp monitors.

0   Disabled   Disabled       Enabled       IPV4,IPV4-RTP,IPV6,IPV6-RTP
1   Disabled   Disabled       Disabled      Disabled
2   Disabled   Disabled       Disabled      Disabled
3   Disabled   Disabled       Disabled      Disabled
4   Disabled   Disabled       Disabled      Disabled
5   Disabled   Disabled       Disabled      Disabled
6   Disabled   Disabled       Disabled      Disabled
7   Disabled   Disabled       Disabled      Disabled
8   Disabled   Disabled       Disabled      Disabled
9   Disabled   Disabled       Disabled      Disabled
10  Disabled   Disabled       Disabled      Disabled
11  Disabled   Disabled       Disabled      Disabled
12  Disabled   Disabled       Disabled      Disabled
13  Disabled   Disabled       Disabled      Disabled
14  Disabled   Disabled       Disabled      Disabled
15  Disabled   Disabled       Disabled      Disabled
```

107 Wireless Clients Threshold Warning

Topics:

- Wireless client threshold warnings

Describes how to configure periodic warnings and threshold percentages so the controller issues syslog, SNMP, and NETCONF/YANG notifications when connected wireless clients approach capacity, enabling proactive capacity management.

Wireless client threshold warnings

This concept explains how wireless client threshold warnings in Cisco IOS XE 17.6.x help administrators manage client limits by sending notifications when the number of wireless clients on a controller approaches a threshold that you define.

A wireless client threshold warning is a controller feature that

- lets you configure warning messages when the number of simultaneous wireless clients reaches a set threshold,
- sets the threshold by default to seventy-five percent of the controller's total capacity,
- notifies users through syslog messages, Simple Network Management Protocol (SNMP) traps, and NETCONF or YANG notifications when the threshold is breached.

Additional information

In Cisco IOS XE 17.6.x, this feature is enabled by default. For example, the Cisco Catalyst 9800-80 Wireless Controller supports up to 64,000 (sixty-four thousand) clients. The default threshold is 48,000 (forty-eight thousand) clients (seventy-five percent of capacity).

When the number of clients exceeds the threshold, the controller sends notifications using syslog messages, Simple Network Management Protocol (SNMP) traps, and NETCONF or YANG notifications.

This feature notifies you when you approach the wireless client limit, so you can act before reaching the maximum. You can also change the allowed number of wireless clients as needed.

> **Note**
>
> To disable the wireless client threshold warning feature, use the **no wireless max-warning** command.

Configure a warning period (CLI)

Learn how to configure a warning period for wireless client checks using CLI commands.

Set the time interval for generating wireless client warning notifications on devices using the CLI

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the periodicity of the wireless client check.

   Example

   ```
   Device(config)# wireless max-warning period interval-in-mins
   ```

   Valid values range from 1 to 60 minutes.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

The device generates wireless client warnings at the configured interval.

Configure the client threshold (CLI)

Learn how to configure the client threshold using CLI commands to set a warning threshold for the maximum number of wireless clients.

Set a threshold percentage to receive warnings when the number of connected wireless clients approaches the maximum limit.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the warning threshold percentage for the maximum number of wireless clients. Valid values range from 50 to 100 percent.

   Example

   ```
   Device(config)# wireless max-warning threshold clients threshold_percentage
   ```

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The device issues a warning when the number of connected wireless clients exceeds the specified threshold percentage of the maximum limit.

108 Intelligent Capture Hardening

Topics:

- Cisco Intelligent Capture (iCAP) feature and enhancements

This chapter explains configuring and verifying anomaly detection and RF statistics for Cisco APs, showing CLI commands for AP profiles and individual APs so administrators can detect, aggregate, and troubleshoot wireless client and AP issues.

Cisco Intelligent Capture (iCAP) feature and enhancements

Use Cisco Intelligent Capture Hardening to improve troubleshooting for wireless clients and APs by aggregating data and providing features such as anomaly detection and RF statistics.

The Cisco Intelligent Capture (iCAP) feature is a troubleshooting tool for wireless clients and APs that provides aggregated data and enhanced analysis capabilities.

- The feature aggregates data from wireless controllers and APs to make troubleshooting easier,
- the feature includes enhancements such as anomaly detection and RF statistics,
- the feature facilitates the identification of onboarding and transmission issues for wireless clients.

Feature history for Cisco Intelligent Capture hardening

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 105: Feature history

Release	Feature	Feature information
Cisco IOS XE Dublin 17.12.1	Cisco Intelligent Capture (iCAP) Hardening	The following enhancements are made to the iCAP feature: • Anomaly Detection • RF Statistics

Additional information

Use the Cisco Intelligent Capture (iCAP) feature in the Cisco DNA Center GUI to identify the cause of onboarding or AP transmission issues by gathering and displaying data from wireless controllers and APs.

For example, you can use the iCAP feature's anomaly detection and RF statistics to quickly identify and resolve wireless client onboarding issues.

Note

In Cisco IOS XE 17.15.2, iCAP virtual interface processor (VIP) packet capture is supported only for 802.11ax clients. iCAP VIP or full packet capture is not supported for 802.11be clients.

Anomaly detection

This topic explains the concept of Anomaly Detection in Cisco APs, describing its purpose, how it works, and its importance in identifying and preventing network issues.

An anomaly is a network event type that

- allows Cisco APs to detect possible irregularities in the behavior or lifecycle of wireless clients and APs

- informs the network administrator of issues, enabling identification of what happened and how to avoid recurrence, and
- supports aggregation and suppression of repeated anomaly notifications to prevent overwhelming Cisco DNA Center with duplicate events from the same client or event type.

How anomaly detection works

Anomaly detection is crucial in identifying network problems and understanding their root causes. Whenever Cisco APs detect an anomaly, they send individual anomaly events to Cisco Catalyst Center. To avoid flooding Cisco Catalyst Center with repeated events of the same type or from the same client within a short time frame, system enhancements automatically collapse and aggregate repeats into a single event.

Anomaly detection configuration enhancements are available on the controller, which now provisions and displays the iCAP (Intelligent Capture) status, providing better visibility into anomaly detection and troubleshooting.

Example of anomaly detection

If a wireless client repeatedly triggers the same anomaly, the AP aggregates these into a single event before sending it to Cisco Catalyst Center, allowing administrators to act on consolidated insights without being overwhelmed by redundant notifications.

RF statistics

This topic explains RF statistics, which provide information about the radio frequency (RF) environment and performance of connected access points (APs) in a wireless network.

A set of RF statistics is a data collection feature that

- gathers real-time and historical information about the radio frequency environment,
- measures the performance and health indicators of wireless network access points, and
- supports monitoring and troubleshooting by providing details such as signal quality, noise levels, and channel usage.

Additional information

In Cisco IOS XE 17.12.1 and later, per-AP RF statistics are delivered directly from the wireless controller through iCAP subscription. In earlier versions (up to Cisco IOS XE 17.11.1), only basic statistical information was available.

Configure anomaly detection in AP profile (CLI)

Learn how to configure anomaly detection in an AP profile using CLI commands to enable and customize client anomaly-detection reporting.

Enable individual anomaly detection and detailed reporting for clients in an AP profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# onfigure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

Example

```
Device(config)# ap profile ap-profile
```

3. Enable individual reports for client anomaly-detection subscription.

   Example

   ```
   Device(config-ap-profile)# icap subscription client anomaly-detection
   report-individual enable
   ```

4. Enable individual reports aggregation for client anomaly-detection subscription.

   Example

   ```
   Device(config-ap-profile)# icap subscription client anomaly-detection
   report-individual enable aggregate
   ```

   This command is disabled by default.

5. Configure event reports per client, every five minutes.

   Example

   ```
   Device(config-ap-profile)# icap subscription client anomaly-detection
   report-individual per-client throttle number-of-event-reports
   ```

   The value of an event report ranges from 0 to 50 reports. The default value is five reports.

6. Configure event reports per type, every five minutes.

   Example

   ```
   Device(config-ap-profile)# icap subscription client anomaly-detection
   report-individual per-type throttle number-of-event-reports
   ```

   The value of an event report ranges from 0 to 100 reports. The default value is five reports.

Anomaly detection is activated for the AP profile, and the system generates individual and aggregated anomaly reports.

Configure anomaly detection in an AP (CLI)

Learn how to configure anomaly detection in an Access Point using CLI commands.

Enable and configure anomaly detection features on a specific access point using CLI commands.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

2. Enable individual reports for client anomaly-detection subscription for a single AP.

   Example

   ```
   Device# ap name ap-name icap subscription client anomaly-detection
   report-individual enable
   ```

3. Enable individual reports aggregation for client anomaly-detection subscription, for a single AP.

   Example

   ```
   Device# ap name ap-name icap subscription client anomaly-detection
   report-individual enable
   ```

4. Configure event reports per client, every five minutes, for a single AP.

   Example

   ```
   Device# ap name ap-name icap subscription client anomaly-detection
   report-individual per-client throttle number-of-event-reports
   ```

   The value of an event report ranges from 0 to 50 reports.

5. Configure event reports per type, every five minutes, for a single AP.

   Example

   ```
   Device# ap name ap-name icap subscription client anomaly-detection
   report-individual per-type throttle number-of-event-reports
   ```

   The value of an event report ranges from 0 to 100 reports.

Anomaly detection is enabled on the targeted AP, and report generation is configured according to your specified parameters.

Verify anomaly detection and RF statistics

Verify Anomaly Detection and RF Statistics.

To verify the current status of the anomaly-detection subscription of an AP, use this command:

```
Device# show ap name cisco-AP icap subscription client anomaly-detection chassis
 active R0
Per-AP ICap configuration

Anomaly detection subscription
  State                     : enabled
  Client filter             : 006b.f107.a520
  Client filter             : 006b.f107.a521
  DHCP timeout (seconds)    : 5
  Trigger AP packet trace   : enabled
  Report Individual         : enabled
  Report Individual aggregate : enabled
  Report Individual throttled events (per 5 minute) : 5
  Report Individual per type throttled events (per 5 minute) : 14
  Report Individual per client throttled events (per 5 minute) : 15
```

```
  Report Summary           : disabled
  Report Summary frequency (minutes) : 5
```

To verify RF statistics, use this command:

> **Note**
>
> The controller **show** command is enhanced to display data from the **txTotalDrops** counter.

```
Device# show wireless client mac-address 00XX.ecXX.7aXX detail
.
.
.
Client Statistics:
  Number of Bytes Received from Client : 62861
  Number of Bytes Sent to Client : 6754
  Number of Packets Received from Client : 455
  Number of Packets Sent to Client : 65
  Number of Data Retries : 0
  Number of RTS Retries : 0
  Number of Tx Total Dropped Packets: x
  Number of Duplicate Received Packets : 0
  Number of Decrypt Failed Packets : 0
  Number of Mic Failured Packets : 0
  Number of Mic Missing Packets : 0
  Number of Policy Errors : 0
  Radio Signal Strength Indicator : -21 dBm
  Signal to Noise Ratio : 73 dB
.
.
.
```

109 Amazon S3 Support

Topics:

- Amazon S3 support

This chapter explains how to configure and verify Amazon S3 cloud storage on Cisco Catalyst devices, including creating S3 profiles, setting credentials, permissions, region and mount points, and commands to enable and confirm operational status.

Amazon S3 support

Amazon Simple Storage Service (Amazon S3) support lets you use scalable, cloud-based storage from Amazon Web Services (AWS) with your Cisco Catalyst device. You can supplement onboard storage for software images, logs, and maintenance updates.

Cisco Catalyst devices support Amazon Simple Storage Service (Amazon S3), a cloud storage solution from Amazon Web Services (AWS). With this integration, you can expand built-in persistent storage to meet increased demands for file storage, logging, and software management.

- You get flexible, scalable storage for large software images and higher logging rates.
- You can distribute software images, service maintenance updates (SMUs), and scripts across multiple devices.
- You can enhance device capabilities by adding cloud storage to existing onboard resources.

Amazon S3 support overview and details

As software images grow and logging requirements increase, you may need additional storage capacity on your Cisco Catalyst device.

Built-in persistent storage may not be sufficient. Starting with IOS-XE release 17.13.1, Cisco Catalyst 9800 Series Wireless Controllers support Amazon Simple Storage Service (Amazon S3).

Amazon S3 is a scalable cloud storage infrastructure that you can access through a web interface.

With Amazon S3 support, you can supplement onboard storage to meet new operational and maintenance demands.

Restrictions and guidelines

This section describes the restrictions and guidelines for configuring and accessing cloud storage.

These restrictions and guidelines apply to cloud storage configuration:

- Only the active device can access the cloud storage.
- You can establish cloud reachability through any service port, including device management ports or forwarding interfaces on the device.
- You can create multiple cloud storage configuration profiles for the same Amazon Simple Storage Service (S3) bucket using different configuration parameters.
- If you host virtual device instances (such as C9800-CL) on AWS, you can use the Identity and Access Management (IAM) role infrastructure to access Amazon Simple Storage Service (S3) storage.

Configure Amazon S3 support (CLI)

Learn how to configure Amazon S3 support by using CLI commands to enable cloud storage integration on your device.

Configure Amazon S3 support so your device can use Amazon S3 cloud storage for remote data management and backup.

Before you begin

- Ensure that connectivity to the cloud is established.
- Ensure that you have the AWS Identity and Access Management (IAM) access key ID and secret key ID.
- Ensure that DNS is configured. If DNS is not configured, S3 bucket resolution fails.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an Amazon S3 cloud services profile.

   Example

   ```
   Device(config)# cloud-services aws s3 profile profile-name
   ```

   Use only alphanumeric characters for the profile name. The underscore (_) is the only supported special character.

3. Configure the Amazon S3 storage bucket and its mount point.

   Example

   ```
   Device(config-s3fs-profile)# bucket bucket-name mount-point s3-mount
   ```

   A bucket is a container for objects stored in Amazon S3.

   The mount point is the directory on your local file system where you mount the Amazon S3 bucket.

   > **Note**
   >
   > Ensure that DNS is configured.

4. Optional: Add a description to the Amazon S3 cloud services profile.

   Example

   ```
   Device(config-s3fs-profile)# description profile-description
   ```

   You can enter a description of up to 255 alphanumeric characters.

5. Optional: Configure the management interface to serve as the VRF interface.

   Example

   ```
   Device(config-s3fs-profile)# vrf mgmt-Intf
   ```

   By default, the forwarding interface is used.

6. Configure the AWS S3 access credentials.

   Example

   ```
   Device(config-s3fs-profile)# access-key key-id iam-id secret-key { 0 | 8 }
   secret-key
   ```

Use the access key ID and secret key that were created for the IAM role in the AWS Management Console.

7. Set the Amazon S3 bucket to read and write permissions.

   Example

   ```
   Device(config-s3fs-profile)# permissions read-write
   ```

   By default, read-only permission is enabled.

8. Specify the Amazon S3 region where the cloud-based storage is used.

   Example

   ```
   Device(config-s3fs-profile)# region region
   ```

9. Configure the HTTP server URL, the HTTPS server URL, or the IPv4 address, along with the port details.

   Example

   ```
   Device(config-s3fs-profile)# proxy https-server url-ip port port-num
   ```

10. Save the configuration and enable it for Amazon S3 services.

    Example

    ```
    Device(config-s3fs-profile)# no shutdown
    ```

11. Return to global configuration mode.

    Example

    ```
    Device(config-s3fs-profile)# exit
    ```

After completing these steps, you can securely access Amazon S3 storage buckets and manage cloud storage operations through the CLI.

Verify Amazon S3 support

Verifying Amazon S3 Support.

To view a summary of all the Amazon S3 profiles, use this command.

```
Device# show cloud-services aws s3 summary

Profile Name                        Profile Status  Service Status
-------------------------------------------------------------------
test                                Started         Active

test2                               Started         Active
```

To view the operational information of a Amazon S3 profile, use this command.

```
Device# show cloud-services aws s3 profile s1

Profile Details

Profile Name          : s1
Bucket Name           : pb-s3-test1
Mount Point           : test
Bucket Permission     : Read-Only
Region                : us-west-1
VRF                   : Global

S3 Service Details

Service Status        : Active
Service PID           : 31934
Mount Time            : 09/28/23 17:06:25
Service Log Level     : Notice
```

110 Amazon Web Services CloudWatch

Topics:

- Amazon Web Services CloudWatch Support

This chapter explains how to configure and verify integration with AWS monitoring services on supported Cisco controllers, detailing prerequisites, CLI profile and credential setup, log group/stream selection, and secure syslog streaming.

Amazon Web Services CloudWatch Support

This concept explains the monitoring and observability features provided by Amazon Web Services (AWS) CloudWatch, including integration with Amazon Elastic Compute Cloud (EC2) instances and on-premises controllers to collect, transmit, and analyze system logs, metrics, and events.

A CloudWatch feature is a monitoring capability that

- collects and aggregates system logs, metrics, and events from Amazon Elastic Compute Cloud (EC2) instances and on-premises controllers,
- transmits these data points to the AWS CloudWatch server for centralized monitoring, and
- provides insights into the performance, health, and operation of applications and resources running on AWS infrastructure.

Additional Information

AWS CloudWatch is a service in the Amazon Web Services (AWS) cloud platform. It monitors server system logs, metrics, and events.

Organizations can integrate CloudWatch with Amazon Elastic Compute Cloud (EC2) instances and on-premises servers. This integration enables efficient transmission of logs, events, and metrics to the AWS CloudWatch server.

The CloudWatch service enables detailed insights into applications, resources, and services running on AWS infrastructure. These insights help ensure performance, support troubleshooting, and maintain the overall health of controllers and systems.

The CloudWatch service is disabled by default and must be enabled to function.

When the AWS CloudWatch agent is active on a controller, the agent gathers system logs from the controller and transmits them securely to the AWS CloudWatch server.

Example: Using AWS CloudWatch for Monitoring

The AWS CloudWatch agent can be used to:

- collect internal system-level metrics from Amazon Elastic Compute Cloud (EC2) instances across multiple operating systems, and
- collect system-level metrics from on-premises devices or controllers.

Benefits of Amazon web services CloudWatch service

This topic explains Amazon Web Services (AWS) CloudWatch features, including unified monitoring, enhanced operational efficiency, and actionable insights through analytics and visualizations.

The AWS CloudWatch feature provides unified monitoring, operational efficiency, and actionable insights for cloud environments.

- **A unified monitoring and observability platform** : All device logs are consolidated in a single location, facilitating easy event monitoring and seamless action using the cloud services tools.
- **Enhanced operational efficiency and resource optimization** : Automate the processes and establish alarms for specific events or logs, thereby improving operational performance and resource management.
- **Gain valuable insights from logs** : Analyze and visualize the logs, allowing you to take appropriate actions based on the events and logs.

The AWS CloudWatch feature is supported on the following controllers: Cisco Catalyst 9800-40, 9800-80, 9800-L, and 9800-CL (private [VMware ESXI, KVM, Hyper-V] and public cloud [AWS C9800-CL instances only] platforms).

Configure an AWS CloudWatch Profile (CLI)

Configure an Amazon Web Services (AWS) CloudWatch profile using CLI commands to stream syslog messages and enable centralized monitoring.

Configure an AWS CloudWatch profile using CLI commands. This configuration enables secure streaming of syslog messages and log data to AWS CloudWatch for centralized logging and monitoring.

The AWS CloudWatch agent transmits buffered syslog messages to the AWS CloudWatch service.

The agent scans and retrieves logs from files in a designated directory. You can specify a single file or use a wildcard pattern to include multiple files.

You can specify the storage location in the AWS CloudWatch agent profile. When files are updated, the agent reads their content.

Before you begin

- Create CloudWatch groups and streams in AWS.
- Create access credentials in AWS.
- Set a Private Configuration Key for Password Encryption
- Ensure that you have the AWS Identity and Access Management (IAM) access key ID and secret key.
- You can run the optional **logging buffered** and **logging persistent** commands to send syslogs to AWS CloudWatch.
- Ensure that DNS is configured.

To create CloudWatch groups and streams, see the AWS documentation: https://docs.aws.amazon.com/IAM/latest/UserGuide/introduction.html

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AWS CloudWatch profile.

   Example

   ```
   Device(config)# cloud-services aws cloudwatch profile profile-name
   ```

3. Optional: Add a description to the AWS CloudWatch profile.

   Example

   ```
   Device(config-cloudwatch-profile)# description profile-description
   ```

4. Optional: Configure the HTTP or HTTPS server URL or IP address and port.

Example

```
Device(config-cloudwatch-profile)# proxy https-server url-ip port port-num
```

5. Optional: Configure the management interface as the VRF interface.

   Example

   ```
   Device(config-cloudwatch-profile)# vrf mgmt-Intf
   ```

   Use this option to send agent traffic through the management interface. By default, the data port interface is used.

   Use this command only if the management interface is available. This interface is not available in C9800-CL public cloud instances.

6. Configure the AWS CloudWatch access credentials.

   Example

   ```
   Device(config-cloudwatch-profile)# access-key key-id iam-key-id secret-key 0
    secret-key
   ```

   Use the access key ID and secret key created for the IAM user in the AWS console.

7. Specify the AWS region where the CloudWatch server is running.

   Example

   ```
   Device(config-cloudwatch-profile)# region us-west-1
   ```

8. Specify the AWS CloudWatch log group name, log stream name, and optionally the log file path.

   Example

   ```
   Device(config-cloudwatch-profile)# log group-name group-name stream-name
   stream-name file-path file-path
   ```

   If you do not provide a log file path, the system uses the default syslog path: /bootflash/syslog/

   The log group and log stream must match those created in AWS CloudWatch.

   If specified, the log file path does not need to match the buffered logging persistent storage directory or file name.

9. Save and enable the configuration for AWS CloudWatch services.

   Example

   ```
   Device(config-cloudwatch-profile)# no shutdown
   ```

10. Return to global configuration mode.

    Example

    ```
    Device(config-cloudwatch-profile)# exit
    ```

The AWS CloudWatch profile is configured and enabled. The device can now transmit syslog messages and log data to the specified AWS CloudWatch log group and stream for monitoring and integration with AWS services.

Verify AWS CloudWatch configuration

Verify AWS CloudWatch Configuration.

To view summary of AWS CloudWatch profiles, run this command:

```
Device# show cloud-services aws cloudwatch summary

Profile Name                      Profile Status  Service Status
-------------------------------------------------------------------
demo3                             Started         Active

demo4                             Started         Active
```

To view details of a specific AWS CloudWatch profile, run this command:

```
Device# show cloud-services aws cloudwatch profile demo3

Profile Details

Profile Name          : demo3
VRF                   : Global
Region                : ap-northeast-1

CloudWatch Service Details

Service Status        : Active
Service PID           : 31785
Service Log Level     : Notice

Log Details

Log Group Name                          Log Stream Name
       Log File
_____________________________________________________________________________
test                                    katar2
```

111 Kernel Minidump and Trustzone Upgrade

Topics:

- Kernel minidump and trust zone upgrade

This chapter describes how to enable and configure kernel minidump collection and automatic TrustZone upgrades on supported Cisco APs, guiding administrators through setup, limits, verification, and retrieving crash TLV files for analysis.

Kernel minidump and trust zone upgrade

This topic explains the Kernel Minidump and Trustzone Upgrade feature, which provides methods for diagnosing kernel crashes on Wi-Fi 6 (802.11AX) access points (APs) by collecting and exporting critical crash data.

The kernel minidump and trust zone upgrade feature is a diagnostic capability that

- collects and preserves kernel and driver data structures when a crash occurs
- stores these data structures for post-crash analysis in a standardized Type-Length-Value (TLV) format
- automatically upgrades the trust zone on supported devices so the system can collect CPU registers for comprehensive debugging.

Feature History for kernel minidump and trustzone upgrade support

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 106: Feature history

Release	Feature information
Cisco IOS XE 17.18.1	This feature enhances control over minidump collection on Wi-Fi 7 (802.11be) APs. A new option has been added to limit the number of kernel core dump directories stored on the AP. This command has been modified: • **core-dump kernel dir-limit** Support has been added for these access points: • Cisco Wireless 9178 Series Access Points. • Cisco Wireless 9176 Series Access Points.
Cisco IOS XE 17.14.1	Kernel Minidump and Trustzone Upgrade feature offers a more effective method for diagnosing kernel issues. This command is introduced: • **core-dump kernel type**

Feature overview

Traditionally, when a kernel crash occurs on Wi-Fi 6 (802.11AX) access points, only AP console logs are available for troubleshooting. However, these logs often lack the detail needed to determine the root cause accurately.

The kernel minidump and trust zone upgrade feature, introduced in Cisco IOS XE 17.14.1, provides a more effective method for diagnosing kernel crashes. The feature captures and organizes internal data structures critical for debugging so you can troubleshoot problems more accurately.

How kernel minidumps work

When you enable the feature, the system collects references to key kernel or driver structures. It stores them in a Type-Length-Value (TLV) format in kernel memory.

After a crash and the access point reboots, the system writes the referenced data structures to the */storage/cores* directory on the access point flash memory. You can export these files for advanced analysis.

Trust zone upgrade

In AP using Qualcomm Software Development Kit (QSDK) version 11.3, the trust zone module collects CPU registers after a crash. It then preserves these registers for later retrieval by the crash dump process.

When you enable the kernel minidump and trust zone upgrade feature, the system automatically upgrades the trust zone to the latest version on supported access points.

> **Note**
>
> The trust zone upgrade occurs only once. If you disable the Kernel Minidump and Trustzone Upgrade feature or downgrade the controller software to a release earlier than Cisco IOS XE 17.14.1, the trust zone upgrade does not roll back.

Supported APs

You can use the kernel minidump and trust zone upgrade feature on the following access points running Cisco IOS XE 17.14.1 or later:

- Cisco Catalyst 9124 Series Access Points
- Cisco Catalyst 9136 Series Access Points

Example: Kernel minidump and trust zone upgrade in action

After you enable the feature, if a supported access point experiences a kernel crash, you can access the TLV files in the */storage/cores* directory and export them. This provides comprehensive diagnostic data to accelerate root cause analysis and issue resolution.

Configure the Minidump from the AP (CLI)

Learn how to configure minidump collection on an access point (AP) using the command-line interface (CLI). Use this procedure to enable kernel coredump collection directly from the AP.

Configure minidump collection on your access point. This lets you capture kernel coredump information for troubleshooting and diagnostics.

Before you begin

- Ensure that no clients are connected to the AP.
- Keep your AP in standalone mode. This prevents it from receiving conflicting payloads from the controller.
- Any configuration pushed from the controller overrides the value you set on your AP.

Procedure

Enable kernel coredump collection on the AP.

Example

```
Device# configure boot minidump enable
```

After you enable kernel coredump collection, the AP stores critical kernel dump data for further analysis.

Configure the minidump from the controller (CLI)

Learn how to configure the Minidump feature from the controller using the CLI, including setting limits and types for kernel core dumps on an AP profile.

Enable and customize collection of kernel core dumps (minidump) for access points to support troubleshooting and diagnostics.

Use this procedure when you need to collect kernel core dumps from access points to assist with root cause analysis of device crashes or unexpected reboots. Configuring the minidump feature allows you to control the number and type of core dumps stored for each AP.

Procedure

1. Enters global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

3. Configure the maximum number of kernel core dumps to be collected on an AP.

   Example

   ```
   Device(config-ap-profile)# core-dump kernel limit limit
   ```

   Valid range is from 0 to 5.

4. Configure the maximum number of directories for AP minidump collection.

   Example

   ```
   Device(config-ap-profile)# core-dump kernel dir-limit dir-limit
   ```

   The valid range is from 15 to 25.

5. Configure the type of kernel core dump to be collected on the AP.

   Example

   ```
   Device(config-ap-profile)# core-dump kernel type mini-dump
   ```

Use the **core-dump kernel type disable** command to disable kernel core dump.

> **Note**
>
> Changing the core dump type from disabled to full-dump or mini-dump or vice versa will cause the APs to reboot.
>
> When you enable the full-dump option, the kernel minidump feature gets deactivated.

6. Return to global configuration mode.

Example

```
Device(config-ap-profile)# end
```

The AP profile is configured to collect kernel minidumps according to the specified limits and type. AP using this profile will store core dumps as configured.

Verify minidump configuration

To view the mini-dump configuration on the controller, use this command:

```
Device# show ap name AP3C57.31C5.99D0 config general  | sec Kernel core dump

Kernel core dump :
  Configured limit                              : 3
  Kernel core dumps collected on AP             : 1
  Kernel core dump type                         : Mini dump
```

To view the mini-dump configuration on the AP, use this command:

```
AP# show boot

--- Boot Variable Table ---
BOOT path-list:      part1
Console Baudrate:    115200
Enable Break:        yes
Manual Boot:         yes
Memory Debug:        no
Crashkernel:         no
Minidump:            yes # Indicates Minidump is enabled.
SCRUB_LIMIT:         40 (default)
Kdump Limit:         5 # Configured limit. (this came from WLC via payload)
Kdump Collected:     0 # Number of times the feature ran after it was enabled.
Debug init:          0
```

Part IX

Cloud Services

Topics:

- Cisco Spaces
- Sensor Connect backup and restore
- AP and Sensor Connect support for full filtering
- Using Cloud Monitoring as a Solution for Network Monitoring
- Wireless Active Testing with ThousandEyes
- Wireless Active Testing SCEP support

112 Cisco Spaces

Topics:

- Cisco Spaces

Cisco Spaces

Introduces Cisco CMX Cloud, covering location analytics features, deployment scenarios, and configuration steps for wireless networks, including integration best practices and supported APIs.

A Cisco Connected Mobile Experience (CMX) deployment is a cloud-based indoor location services platform that

- provides real-time location analytics and engagement capabilities for indoor environments
- enables integration with Cisco wireless controllers through secure protocols such as HTTPS, and
- supports scalable, cloud-delivered access to location-based services.

The Network Mobility Services Protocol (NMSP) cloud-service of the wireless controller communicates with Cisco Spaces using HTTPS as a transport protocol.

Configure Cisco Spaces

Configure Cisco Spaces using commands.

Ensure successful operation of NMSP cloud-services by configuring DNS, importing root CAs, and registering to Cisco Spaces.

To configure, Cisco Spaces, perform these procedures. These steps enable the controller to communicate securely with cloud services and Cisco Spaces.

Before you begin

- *Configure DNS*: To resolve fully qualified domain names used by NMSP cloud-services, configure a *DNS* server using the **ip name-server** *server_address* configuration command as shown in Step 2.
- *Import 3rd party root CAs*: The controller verifies the peer and host based on the certificate sent by the CMX when a connection is established. Root CA certificates are not preinstalled on the controller. Import a set of root CAs trusted by Cisco to the trustpool of the crypto PKI using the **crypto pki trustpool import url** <**url**> configuration command as shown in Step 3.
- Successful registration to Cisco Spaces is required before configuring the **server url** and **server token** parameters to complete setup.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the DNS on the controller to resolve the FQDN names used by the NMSP cloud-services.

   Example

   ```
   Device(config)# ip name-server 10.10.10.205
   ```

3. Import the 3rd party root CA.

Example

```
Device(config)# crypto pki trustpool import url
http://www.cisco.com/security/pki/trs/ios.p7b
```

The controller verifies the peer using the imported certificate.

4. Configure the URL used for cloud services.

Example

```
Device(config)# nmsp cloud-services server url https://cisco.com
```

Use the *no* form of the command to delete the server url from the configuration.

5. Configure the authentication token for the NMSP cloud service.

Example

```
Device(config)# nmsp cloud-services server token test
```

Use the *no* form of the command to delete the server token from the configuration.

6. Optional: Configure HTTP proxy details for the NMSP cloud service.

Example

```
Device(config)# nmsp cloud-services http-proxy 10.0.0.1 port 10
```

Use the *no* form of the command to disable the use of a HTTP proxy.

7. Enable NMSP cloud services.

Example

```
Device(config)# nmsp cloud-services enable
```

Use the *no* form of the command to disable the feature.

Verify Cisco Spaces configuration

Verify Cisco Spaces configuration.

Use the commands to verify the Cisco Spaces configuration.

To view the status of active NMSP connections, use the command:

```
Device# show nmsp status

MSE IP Address   Tx Echo Resp  Rx Echo Req   Tx Data     Rx Data   Transport

------------------------------------------------------------------------------
192.0.2.1        0             0             1           1         TLS
198.51.100.1    0             0            1230        2391      HTTPs
```

To view the NMSP cloud service status, use the command:

```
Device# show nmsp cloud-services summary

CMX Cloud-Services Status
-------------------------
Server:                                 https://yenth8.cmxcisco.com
IP Address:                             198.51.100.1
Cmx Service:                            Enabled
Connectivity:                           https: UP
Service Status:                         Active
Last Request Status:                    HTTP/1.1 200 OK
Heartbeat Status:                       OK
```

To view the NMSP cloud service statistics, use the command:

```
Device# show nmsp cloud-services statistics

CMX Cloud-Services Statistics
-----------------------------
Tx DataFrames:                           3213
Rx DataFrames:                           1606
Tx HeartBeat Req:                       31785
Heartbeat Timeout:                          0
Rx Subscr Req:                           2868
Tx DataBytes:                           10069
Rx DataBytes:                           37752
Tx HeartBeat Fail:                          2
Tx Data Fail:                               0
Tx Conn Fail:                               0
```

To view the mobility services summary, use the command:

```
Device# show nmsp subscription summary

Mobility Services Subscribed:
Index Server IP Services
----- --------- --------
1 209.165.200.224 RSSI, Info, Statistics, AP Monitor, AP Info
2 209.165.200.224 RSSI, Statistics, AP Info
```

113 Sensor Connect backup and restore

Topics:

- Sensor Connect backup and restore

This chapter explains Sensor Connect backup and restore.

Sensor Connect backup and restore

Explains how Sensor Connect backup and restore mechanisms safeguard application and configuration data that controller configuration backups alone do not cover.

A Sensor Connect backup and restore is an application recovery mechanism that

- creates a separate backup of the Sensor Connect application database starting from the Sensor Connect 1.2.2 release
- restores the Sensor Connect configuration and operational data that the controller configuration backup does not include, and
- returns a freshly deployed Sensor Connect application to its saved Day-N state.

The Cisco Catalyst 9800 Wireless Controller configuration backup preserves controller settings, such as VLAN and subnet configurations. It does not preserve the Sensor Connect application database, which contains information such as AP and endpoint configurations, application credentials, and controller registration data. To recover a deployment completely, create separate backups of the controller configuration and the Sensor Connect application.

You can use Sensor Connect backup and restore when you replace a controller or redeploy the Sensor Connect application.

During a restore operation, the controller

- stops the running Sensor Connect application
- restores the application database from the backup file
- restarts the application, and
- returns the application to the **Running** state.

Feature history for Sensor Connect backup and restore

This section explains the feature history details of the Sensor Connect backup and restore feature.

- This table provides release and related information for the feature explained in this module.
- This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 107: Feature history for Sensor Connect backup and restore

Feature Name	Release Information	Feature Description
Sensor Connect backup and restore	Cisco IOS XE 26.2.1	Sensor Connect backup and restore is an application recovery mechanism that separately backs up the Sensor Connect application database and restores configuration and operational data not included in the controller configuration backup. It enables a newly deployed Sensor Connect application to return to its saved Day-N state after controller replacement or application redeployment.

Guidelines for Sensor Connect backup and restore

Outlines the essential best practices to follow when backing up or restoring Sensor Connect.

Follow these guidelines when you back up or restore Sensor Connect:

- Create a Sensor Connect application backup in addition to the controller configuration backup.
- For controller replacement, restore the controller configuration and deploy Sensor Connect before you restore the Sensor Connect application backup.
- Ensure that the Sensor Connect application is in the **Running** state before you start a restore operation.
- Allow sufficient time for a backup or restore operation to complete. The operation duration depends on the deployment scale and the resulting backup-file size.

Restrictions for Sensor Connect backup and restore

Outlines the essential restrictions to follow when backing up or restoring Sensor Connect.

Sensor Connect backup and restore has these restrictions:

- Restore is supported only when the destination controller runs the same Sensor Connect application version that was running when you created the backup.
- Cross-platform restore is not supported. Restore the backup only on the same controller platform type on which you created it because controller platforms support different application scales.

Backup the Sensor Connect application

Configure and save a backup artifact of the Sensor Connect application database using HTTP or SFTP methods.

Create a backup artifact of the Sensor Connect application database by downloading the file to your local system through HTTP or storing the file on an SFTP server.

Perform this task from the controller. A successful backup does not stop the Sensor Connect application; the application remains in the **Running** state.

Procedure

1. Log in to the controller and choose **Configuration** > **Services** > **IoT Services**.
2. From the More Actions menu, choose **Backup**.
   The **Backup** panel opens.
3. From the **Transfer Mode** drop-down list, choose a transfer method:
   - **HTTP**: Choose this method to download the generated `.tar.gz` backup file to your local system.
   - **SFTP**: Choose this method and enter the **Server IP Address**, **File Path**, **File Name**, **Server Login Username**, and **Server Login Password** to store the backup on an SFTP server.
4. Click **Backup**.
   The controller displays the backup progress. For **HTTP**, the controller downloads the generated backup file to your local system. For **SFTP**, the controller writes the backup file to the specified server path.

   Ensure `tar.gz` is sucessfully downloaded in your local system or in the specified **SFTP** server path.
5. Confirm that the controller reports a successful backup and displays the last-completed backup timestamp.

The Sensor Connect backup artifact is available at the selected destination, and the application remains in the **Running** state.

Restore the Sensor Connect application

Restore the Sensor Connect application database from a backup file using the controller.

Recover the Sensor Connect application to a previous state by restoring its database from a backup file.

Perform this task after controller configuration is restored and Sensor Connect is redeployed. The restore workflow stops the running application, applies the backup database, and restarts the application.

Before you begin

- Ensure the destination controller is the same platform type as the one used for the backup.
- Deploy the same Sensor Connect application version that was running when you created the backup.
- Verify the Sensor Connect deployment is successful, with the application status as **Running**.
- Make the Sensor Connect `.tar.gz` backup file available on your local system or an SFTP server.

Procedure

1. Log in to the controller and choose **Configuration** > **Services** > **IoT Services**.
2. From the **More Actions** menu, choose **Restore**.
   The Restore panel opens.
3. From the **Transfer Mode** drop-down list, choose a transfer method:
   - **HTTP**: Choose this method and select the Sensor Connect backup file from your local system.
   - **SFTP**: Choose this method and enter the **Server IP Address**, **File Path**, **File Name**, **Server Login Username**, and **Server Login Password**.
4. Click **Restore**.
5. When the controller requests consent to stop the application, confirm that you want to continue.
   The controller stops the application and starts the restore operation.
6. Monitor the operation until the controller reports that the restore is successful and the application returns to the **Running** state.
7. Confirm that the controller displays the last-completed restore timestamp.

The controller restores the Sensor Connect application database and restarts the application in its saved Day-N state.

What to do next

Launch Sensor Connect and verify the restored application state. For more information, refer the Sensor Connect Online Help guide.

Sensor Connect launch and Day-0 configuration

Provides reference information on launching Sensor Connect, Day-0 workflows, restore behaviors, credential use, and help resources.

Sensor Connect offers several key functions during initial setup and launch:

- You must launch Sensor Connect from the controller **IoT Services** page after the application reaches the **Running** state.

- After a restore operation, Sensor Connect opens in its saved Day-N state. Use the old credentials from the restored database to login to the application and verify that the expected AP inventory is present and controller registration is active.
- For fresh deployments without a restored database, Sensor Connect opens the Day-0 workflow. Use your credentials to sign in for the first time, perform credential changes, onboarding, and other application procedures.
- For additional details on sign-in, credential management, onboarding, or other Sensor Connect operations, refer the Sensor Connect Online Help guide.

114 AP and Sensor Connect support for full filtering

Topics:

- AP and Sensor Connect support for full filtering

This topic describes the AP and Sensor Connect support for full filtering details.

AP and Sensor Connect support for full filtering

Explains how Cisco Catalyst access points filter Bluetooth Low Energy (BLE) advertisements before forwarding them to Sensor Connect and Cisco Spaces.

AP and Sensor Connect support for full filtering enables Cisco Catalyst access points to filter Bluetooth Low Energy (BLE) advertisements before forwarding them to Sensor Connect and Cisco Spaces that

- reduces unnecessary BLE advertisement traffic from dense bluetooth environments before forwarding advertisements to upstream applications
- filters BLE advertisements using Bluetooth MAC address type, RSSI threshold, and MAC address prefix criteria, evaluates BLE advertisements using the **Deny List**, **Static Allow List**, **Dynamic Allow List**, and **Limit List** in the configured order, and
- automatically forwards advertisements from onboarded BLE devices through the Dynamic Allow List after the devices are detected by an AP.

Feature history for AP and Sensor Connect support for full filtering

This section explains the feature history details of the AP and Sensor Connect support for full filtering feature.

- This table provides release and related information for the feature explained in this module.
- This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 108: Feature history for AP and Sensor Connect support for full filtering

Feature Name	Release Information	Feature Description
AP and Sensor Connect support for full filtering	Cisco IOS XE 26.2.1	AP and Sensor Connect support for full filtering enables Cisco Catalyst APs to filter Bluetooth Low Energy (BLE) advertisements before forwarding them to Sensor Connect and Cisco Spaces. It reduces unnecessary BLE traffic by applying configurable filtering criteria and ordered deny, allow, and limit lists while automatically forwarding advertisements from onboarded BLE devices.

Guidelines for BLE full filtering

Outlines best practices for configuring and managing BLE full filtering to optimize device detection and security.

Follow these guidelines when using BLE full filtering:

- BLE advertisements are evaluated in these order:
  1. Deny List
  2. Static Allow List
  3. Dynamic Allow List
  4. Limit List for default behavior

- The deny list has the highest precedence and always overrides the static and dynamic allow lists.
- Static deny and static allow lists support up to 64 configured MAC prefix entries.
- The **Dynamic Allow List** is populated automatically with onboarded BLE devices after they are detected by an AP and the advertisement events are enabled for them.
- Devices that do not match the **Deny list**, **Static Allow List**, or **Dynamic Allow List** are processed by the limit list.
- The limit list is the default forwarding behavior and is not configured as an explicit filtering list.
- By default, BLE filtering is disabled.

> **Note**
>
> The **Static Allow List** does not override the deny list. If a BLE advertisement matches both lists, the deny action takes precedence.

Launch Sensor Connect from the Catalyst 9800 controller

Configure BLE full filtering policies using the Sensor Connect application launched from the Catalyst 9800 controller.

Enable BLE full filtering to optimize IoT device communications by configuring appropriate policies with Sensor Connect.

Use the Catalyst 9800 controller to access the Sensor Connect application, which allows you to create and manage BLE full filtering policies for IoT Services.

Procedure

1. Choose **Configuration** > **Services** > **IoT Services**.
2. Click **Launch Sensor Connect**.
3. Configure the required BLE full filtering policy in the **Sensor Connect** application.

> **Note**
>
> For information about configuring BLE filtering policies, refer the Cisco Sensor Connect for IoT Services Online Help.

115 Using Cloud Monitoring as a Solution for Network Monitoring

Topics:

- Cloud monitoring

This chapter explains how to enable and verify Cisco Meraki cloud monitoring for Catalyst 9800 wireless controllers and access points, covering prerequisites, GUI/CLI onboarding, tunnel status, registration verification, and basic troubleshooting.

Cloud monitoring

This topic describes the Cloud Monitoring feature, which enables centralized monitoring of Cisco Catalyst 9800 Wireless Controllers by using the Cisco Meraki Dashboard.

Cloud Monitoring is a network management service that:

- simplifies onboarding without requiring an external onboarding agent
- improves tunnel connectivity by using native Meraki Nextunnel
- aligns operational data models with the Cisco Meraki Dashboard for seamless device authentication and operations

Cloud Monitoring allows network administrators to monitor Cisco Catalyst 9800 Wireless Controllers by using the Cisco Meraki Dashboard.

Feature history for Cloud Monitoring

This table lists release information for the feature described in this section.

This feature is available in all releases after its introduction unless noted otherwise.

Table 109: Feature history

Release	Feature	Feature information
Cisco IOS XE 17.15.1, 17.12.4, and 17.9.5	Cloud Monitoring	Cloud Monitoring is a cloud-native solution that connects devices to the cloud for network monitoring.

Examples of Cloud monitoring

- Automated onboarding requires no external onboarding agent for device registration.
- Native Meraki Nextunnel technology enhances tunnel connectivity.

When to use cloud monitoring

This topic explains when to use Cloud Monitoring and the benefits of using the Cisco Meraki Dashboard for network monitoring without additional on-premises resources.

Cloud Monitoring allows you to monitor your devices from the Cisco Meraki Dashboard without additional on-premises resources.

To monitor your network, you must log in to a specific device or deploy on-premises solutions.

To deploy an on-premises solution, you must set up additional servers and pay ongoing maintenance costs. Supporting on-premises solutions requires significant resources. Offloading these operations to the cloud reduces overhead.

Use Cloud Monitoring to monitor your devices from the Cisco Meraki Dashboard without additional resources.

Cloud monitoring features

This topic explains the features of cloud monitoring, including simplified onboarding, improved tunnel connectivity, operational data alignment, and seamless authentication with the Cisco Meraki dashboard.

This topic explains the features of cloud monitoring, including simplified onboarding, improved tunnel connectivity, operational data alignment, and seamless authentication with the Cisco Meraki dashboard.

- Simplified onboarding without any external onboarding agent.

- Improved tunnel connectivity with native Meraki Nextunnel.
- Aligning pull-based operational data with the current Cisco Meraki dashboard models.
- Seamless authentication from Cisco Meraki dashboard to the device using the cloud console.

Prerequisites for cloud monitoring

Use these guidelines when preparing controllers and user credentials for cloud monitoring with the Cisco Meraki dashboard.

- To enable cloud monitoring for controllers, the controllers must be connected to, registered, and provisioned by the Cisco Meraki dashboard.
- To add a wireless controller to a network, the username and password must have **privilege 15 access** and **enable password** (optional) in the dashboard.
- The wireless controller must have 4 unused consecutive VTY slots.

> **Note**
>
> The VTY lines must be provisioned and secured for only the dashboard to access the controller on these lines.

Enable cloud monitoring (GUI)

Learn how to enable cloud monitoring for Meraki devices using the graphical user interface (GUI) by configuring the required services and applying the changes.

Enable cloud monitoring through the GUI so that you can monitor Meraki devices centrally and have visibility into their registration and connection status.

Procedure

1. Choose **Configuration** > **Services** > **Cloud Services** > **Meraki** .
2. Use the slider to enable **Meraki Connect** .
3. Click **Apply** to automatically refresh and view the registration or Nextunnel connection status.

> **Note**
>
> Click **Refresh** to update the changes.

Cloud monitoring is activated for Meraki devices, and you can view their registration or Nextunnel connection status from the interface.

Enable cloud monitoring (CLI)

Learn how to enable cloud monitoring using CLI commands on your device.

Enable cloud monitoring features so that your device can integrate with cloud-based management tools.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable cloud monitoring.

   Example

   ```
   Device(config)# service meraki connect
   ```

Cloud monitoring is enabled on your device, allowing it to connect to the cloud management service.

Onboard the controller using Cisco Meraki dashboard

This topic explains how to onboard the controller using the Cisco Meraki dashboard for wireless device monitoring.

To monitor wireless devices, claim an eligible wireless controller into your Dashboard inventory.

For more information, see the Catalyst Wireless Onboarding Guide.

Verify cloud monitoring

Use CLI commands to verify the Cloud ID, device registration status, and access point registration for cloud monitoring.

To verify the Cloud ID (Meraki Serial Number) fetched as part of the registration and status of the operation, use this command:

```
Device# show meraki connect
Service meraki connect: enable

Meraki Tunnel Config
-----------------------------------
Fetch State:                Config fetch succeeded
Fetch Fail:
Last Fetch(UTC):            2024-07-11 15:13:07
Next Fetch(UTC):            2024-07-11 16:39:21
Config Server:              cs594-2037.meraki.com
Primary:                    apa.nt.meraki.com
Secondary:                  aps.nt.meraki.com
Client IPv6 Addr:           2001:DB8::1
Network Name:               WLC - wireless controller

Meraki Tunnel State
-----------------------------------
Primary:                    Up
Secondary:                  Up
Primary Last Change(UTC):   2024-07-09 19:02:09
Secondary Last Change(UTC): 2024-07-09 19:02:09
Client Last Restart(UTC):   2024-07-05 19:56:58

Meraki Tunnel Interface
-----------------------------------
```

```
        Status:                      Enable
        Rx Packets:                  26595318
        Tx Packets:                  32514152
        Rx Errors:                   0
        Tx Errors:                   0
        Rx Drop Packets:             0
        Tx Drop Packets:             0

        Meraki Device Registration
        ----------------------------------
        url:                        https://catalyst.meraki.com/nodes/register

        Device Number:               1
        Serial Number:               FCL264000NN
        Cloud ID:                    Q2ZZ-3HC4-5R5A
        Mac Address:                 00:00:5E:00:53:AF
        Status:                      Registered
        Timestamp(UTC):              2024-06-03 11:54:28
        Device Number:               2
        Serial Number:               FCL263900RW
        Cloud ID:                    Q2ZZ-GC8U-Y24D
        Mac Address:                 00:00:5E:00:53:B0
        Status:                      Registered
        Timestamp(UTC):              2024-06-03 11:23:55
```

To verify the access point registration status, use this command:

```
        Device# show ap meraki monitoring summary

        Meraki Monitoring          : Enabled
        Number of Supported APs    : 3

        AP Name          AP Model        Radio MAC           MAC Address
   AP Serial Number  Cloud ID         Status

 ------------------------------------------------------------------------------------------------

       APM-9164-1       CW9164I-ROW     00:00:5E:00:53:A1   00:00:5E:00:53:A2
  FGL2704LXZ5      Q5AN-2RAT-SZUE   Registered
       APM-9120-1       C9120AXI-D      00:00:5E:00:53:A3   00:00:5E:00:53:A4
  FGL2532LNR7      Q2ZZ-FL9D-HL8Z   Registered
       APM-9136-1       C9136I-ROW      00:00:5E:00:53:A5   00:00:5E:00:53:A6
  FGL2717MEFJ      Q2ZZ-VX3L-66MT   Registered
```

Troubleshoot cloud monitoring

This topic explains how to troubleshoot cloud monitoring issues, including common scenarios, reasons for failures, and recommended actions for device and access point registration problems.

This topic provides troubleshooting steps for cloud monitoring, focusing on device and access point registration issues with the Cisco Meraki Dashboard.

Scenario	Reason	Action
Your device cannot register with the Cisco Meraki Dashboard.	You see this error message: **No required SSL certificate was sent**	Check that the required certificate is installed on the device. **Note** Ensure that your device has the hardware SUDI certificates.
Your device cannot register with the Cisco Meraki Dashboard.	You see this error message: **Error message: ip http client source-interface not configured.**	Configure the **http client source interface** by using the **ip http client source-interface <interface name>** command.
If the controller fails to register with the Cisco Meraki Dashboard, it retries nine times.		Disable and then enable **service meraki connect** to reinitiate the registration.
If the access point fails to register with the Cisco Meraki Dashboard, it retries five times.	The **show ap meraki monitoring summary** command displays this status: **AP Registration Has Failed Five Times. Please Reboot the AP.**	Reload the access point to reinitiate the registration.

116 Wireless Active Testing with ThousandEyes

Topics:

- Wireless active testing (WAT)

Wireless active testing (WAT)

This section explains wireless active testing details.

Wireless active testing (WAT) is a solution that

- proactively detects and reports Wi-Fi issues before end-users encounter them
- integrates the Cisco Catalyst 9800 Wireless LAN Controller with ThousandEyes for comprehensive network and application testing, and
- helps troubleshoot Wi-Fi network unpredictability by leveraging APs as WAT devices.

WAT functionality allows you to test wireless network health and application connectivity directly from the edge of your network, with the controller managing the onboarding and configuration of these active test APs to the ThousandEyes platform.

Key aspects of wireless active testing:

- Wireless onboarding test: This part of WAT performs Layer 2 onboarding tests, including 802.11 authentication, association, EAP, key exchange, and DHCP. It reports failures and collects metrics and telemetry for successful onboarding attempts, providing insights into the wireless client experience.
- Synthetic testing with ThousandEyes: This integrates with the ThousandEyes Endpoint Agent, deployed on Catalyst APs, to run synthetic tests over the wireless connection. It checks for success or failure to various web server applications and collects metrics for application performance, offering Layer 3 and above testing capabilities directly from the AP. The controller facilitates the connection and management of these agents with the ThousandEyes platform.

Feature history for wireless active testing with ThousandEyes

This section explains the feature history details of the WAT feature with ThousandEyes.

- This table provides release and related information for the feature explained in this module.
- This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 110: Feature history for wireless active testing (WAT) with ThousandEyes

Feature Name	Release Information	Feature Description
Wireless active testing with ThousandEyes	Cisco IOS XE 17.18.2	Wireless Active Testing (WAT) proactively detects and reports Wi-Fi issues, integrating with the Cisco Catalyst 9800 Wireless LAN Controller and ThousandEyes for comprehensive network and application testing. It leverages APs as WAT devices to troubleshoot Wi-Fi network unpredictability and test network health directly from the edge.

Supported controller and AP platforms

This section explains the supported controller and AP platforms details.

Wireless active testing (WAT) is supported on Cisco Catalyst Wireless LAN Controllers and APs, offering a range of features for network monitoring and assurance by integrating with ThousandEyes.

Supported controller Platforms:

- Cisco CW9800 Series (CW9800L, CW9800M, CW9800H1, CW9800H2).
- Cisco Catalyst 9800 Series (C9800-L, C9800-40, C9800-80).
- Catalyst 9800-CL (virtual controller).

Supported AP Platforms:

- Cisco Wireless 9172H Catalyst AP (Wi-Fi 7 wall-plate model).

Supported security types

This section explains the supported security types details.

WAT supports a variety of security configurations for testing wireless networks, enabling comprehensive assessment of different WLAN setups.

This table includes the supported and non supported WLAN security types.

WLAN types	Supported types	Non supported types
AKM Suites	• 802.1X • PSK • FT-802.1X • FT-PSK • 802.1X-SHA256 • PSK-SHA256 • SAE • FT-SAE • OWE, including OWE Transition Mode	• Suite-B (11) • Suite-B-192 (12) • SAE-EXT-KEY (24) • FT-SAE-EXT-KEY (25)
Ciphers	CCMP-128	• GCMP-128 • CCMP-256 • GCMP-256

WLAN types	Supported types	Non supported types
PHY Mode / Wi-Fi versions	• HE (802.11ax / Wi-Fi 6 + 6E). • VHT (802.11ac / Wi-Fi 5). • HT (802.11n / Wi-Fi 4). • Legacy (802.11abg). **Note** All PHY modes up to Wi-Fi 6E are supported	EHT (802.11be or Wi-Fi 7)
Bands	• 2.4 GHz • 5 GHz • 6 GHz	-

Note

- WPA Support: WPA2 and WPA3 are both supported, including PMF (disabled, optional, required).
- Beacon protection is not supported.
- WAT supports IPv4 for wireless connectivity tests, ThousandEyes agent tests, and HTTP(S) proxy for management traffic.
- Hidden SSIDs are not supported.

Guidelines of wireless active testing

This section explains the guidelines of wireless active testing.

- There is a download server used to obtain the ThousandEyes endpoint agent. There is also a telemetry server, known as the platform server, with which the ThousandEyes endpoint agent communicates. Both servers require internet connectivity, either through an HTTPS proxy or through a direct connection. Ensure that the required IP addresses and URLs are allowlisted to enable access through any firewall. For more information, check Network Connections.
- The Cisco Wireless 9172H Catalyst AP does not broadcast SSIDs when WAT is enabled on the AP. They do not broadcast WLANs and do not serve clients. Use these APs exclusively for monitoring and testing. As a result, all WLANs tagged to the WAT AP in its assigned policy profile are communicated to the TE Platform, allowing selection for SSID test configuration and credentials.
- Because WAT APs require a tag, assign a policy tag based on the location of your AP and tag it as you would for any other AP. If you do not tag at least one of the AP's slots in the RF Tag with a radio profile that has WAT enabled, the AP will not be able to test.

> **Note**
>
> If you use an HTTP(S) proxy with an FQDN, configure a DNS IP address on the AP, either statically or using DHCP. The AP must resolve the proxy FQDN using DNS to connect to the ThousandEyes server.

Configure wireless active testing (GUI)

Use this procedure to configure wireless active testing using the GUI.

Enable and configure wireless active testing (WAT) on your controller and associated APs for integration with ThousandEyes using the GUI.

This task involves setting up global WAT parameters on the controller, configuring AP-specific settings, and enabling WAT on selected APs. It allows APs to act as active test APs, performing wireless and application tests managed by the ThousandEyes platform.

Before you begin

Ensure your controller and APs are running compatible software versions (17.18.2 or later is required).

- You must have a ThousandEyes account and obtain the **ThousandEyes Endpoint Agent Connection String** found in the **Endpoint Experience** section from your ThousandEyes dashboard account settings. This string links your controller and APs to your ThousandEyes organization.

Procedure

1. Choose **Configuration** > **Services** > **Cloud Services**.
2. Click the **Thousand Eyes** tab.
3. The **Wireless Active Testing (WAT) status** toggle is **Enabled** by default.
4. Paste the **ThousandEyes Endpoint Agent Connection String** from the ThousandEyes platform.
5. Click **Apply**.

Configure wireless active testing (CLI)

Use this procedure to configure wireless active testing using commands.

Enable and configure Wireless active testing (WAT) on your controller and associated APs for integration with ThousandEyes using commands.

Before you begin

Ensure your controller and APs are running compatible software versions (17.18.2 or later is required).

- You must have a ThousandEyes account and obtain the **ThousandEyes Endpoint Agent Connection String** found in the **Endpoint Experience** section from your ThousandEyes platform Endpoint Agent settings. This string links your controller and APs to your ThousandEyes organization.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure ThousandEyes endpoint agent connection string.

Example

```
Device(config)# wireless active testing thousand-eyes connection-string STRING
```

This string associates the AP's ThousandEyes agent with your ThousandEyes organization. Use the **no** form of the command to remove the connection string.

> **Note**
>
> Removing the connection string unregisters all WAT APs from the ThousandEyes Platform.

3. Exit the global configuration mode.

Example

```
Device(config)# end
```

Configure HTTP(S) proxy for WAT (GUI)

Use this procedure to configure HTTP(S) proxy for WAT using the GUI.

Follow these steps to configure an HTTP(S) proxy for WAT management traffic to the ThousandEyes server, using GUI.

Before you begin

Unauthenticated HTTP forward proxy is supported.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. Click **Add**.
   The **Add AP Join Profile** page appears.
3. In the **General** tab, enter the **Name** and **Description** for the AP join profile.
4. In the **Management** tab, go to the **HTTP Proxy Configuration** section.
5. Enter the IPV4 or Hostname in the **Client Proxy Server** field.
6. Enter a value in the **Client Proxy Port** field.
7. Click **Apply to Device**.

Configure HTTP(S) proxy for WAT (CLI)

Use this procedure to configure HTTP(S) proxy for WAT using commands.

Follow these steps to configure an HTTP(S) proxy for WAT management traffic to the ThousandEyes server, using commands.

Before you begin

Unauthenticated HTTP forward proxy is supported.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter AP Join Profile configuration mode.

   Example

   ```
   Device(config)# ap profile default-ap-profile
   ```

3. Configure the HTTP(S) proxy.

   Example

   ```
   Device(config-ap-profile)# ip http client proxy hostname URL PORT
   ```

   The valid value range for the port number is 0 to 65535. The **URL** needs to be in the format of `{http|https}://<FQDN or IP>[/<path>]`.

   > **Note**
   >
   > This proxy enables the ThousandEyes Agent IOx package download and manages all subsequent communications between the ThousandEyes Agent and the ThousandEyes platform.

4. Exit AP Join Profile configuration mode.

   Example

   ```
   Device(config-ap-profile)# end
   ```

Configure WAT radio selection and tag the RF profile (GUI)

Use this procedure to configure WAT radio selection and tag the RF profile using the GUI.

Follow these steps to configure the radio profile and RF tag to designate AP radios for WAT using the GUI.

This process helps you to identity the APs that you want to become WAT APs.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF/Radio**.
2. Create a new radio profile in the **Radio** tab and then select the profile.
   The **Edit Radio Profile** tab appears.
3. Select **Enabled (Always)** for the **Wireless Active Testing Radio Selection** field's drop-down list.

   The options are **Enabled (Always)** and **Disabled**.
4. Click **Update & Apply to Device**.
5. Choose **Configuration** > **Tags & Profiles** > **Tags**.
6. In the **RF** tab, click the newly created and configured RF profile.
   The **Edit RF Tag** page appears.
7. Tag the profile to **Slot 2** for the **6 GHz Radio Profile** band, to **Slot 1** for the **5 GHz Radio Profile** band and to **Slot 0** for the **2.4 GHz Radio Profile** band.
8. Click **Update & Apply to Device**.

Configure wireless active testing radio selection (CLI)

Use this procedure to configure wireless active testing radio selection using commands.

Follow these steps to configure the radio profile and RF tag to designate AP radios for WAT using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Create or modify a radio profile.

   Example

   ```
   Device(config)# wireless profile radio RADIO_PROFILE_NAME
   ```
3. Enable wireless active testing radio selection within the radio profile.

   Example

   ```
   Device(config-wireless-radio-profile)# active testing radio-selection always
   ```
4. Exit radio profile configuration mode and create or modify an RF tag.

   Example

   ```
   Device(config)# wireless tag rf RF_TAG_NAME
   ```
5. Tag the profile to **Slot 2** for the **6 GHz Radio Profile** band, to **Slot 1** for the **5 GHz Radio Profile** band and to **Slot 0** for the **2.4 GHz Radio Profile** band.

Example

```
Device((config-wireless-rf-tag))# dot11 24ghz slot0 radio-profile
RADIO_PROFILE_NAME
```

```
Device(config-wireless-rf-tag)# dot11 5ghz slot1 radio-profile
RADIO_PROFILE_NAME
```

```
Device(config-wireless-rf-tag)# dot11 6ghz slot2 radio-profile
RADIO_PROFILE_NAME
```

6. Exit RF tag configuration mode.

Example

```
Device(config)# end
```

Enable WAT management for an individual AP (GUI)

Use this procedure to enable WAT management for an individual AP using the GUI.

Follow these steps to enable or disable WAT Management on an individual AP using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. Click the **AP** tab.
3. Go to the **Static** tab for a per AP enablement. Click **Add.**
4. Enter the MAC address in the **AP MAC Address** field.
5. Select the policy-tag, site-tag and RF-tag in the **Policy Tag Name**, **Site Tag Name** and **RF Tag Name** fields as per preference.
6. Click **Apply to Device**.

Enable WAT management for an individual AP (CLI)

Use this procedure to enable WAT management for an individual AP using commands.

Follow these steps to enable or disable WAT management on individual APs using commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the Ethernet MAC address of the AP.

Example

```
Device(config)# ap H.H.H
```

3. Map a policy tag to the AP.

   Example

   ```
   Device(config)# policy-tag default-policy-tag
   ```

   Use the **no** form of this command to disable it.

4. Map a RF tag to the AP.

   Example

   ```
   Device(config)# rf-tag wat-rf-tag
   ```

   Use the **no** form of this command to disable it.

5. Map a site tag to the AP.

   Example

   ```
   Device(config)# site-tag wat-site-tag
   ```

   Use the **no** form of this command to disable it.

6. Exit the global configuration mode.

   Example

   ```
   Device(config)# end
   ```

Enable WAT management for multiple APs using location (GUI)

Use this procedure to enable WAT management for multiple APs using location from the GUI.

Follow these steps to enable or disable WAT Management on multiple APs using the location process from the GUI.

Before you begin

For multiple APs, follow these steps of enabling WAT management for multiple APs using location or by using the filter process.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. Click the **AP** tab.
3. Go to the **Location** tab. Click **Add**.
4. Enter a location name and description in the **Location** and **Description** fields.
5. In the **General** tab, select the policy-tag, site-tag and RF-tag in the **Policy Tag Name**, **Site Tag Name** and **RF Tag Name** fields as per preference.

6. Click **Apply to Device**.
7. In the **AP Provisioning** tab, select or add APs in the **Import AP MAC**, **AP MAC Address**, **Available AP list** and **Associated AP list** fields to associate them with the location.
8. Click **Apply to Device**.

Enable WAT management for multiple APs using location (CLI)

Use this procedure to enable WAT management for multiple APs using location from the console.

Follow these steps to enable or disable WAT Management on multiple APs using the location process from the console.

Before you begin

For multiple APs, follow these steps of enabling WAT management for multiple APs using location or by using the filter process.

Procedure

1. Enter the global configuration mode and modify or create AP location configuration.

   Example

   ```
   Device# configure terminal
   ```

   ```
   Device(config)# ap location name wat-location
   ```

2. Add AP to this location.

   Example

   ```
   Device(config-ap-location)# ap-eth-mac H.H.H
   ```

   Use the **no** form of this command to disable it.

3. Configure tags based for this location, configure policy tag for this location and configure rf and site tag for this location.

   Example

   ```
   Device(config-ap-location)# tag policy default-policy-tag
   ```

   ```
   Device(config-ap-location)# tag rf wat-rf-tag
   ```

   ```
   Device(config-ap-location)# tag site wat-site-tag
   ```

   Use the **no** form of this command to disable it.

4. Exit the global configuration mode.

   Example

   ```
   Device(config-ap-location)# end
   ```

Enable WAT management for multiple APs using filter (GUI)

Use this procedure to enable WAT management for multiple APs using filter from the GUI.

Follow these steps to enable or disable WAT Management on multiple APs using the filter process from the GUI.

Before you begin

For multiple APs, follow these steps of enabling WAT management for multiple APs using location or by using the location process.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. Click the **AP** tab.
3. Go to the **Filter** tab for a regex on the AP name and select a WAT AP.
   The **Edit Tags** page appears.
4. Enter the **Rule Name** and the **AP name regex**.

   This results in the tag going to only those selected APs.

5. In the **Active** field, enable the toggle to **Yes**. Click **Update & Apply to Device**.

   Note

   Tags have a priority order. In the **Tag Source** tab, the default setting is static, so the per-AP priority is zero. If no per-AP tag exists, the system checks the location. If the location is not set, it checks for a filter. If there is no filter, the system checks whether the AP has an assigned tag and saves it. To enable this feature, check the **Enable AP Tag Persistency** checkbox. Click **Apply**.

Enable WAT management for multiple APs using filter (CLI)

Use this procedure to enable WAT management for multiple APs using filter from the console.

Follow these steps to enable or disable WAT Management on multiple APs using the filter process from the console.

Before you begin

For multiple APs, follow these steps of enabling WAT management for multiple APs using location or by using the location process.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure ap filter and enter AP filter name.

Example

```
Device(config-ap-filter)# ap filter name wat-aps
```

3. Configure filter based on AP name regular expression match.

   Example

   ```
   Device(config-ap-filter)# ap name-regex WAT-9172H
   ```

   Use the **no** form of this command to disable it.

4. Configure tags based for this filter, configure policy tag for this filter and configure rf and site tag for this filter.

   Example

   ```
   Device(config-ap-filter)# tag policy default-policy-tag
   ```

   ```
   Device(config-ap-filter)# tag rf wat-rf-tag
   ```

   ```
   Device(config-ap-filter)# tag site wat-site-tag
   ```

   Use the **no** form of this command to disable it.

5. Exit the global configuration mode.

   Example

   ```
   Device(config-ap-filter)# end
   ```

Verify wireless active testing

This section provides verifying wireless active testing details.

Use these commands to verify the status of WAT.

- **show run | sec wat-radio-profile**
- **show ap profile name wat-ap-profile detailed**
- **show wireless profile radio detailed wat-radio-profile**
- **show wireless tag rf detailed wat-rf-tag**
- **show wireless active testing configuration**

To verify the WAT radio profile, enter this command from the console:

```
Device# show run | sec wat-radio-profile

wireless profile radio wat-radio-profile
 active testing radio-selection always
Device# show run | sec wat-rf-tag

wireless tag rf wat-rf-tag
```

```
 dot11 24ghz slot0 radio-profile wat-radio-profile
 dot11 5ghz slot1 radio-profile wat-radio-profile
 dot11 6ghz slot2 radio-profile wat-radio-profile
```

To verify the WAT AP profile detailed, enter this command from the console:

```
Device# show ap profile name wat-ap-profile detailed
…
Proxy :
  Hostname                    : http://192.168.1.100
  Port                        : 80
  NO_PROXY list               : Not Configured
  Username                    : Not Configured
…
```

To verify the WAT radio profile detailed, enter this command from the console:

```
Device# show wireless profile radio detailed wat-radio-profile
…
Wireless Active Testing (WAT) Configuration
  WAT Admin State (Radio Selection)              : Enabled (Always)
…
```

To verify the WAT RF tag detailed, enter this command from the console:

```
Device# show wireless tag rf detailed wat-rf-tag

Tag Name                      : wat-rf-tag
Description                   :
---------------------------------------
6ghz RF Policy                : default-rf-profile-6ghz
5ghz RF Policy                : Global Config
2.4ghz RF Policy              : Global Config
2.4ghz slot 0 Radio Profile   : wat-radio-profile
5ghz slot 0 Radio Profile     : default-radio-profile
5ghz slot 1 Radio Profile     : wat-radio-profile
6ghz slot 1 Radio Profile     : default-radio-profile
5ghz slot 2 Radio Profile     : default-radio-profile
6ghz slot 2 Radio Profile     : wat-radio-profile
6ghz slot 3 Radio Profile     : default-radio-profile
AP Beam State                 : Boresight
URWB Profile                  :
```

To verify the WAT wireless active testing configuration, enter this command from the console:

```
Device# show wireless active testing configuration

Wireless Active Testing (WAT) Management - Configuration
=========================================================
Administrative State                          : Enabled

ThousandEyes Endpoint Agent
---------------------------
  Connection String                           :
S3FD77CY+VOrk+*****************************************************************

  Download URL                                :
https://downloads.thousandeyes.com/endpointagent/iox/arm64/latest.tar
```

117 Wireless Active Testing SCEP support

Topics:

- Wireless Active Testing SCEP support

This chapter explains Wireless Active Testing SCEP support.

Wireless Active Testing SCEP support

Explains how Wireless Active Testing SCEP support provisions, renews, and manages client certificates for EAP-TLS SSIDs, integrating with customer SCEP servers to automate certificate management.

Wireless Active Testing SCEP support is a certificate provisioning capability that

- automatically enrolls client certificates for Wireless Active Testing EAP-TLS SSIDs
- eliminates manual client certificate provisioning by integrating with a customer SCEP server, and
- automatically renews client certificates before they expire.

Key features of Wireless Active Testing SCEP support

Wireless Active Testing SCEP support:

- supports standard RFC 8894 SCEP servers, including Microsoft NDES and EJBCA.
- continues to support PKCS#12 and LSC certificate sources for deployments that do not use SCEP.

Configure the SCEP parameters for the Wireless Active Testing SSID from the ThousandEyes Dashboard. For detailed configuration steps, refer the ThousandEyes Endpoint Experience documentation. After the configuration is pushed, the Wireless Active Testing Manager automatically provisions and manages the client certificate.

Feature history for Wireless Active Testing SCEP support

This section explains the feature history details of the Wireless Active Testing SCEP support feature.

- This table provides release and related information for the feature explained in this module.
- This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 111: Feature history for Wireless Active Testing SCEP support

Feature Name	Release Information	Feature Description
Wireless Active Testing SCEP support	Cisco IOS XE 26.2.1	Wireless Active Testing SCEP support is a certificate-provisioning capability that automatically enrolls and renews client certificates for EAP-TLS SSIDs by integrating with customer SCEP servers, eliminating manual certificate provisioning. It supports RFC 8894-compliant SCEP servers, including Microsoft NDES and EJBCA, while retaining support for PKCS#12 and LSC certificate sources.

SCEP enrollment for Wireless Active Testing

Explains the certificate provisioning process for Wireless Active Testing using SCEP to automate client certificate enrollment and renewal for EAP-TLS SSIDs.

SCEP enrollment for Wireless Active Testing is a certificate provisioning process that

- automatically requests and installs a client certificate from a customer SCEP server for each Wireless Active Testing EAP-TLS SSID
- replaces manual certificate provisioning by using automated certificate enrollment, and
- maintains certificate availability by supporting automatic certificate renewal.

SCEP enrollment process

When you configure SCEP settings for a Wireless Active Testing SSID from the ThousandEyes dashboard, the ThousandEyes agent sends the SCEP configuration to the AP. The Wireless Active Testing Manager creates a trustpoint for the SSID and starts the certificate enrollment process.

During enrollment, the Wireless Active Testing Manager:

- retrieves the CA certificate from the SCEP server.
- validates the configured CA fingerprint, when fingerprint validation is enabled.
- generates the private key and certificate signing request (CSR).
- requests a client certificate from the SCEP server.
- stores the enrolled certificate, private key, and CA certificate in the Wireless Active Testing trustpoint.

After successful enrollment, Wireless Active Testing uses the enrolled client certificate to perform EAP-TLS authentication for the configured SSID. When automatic re-enrollment is enabled, the Wireless Certificate Platform (WCP) renews the certificate before it expires without requiring manual certificate replacement.

Guidelines for Wireless Active Testing SCEP

Outlines recommended actions for configuring and deploying Wireless Active Testing SCEP to ensure reliable certificate enrollment and secure network operation.

Follow these guidelines when deploying Wireless Active Testing SCEP.

- Configure all SCEP parameters from the ThousandEyes dashboard.
- Use AP or controller commands only for verification and troubleshooting.
- Ensure that the AP can reach the SCEP server.
- Synchronize AP time using NTP before certificate enrollment.
- Configure the **Common Name** and **Server URL** correctly before saving the SSID configuration.
- Verify the optional CA fingerprint before enabling fingerprint validation.

Part X

Security

Topics:

- MAC Filtering
- Web-Based Authentication
- Central Web Authentication
- Private Shared Key
- Multi-Preshared Key
- Wireless AAA Authentication Survivability Cache
- Wi-Fi Protected Access 3
- WPA3 Security Enhancements for Access Points
- IP Source Guard
- Traffic Filtering on AP by Source IP ACL
- 802.11w
- Management Frame Protection
- IPv4 ACLs
- Downloadable ACL
- DNS-Based Access Control Lists
- Allowed List of Specific URLs
- Cisco Umbrella WLAN
- RADIUS Server Load Balancing
- AAA Dead-Server Detection
- ISE Simplification and Enhancements
- RADIUS DTLS
- Policy Enforcement and Usage Monitoring
- Local Extensible Authentication Protocol
- Local EAP Ciphersuite
- Authentication and Authorization Between Multiple RADIUS Servers
- CUI Information in RADIUS Accounting
- Secure LDAP
- Network Access Server Identifier
- Locally Significant Certificates
- Public Key Infrastructure Management
- Controller Self-Signed Certificate for Wireless AP Join
- Managing Rogue Devices
- Classifying Rogue Access Points
- Advanced WIPS

- Cisco TrustSec
- SGT Inline Tagging and SXPv4
- Ciphersuites for CAPWAP-DTLS
- Configure Secure Shell
- MACsec support on APs
- FIPS
- Internet Protocol Security
- Transport Layer Security Tunnel Support
- Configuring RFC 5580 Location Attributes
- IP MAC Binding
- Disabling IP Learning in FlexConnect Mode
- Disabling Device Tracking to Support NAC Devices
- Disabling IP Learning in Local Mode
- Security-Enhanced Linux

118 MAC Filtering

Topics:

- MAC Filtering

MAC Filtering

MAC Filtering details.

You can configure the controller to authorize clients based on the client MAC address by using the MAC filtering feature.

When MAC filtering is enabled, the controller uses the MAC address as the client identity. The authentication server has a database of client MAC addresses that are allowed network access. The controller sends the authentication server a RADIUS-access/request frame with a username and password based on the client MAC address as soon as it gets the association request from the client. If authorization succeeds, the controller sends a successful association response to the client. If authorization fails, the controller rejects the client association.

Clients that were authorized with MAC filtering can be re-authenticated through the WLAN session timeout feature.

MAC Filtering Configuration Guidelines

- MAC filtering authentication occurs at the 802.11 association phase and delays the association response until authentication is done. If you use a RADIUS server for MAC filtering, it is advised to keep a low latency between the controller and the RADIUS server. When latency is too high, the client might timeout while waiting for the association response.
- MAC filtering can be combined with other authentication methods such as 802.1X, Pre-Shared Key or it can be used alone.
- MAC addresses can be spoofed and MAC filtering does not consist in a security measure.
- Many clients can use a private MAC address to connect and change it at every session, therefore making it harder to identify devices through their MAC address.

Note

If wlan-profile-name is configured for a user, guest user authentication is allowed only from that WLAN.

If wlan-profile-name is not configured for a user, guest user authentication is allowed on any WLAN.

The AP fails to join the controller due to an authentication rejection on the RADIUS server. The failure occurs on the Cisco Catalyst 9800 controller, only when the RADIUS server is configured to authenticate the APs with method MAB as endpoints. The reason is that the RADIUS calling-station-id attribute is required for MAB authentication and is not present within the access request packet during the AP join. The workaround is to use a different AP authentication method than MAB as endpoints such as PAP-ASCII using a username and a password.

If you want the client to connect to SSID1, but not to SSID2 using mac-filtering, ensure that you configure **aaa-override** in the policy profile.

In the following example, when a client with MAC address 112233440001 tries to connect to a WLAN, the request is sent to the local RADIUS server, which checks the presence of the client MAC address in its attribute list (FILTER_1 and FILTER_2). If the client MAC address is listed in an attribute list (FILTER_1), the client is allowed to join the WLAN (WLAN_1) that is returned as *ssid attribute* from the RADIUS server. The client is rejected, if the client MAC address is not listed in the attribute list.

Local RADIUS Server Configuration

```
!Configures an attribute list as FILTER_2
aaa attribute list FILTER_2
!Defines an attribute type that is to be added to an attribute list.
attribute type ssid "WLAN_2"
```

```
!Username with the MAC address is added to the filter
username 112233440001 mac aaa attribute list FILTER_2

!
aaa attribute list FILTER_1
attribute type ssid "WLAN_1"
username 112233440001 aaa attribute list FILTER_1
```

Controller Configuration

```
! Sets authorization to the local radius server
aaa authorization network MLIST_MACFILTER local

!A WLAN with the SSID WLAN_2 is created and MAC filtering is set along with
security
parameters.
wlan WLAN_2 2 WLAN_2
mac-filtering MLIST_MACFILTER
no security wpa
no security wpa wpa2 ciphers

!WLAN with the SSID WLAN_1 is created and MAC filtering is set along with
security parameters.
wlan WLAN_1 1 WLAN_1
mac-filtering MLIST_MACFILTER
no security wpa
no security wpa wpa2 ciphers aes
no security wpa akm dot1x
security web-auth
security web-auth authentication-list WEBAUTH

! Policy profile to be associated with the above WLANs
wireless profile policy MAC_FILTER_POLICY
aaa-override
vlan 504
no shutdown
```

Configuring MAC Filtering for Local Authentication (CLI)

Configure MAC Filtering for Local Authentication using commands.

Follow the procedure given below to configure MAB for local authentication.

Before you begin

Configure AAA local authentication.

Configure the username for WLAN configuration (local authentication) using **username** *mac-address* **mac** command.

Note

The mac-address must be in the following format: *abcdabcdabcd*

Procedure

1. **wlan** *profile-name wlan-id*

   Example

   ```
   wlan CR1_SSID_mab-local-default 1 CR1_SSID_mab-local-default
   ```

   Specifies the WLAN name and ID.

2. **mac-filtering default**

   Example

   ```
   Device(config-wlan)# mac-filtering default
   ```

   Sets MAC filtering support for the WLAN.

3. **no security wpa**

   Example

   ```
   Device(config-wlan)# no security wpa
   ```

   Disables WPA secuirty.

4. **no security wpa akm dot1x**

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

   Disables security AKM for dot1x.

5. **no security wpa wpa2**

   Example

   ```
   Device(config-wlan)# no security wpa wpa2
   ```

   Disables WPA2 security.

6. **no security wpa wpa2 ciphers aes**

   Example

   ```
   Device(config-wlan)# no security wpa wpa2 ciphers aes
   ```

Disables WPA2 ciphers for AES.

7. **no shutdown**

Example

```
Device(config-wlan)# no shutdown
```

Enables the WLAN.

Configuring MAC Filtering (GUI)

Configure MAC Filtering using GUI.

Before you begin

Configure AAA external authentication.

Procedure

1. Choose **Configuration** > **Wireless** > **WLANs**.
2. On the **Wireless Networks** page, click the name of the WLAN.
3. In the **Edit WLAN** window, click the **Security** tab.
4. In the **Layer2** tab, check the **MAC Filtering** check box to enable the feature.
5. With MAC Filtering enabled, choose the **Authorization List** from the drop-down list.
6. Save the configuration.

Configuring MAB for External Authentication (CLI)

Configure MAB for External Authentication using commands.

Follow the procedure given below to configure MAB for external authentication.

Before you begin

Configure AAA external authentication.

Procedure

1. **wlan** *wlan-name wlan-id ssid-name*

Example

```
wlan CR1_SSID_mab-ext-radius 3 CR1_SSID_mab-ext-radius
```

Specifies the WLAN name and ID.

2. **mac-filtering** *list-name*

Example

```
Device(config-wlan)# mac-filtering ewlc-radius
```

Sets the MAC filtering parameters. Here, *ewlc-radius* is an example for the *list-name*

3. **no security wpa**

   Example

   ```
   Device(config-wlan)# no security wpa
   ```

   Disables WPA secuirty.

4. **no security wpa akm dot1x**

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

   Disables security AKM for dot1x.

5. **no security wpa wpa2**

   Example

   ```
   Device(config-wlan)# no security wpa wpa2
   ```

   Disables WPA2 security.

6. **mab request format attribute {1 groupsize** *size* **separator** *separator* **[lowercase | uppercase] | 2 {0 | 7 | LINE} LINE** *password* **| 32 vlan access-vlan}**

   Example

   ```
   Device(config)# mab request format attribute 1 groupsize 4 separator
   ```

   Optional. Configures the delimiter while using MAC filtering in a WLAN.

   Here,

   1- Specifies the username format used for MAB requests.

   groupsize *size*- Specifies the number of hex digits per group. The valid values range from 1 to 12.

   separator *separator*- Specifies how to separate groups. The separators are comma, semicolon, and full stop.

   lowercase- Specifies the username in lowercase format.

   uppercase- Specifies the username in uppercase format.

   2- Specifies the global password used for all the MAB requests.

   0- Specifies the unencrypted password.

   7- Specifies the hidden password.

   LINE- Specifies the encrypted or unencrypted password.

   password- LINE password.

   32- Specifies the NAS-Identifier attribute.

vlan- Specifies a VLAN.

access-vlan- Specifies the configured access VLAN.

7. **no security wpa wpa2 ciphers aes**

   Example

   ```
   Device(config-wlan)# no security wpa wpa2 ciphers aes
   ```

   Disables WPA2 ciphers for AES.

8. **no shutdown**

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

   Enables the WLAN.

119 Web-Based Authentication

Topics:

- Local web authentication
- External Web Authentication
- Authenticating sleeping clients
- Sleeping Clients With Multiple Authentication
- Multiple Authentication Combinations with 802.1X and LWA

This chapter describes how to configure web-based authentication on the device. It contains these sections:

Local web authentication

Introduces local web authentication concepts, detailing authentication process, device roles, usage restrictions, customizable web pages, banners, and guidelines. Guides configuration through GUI and CLI, including HTTP/HTTPS requests, parameter maps, WLANs, ACLs, banners, TrustPoint, and provides practical configuration examples.

Local web authentication is a network security mechanism that

- authenticates users through a web browser login page
- enables access control on host systems that do not run IEEE 802.1X supplicants, and
- communicates with authentication, authorization, and accounting (AAA) servers to enforce security policies.

Feature history

Feature Name	Release	Description
Built-in Captive Portal Improvement	Cisco IOS XE 17.1.1	This release introduces support for special characters in the login portal banner title and banner text. The number of characters supported on the banner text has doubled to 400. The **exec-character-bits** command has been introduced.

Presentation options for local web authentication web pages

Local web authentication intercepts HTTP sessions on Layer 2 interfaces, and in some cases, Layer 3 interfaces (with restrictions for some switch models). When users try to access the network, local web authentication displays a login page and verifies user credentials with AAA servers, granting or denying access accordingly.

Local web authentication is categorized by the location where its web pages are hosted.

- **Internal**: Uses HTML pages (login, success, fail, and expire) stored on the controller .
- **Customized**: Uses customized HTML pages (login, success, fail, and expire) downloaded onto the controller for a customized user experience.
- **External**: Uses HTML pages hosted on an external web server.

We recommend that you follow the Cisco guidelines to create a customized web authentication login page. If you use the latest versions of Google Chrome or Mozilla Firefox browsers, ensure that your webauth bundle uses this line in the *login.html* file:

```
<body onload="loadAction();">
```

Web authentication modes

The types of web authentication differ according to the available web authentication pages.

- **Webauth**–The controller displays a page with the user name and password. Users enter valid credentials to gain network access.
- **Consent or web-passthrough**–The controller presents a policy page with the **Accept** and **Deny** buttons. Users simply click **Accept** to access the network – no credentials are required.

- **Webconsent**–This mode combines the features of Webauth and Consent. The controller displays a policy page with Accept or Deny buttons along with user name or password. Users must enter the correct credentials and click **Accept** to access the network.

Additional reference information

- You can view the webauth parameter-map information using the **show running-config** command.
- If you see tracebacks during client authentication, performance and behavior are not affected. Tracebacks may occur if Flexible Forwarding Mode (FFM) replies to Endpoint Profiler Module (EPM) for ACL application after the session is dequeued, usually because a timer expired or the session became **unauthorized**.
- Apply web authentication methods (such as consent, web consent, and webauth) using either a global or named parameter-map under WLAN (for method-type, custom, and redirect). If you do not configure a parameter-map under WLAN, the global parameter-map applies by default.
- You can configure web-based authentication on Layer 2 and layer 3 interfaces.
- When a client reaches the maximum limit of 200 HTTP connections, the system resets TCP connections and excludes your client from the network.

How local web authentication works

Summary

authentication enables secure network access for clients by prompting users to authenticate through a web login page. The process coordinates actions between the user, authentication server, and the network switch. The key components are:

- **User**: Initiates the HTTP session and enters authentication credentials.
- **Network switch**: Intercepts traffic, presents login pages, applies policies, and communicates with the authentication server.
- **Authentication server**: Verifies credentials and provides policy enforcement details.

Workflow

These are the stages of the process.

1. **Session initiation**: The user starts an HTTP session by attempting to access the network.
2. **Traffic interception and login page presentation**: The network switch intercepts the HTTP request and triggers the authorization process. It presents a login page for the user to enter their username and password.
3. **Credential submission and authentication**: After the user submits credentials, the switch forwards them to the authentication server
4. **Authentication outcome:**
   - If authentication succeeds, the switch downloads and activates the access policy assigned to the user from the server, then displays a login success page.
   - If authentication fails, the switch displays a login failure page. After a maximum number of failures, the login expired window appears, and the host is put on a watch list. Once the timeout expires, the user may try again
5. **Server non-response handling**: If the authentication server does not respond, and an AAA fail policy is in place, the switch applies the failure policy to the host and displays a login success page.
6. **Reauthentication triggers**: The switch reauthenticates a client if the host does not respond to an ARP probe (Layer 2) or does not send traffic within an idle timeout (Layer 3)

suppress-feature-id="uabu_2960l_sw"

.

7. **Session timeout enforcement**: The switch applies either the session timeout configured locally or provided by the server. The default local web authentication session timeout on controller is 1800 seconds from Cisco IOS XE 16.1.1 and later. The default session timeout value was infinite seconds, prior to Cisco IOS XE Denali 16.1.1.
8. **Session termination:**
   - If the terminate action is set to RADIUS, the switch sends a nonresponsive host request to the server; the server's response dictates the next action.
   - If the terminate action is set to default, the switch dismantles the session and removes the access policy.

Result

If authentication succeeds and the correct policies are applied, the user gains network access. If authentication fails, failure policies or timeout mechanisms may deny access.

Restrictions

- You cannot configure bypass authentication with the wireless web authentication feature.
- To update the redirect login URL, enable and then disable the WLAN. The redirect login URL specified in the web authentication parameter map changes when an AP rejoins.
- If authentication fails, users receive a failure page and can try to log in again. If the number of allowed attempts is exceeded, users may be excluded and receive a specific reason for the exclusion.
- Use the local web authentication feature to authenticate end users on host systems that do not run the IEEE 802.1x supplicant.

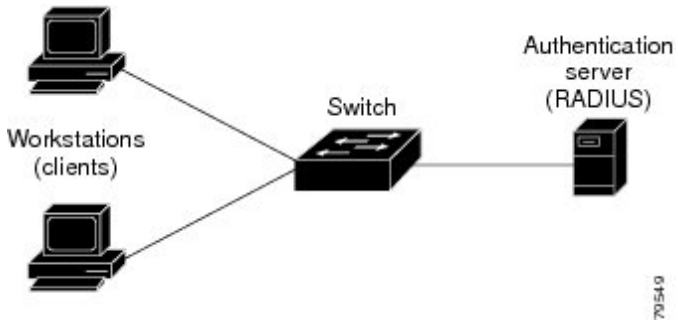
Roles of devices in local web authentication

In a local web authentication scenario, network devices assume specific roles to manage authentication and access to the LAN.

- **Client**: A device, such as a workstation, that requests network access and responds to authentication requests from the switch. The client must have an HTML browser with JavaScript enabled.
- **Authentication server**: A server that validates the identity of the client. The authentication server notifies the switch if the client is allowed or denied access to the LAN and related services.
- **Switch**: A network device that manages physical access to the network based on the authentication status of the client. The switch relays identity information and authorization responses between the client and the authentication server.

These device roles work together to ensure secure access control through local web authentication processes.

Figure 12: Local web authentication device roles



Banner messages for local web authentication

Local web authentication banners provide visual feedback to users during authentication on switches. The banners can display default or customized messages. They may also include branding or additional information on the login and result screens.

Default banner messages

When web authentication is enabled, one of these default messages appears on both the login screen and the authentication result pop-up page:

- Authentication Successful
- Authentication Failed
- Authentication Expired

Commands to configure local web authentication banner

You can configure the local web authentication banner using the new style (Session-aware) CLI mode.

Use the command in the global configuration mode:

```
Device(config)# parameter map type webauth global
Device(config-params-parameter-map)# banner ?
file <file-name>
text <Banner text>
title <Banner title>
```

You can add a custom message, such as a switch, router, or company name, to the banner using the command:

```
Device(config)# parameter map type webauth global
Device(config-params-parameter-map)# banner text <text>
```

You can add a logo or text file to the banner using the command:

```
Device(config)# parameter map type webauth global
Device(config-params-parameter-map)# banner text <filepath>
```

Banner examples

Figure 13: Customized web banner: Authentication Successful

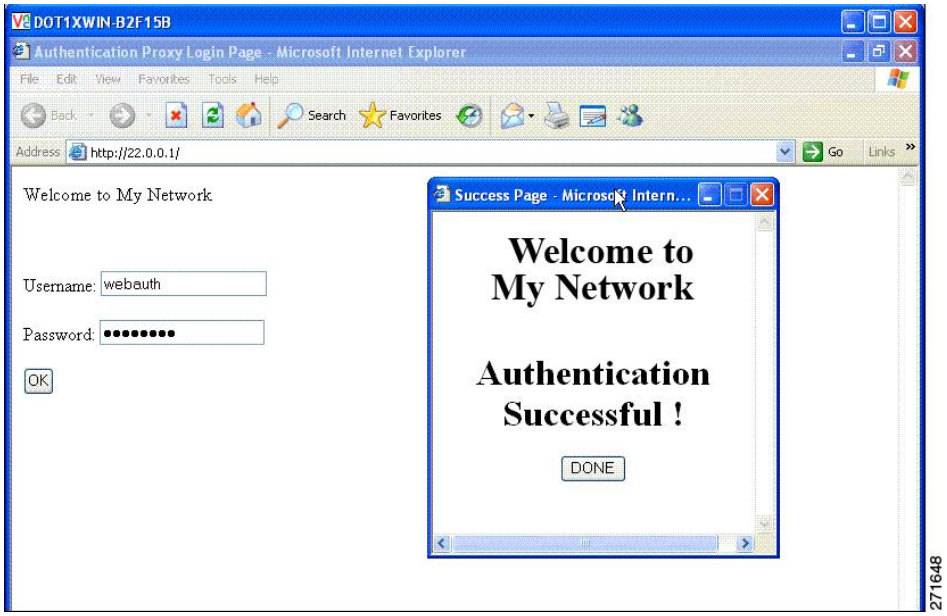
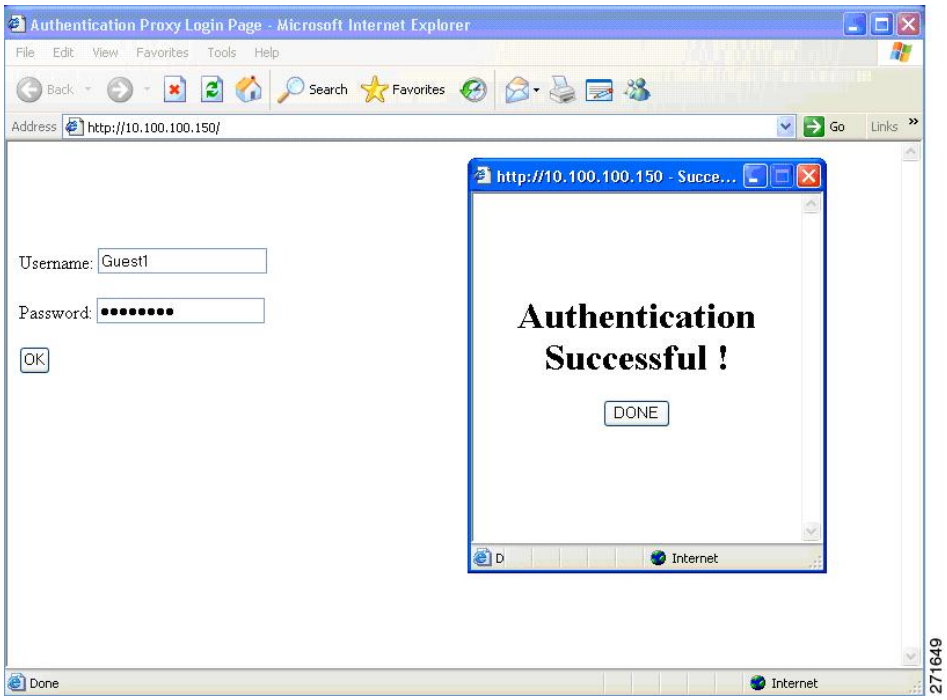


Figure 14: Login screen with no banner



Banner usage and behavior

- The default banners are *Cisco Systems* and *Switch host-name Authentication*. These banners appear on the login page. The *Cisco Systems* banner also appears on the authentication result pop-up page.
- The default banners appear unless custom banners are configured.
- If you do not enable a banner, only username and password dialog boxes appear on the web authentication login screen. You will not see a banner when you log into the switch.

Authentication states in customized local web authentication

During local web authentication, the switch's internal HTTP server hosts four HTML pages that communicate authentication states to the client.

You can replace the default internal HTML pages or specify a URL that redirects users after authentication. This URL replaces the internal Success page.

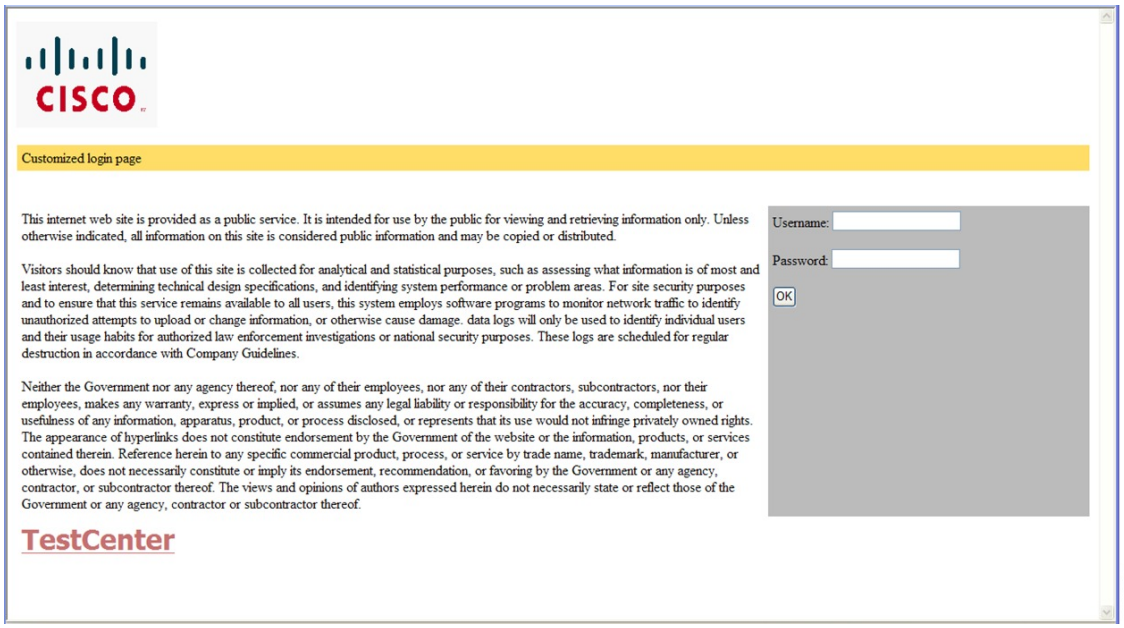
The four authentication pages and states are:

- Login: Credentials are requested from the client.
- Success: The client has authenticated successfully.
- Fail: Authentication attempt failed.
- Expire: The session expires after excessive failures.

You must configure all four pages. You can specify text or add a logo to each of the four pages.

Banner examples

Figure 15: Customizable Authentication Page



Best practices for customizing web authentication pages

- Add appropriate text to the banner and login pages as needed.
- Always include a valid HTML redirect command in the success page to redirect users to a specific URL after login
- Ensure the URL string is well-formed (for example, "http://www.cisco.com") to avoid browser errors.
- If you configure web pages for HTTP authentication, include the appropriate HTML commands. For example, HTML commands to set the page time out, to set a hidden password, or to confirm that the same page is not submitted twice.
- You can copy the configured web pages to the switch boot flash or the flash.
- The login page can reside on one flash device, while the success and failure pages can be stored on another flash device

- You must configure all four pages.
- All logo files—including image, flash, audio, video, and similar file types—stored in the system directory must use *web_auth_ <filename>* as the file name. System directory examples are flash, disk0, or disk.
- You can copy the configured web pages to the switch boot flash or the flash.

Restrictions for customizing web authentication pages

- The banner page has no effect if it is configured together with a web authentication page.
- When the configured login form is enabled, the CLI command for redirecting users to a specific URL is unavailable. Configure redirection in the web page.
- If you enter the CLI command to redirect users to a specific URL after authentication and then configure web pages, the redirect command does not take effect.
- The configured authentication proxy feature supports both HTTP and SSL.

Guidelines for configuring a redirection URL for Successful Login page

Redirection URL for Successful Login Guidelines

- If you enable the custom authentication proxy web pages feature, you cannot use the redirection URL feature in the CLI. To redirect users after login, configure redirection in the custom login success page.
- If you enable the redirection URL feature, the configured authorization proxy banner is not be used.
- To remove the specification of a redirection URL, use the **no** form of the command.
- If a redirection URL is required after successful authentication, it must begin with a valid protocol prefix (such as http://) followed by the URL. If http:// is omitted, the browser might show a page not found error or similar issue

How to configure local web authentication

-

Configure default local web authentication

Table 112: Default local web authentication configuration

Feature	Default Setting
AAA	Disabled
RADIUS server • IP address • UDP authentication port • Key	• None specified
Default value of inactivity timeout	3600 seconds
Inactivity timeout	Enabled

Configure AAA settings using the wizard

Use the AAA Wizard to streamline the addition of authentication, authorization, and accounting (AAA) servers and settings through a guided interface.

The AAA wizard helps you add the authentication, authorization and accounting details without going into multiple screens.

The AAA Wizard guides you through configuring RADIUS, TACACS+, and LDAP servers, and mapping AAA server groups and methods. If you need to edit details entered with the wizard, use the appropriate configuration screens.

> **Note**
>
> Configure attribute formats as listed in the *Supported authentication and authorization attribute formats* section to ensure successful transactions.

Before you begin

Ensure you have network information and credentials available for the AAA servers to be configured.

Procedure

1. Choose **Configuration > Security** > **AAA**.
2. Click **+ AAA Wizard**.

   The **Add Wizard** window is displayed.
3. Go to the **RADIUS** tab.

   The RADIUS server option is enabled by default. You can switch between **Basic** and **Advanced** options using the radio buttons.

   a) In the **Name** field, enter the name of the RADIUS server.
   b) In the **IPv4 / IPv6 Server Address** field, enter the IPv4 or IPv6 address, or hostname of the server.
   c) Select the **PAC Key** check box to enable Protected Access Credential (PAC) authentication key option.
   d) From the **Key Type** drop-down list, select the authentication key type.
   e) In the **Key** field, enter the authentication key.
   f) In the **Confirm Key** field, re-enter the authentication key.
   g) Click **Advanced** radio button.

      This enables the advanced options.

   h) In the **Auth Port** field, enter the authorization port number.
   i) In the **Acct Port** field, enter the accounting port number.
   j) In the **Server Timeout** field, enter the timeout duration, in seconds.
   k) In the **Retry Count** field, enter the number of retries.
   l) In the **Support for CoA** field, use the toggle button to enable or disable change-of-authorization (CoA).
4. To enter TACACS+ options, check the **TACACS+** check box.

   This enables the TACACS+ options. You can switch between **Basic** and **Advanced** options using the radio buttons.

   a) In the **Name** field, enter the TACACS+ server name.
   b) In the **IPv4 / IPv6 Server Address** field, enter the IPv4 or IPv6 address, or hostname.
   c) In the **Key** field, enter the authentication key.
   d) In the **Confirm Key** field, re-enter the authentication key.
   e) Click **Advanced** radio button.

This enables the advanced options.

f) In the **Port** field, enter the port number to use.
g) In the **Server Timeout** field, enter the timeout duration, in seconds.

5. To enter LDAP options, check the **LDAP** check box.

   This enables the LDAP options. You can switch between **Basic** and **Advanced** options using the radio buttons.

   a) In the **Server Name** field, enter the **LDAP** server name.
   b) In the **IPv4 / IPv6 Server Address** field, enter the IPv4 or IPv6 address, or hostname.
   c) In the **Port Number** field, enter the port number to use.
   d) From the **Simple Bind** drop-down list, select the authentication key type.
   e) In the **User Base DN** field, enter the user base distinguished name.
   f) Click **Advanced** radio button.

      This enables the advanced options.

   g) From the **User Attribute** drop-down list, select the user attribute.
   h) In the **User Object Type** field, enter the details and click the + icon.

      The objects you add appear in the list, and you can use the x mark beside each object to remove it.

   i) In the **Server Timeout** field, enter the timeout duration, in seconds.
   j) Check the **Secure Mode** check box to enable secure mode.

      Selecting the **Secure Mode** option enables the **Trustpoint Name** drop-down list.

   k) From the **Trustpoint Name** drop-down list, select the trustpoint.
   l) Click **Next**.

      This enables the **Server Group Association** screen and RADIUS tab is selected by default.

6. Go to the **RADIUS** tab.
   a) In the **Name** field, enter the name of the RADIUS server group.
   b) From the **MAC-Delimiter** drop-down list, choose the delimiter to be used in the MAC addresses that are sent to the RADIUS servers.
   c) From the **MAC Filtering** drop-down list, choose a value based on which to filter MAC addresses.
   d) To configure dead time for the server group and direct AAA traffic to alternative groups of servers that have different operational characteristics, in the **Dead-Time** field, enter the amount of time, in minutes, after which a server is assumed to be dead.
   e) Choose the servers that you want to include in the server group from the **Available Servers** list and move them to the **Assigned Servers** list.
   f) Click **Next**.

      The **TACACS+** window is displayed, if you have selected **TACACS+** in Server configuration.

7. Use the **TACACS+** window to enter details.
   a) In the **Name** field, enter the name of the TACACS+ server group.
   b) Choose the servers that you want to include in the server group from the **Available Servers** list and move them to the **Assigned Servers** list.
   c) Click **Next**.

      The **LDAP** window is displayed, if you have selected **LDAP** in server configuration.

8. Use the **LDAP** window to enter details.
   a) In the **Name** field, enter the name of the LDAP server group.
   b) Choose the servers that you want to include in the server group from the **Available Servers** list and move them to the **Assigned Servers** list.

9. Click **Next**.

   The **MAP AAA** window is displayed.

   You can use the check boxes to enable the **Authentication**, **Authorization**, and **Accounting** tabs. You cannot deselect all three options. At least one option has to be selected.

10. Use the **Authentication** tab to enter authentication details.
    a) In the **Method List Name** field, enter the name of the method list.
    b) From the **Type** drop-down list, choose the type of accounting you want to perform before allowing access to the network.
    c) From the **Group Type** drop-down list, choose if you want to assign a group of servers as your access server, or if you want to use a local server to authenticate access.

       If you choose local option, **Fallback** to local option is removed.

    d) Check the **Fallback to local** check box to configure a local server to act as a fallback method when servers in the group are unavailable.
    e) From the **Available Server Groups** list, choose the server groups you want to use to authenticate access to your network and click **>** icon to move them to the **Assigned Server Groups** list.
11. Check the **Authorization** check box to configure authorization details.
    a) In the **Method List Name** field, enter the name of the method list.
    b) From the **Type** drop-down list, choose the type of authorization you want to perform before allowing access to the network.
    c) From the **Group Type** drop-down list, choose if you want to assign a group of servers as your access server, or if you want to use a local server to authorize access.

       If you choose local option, **Fallback** to local option is removed.

    d) Check the **Fallback to local** check box to configure a local server to act as a fallback method when servers in the group are unavailable.
    e) From the **Available Server Groups** list, choose the server groups you want to use to authorize access to your network and click **>** icon to move them to the **Assigned Server Groups** list.
12. Check the **Accounting** check box to configure accounting details.
    a) In the **Method List Name** field, enter the name of the method list.
    b) From the **Type** drop-down list, choose the type of accounting you want to perform.
    c) From the **Available Server Groups** list, choose the server groups you want to use to authorize access to your network and click **>** icon to move them to the **Assigned Server Groups** list.
13. Click **Apply to Device**.

The system saves your AAA configuration, associating servers and methods as specified. AAA features are ready for use.

Configure AAA authentication (GUI)

Set up AAA authentication to control network access using the GUI.

Use this task to define and assign authentication methods and server groups for device access.

Before you begin

Confirm that server groups are configured if you plan to use them.

Procedure

---

1. Choose **Configuration** > **Security** > **AAA**.
2. In the **Authentication** section, click **Add**.

3. In the **Quick Setup: AAA Authentication** window that is displayed, enter a name for your method list.
4. Choose the type of authentication you want to perform before allowing access to the network, in the **Type** drop-down list.
5. In the **Group** Type drop-down list, select either a group of servers as your access server or a local server to authenticate access.
6. To configure a local server to act as a fallback method when servers in the group are unavailable, check the **Fallback** to local check box.
7. Choose the server groups you want to use to authenticate access to your network, from the **Available Server Groups** list and click **>** icon to move them to the **Assigned Server Groups** list.
8. Click **Save & Apply to Device**.

The controller is now configured with AAA authentication, and users are authenticated using the chosen methods.

Configure AAA authentication (CLI)

Enable authentication, authorization, and accounting (AAA) on the device to manage login methods and network access.

AAA centralizes user authentication for device access and network services. Select a named method list or the default list, depending on your VTY line configuration.

If no method list is configured under VTY lines, add the default method list to the AAA configuration."

```
line vty 0 4
 authorization commands 15 abc
aaa authorization commands 15 abc group tacacs+
```

If a method-list is not configured under VTY lines, you must add the default method list to the AAA configuration:

```
line vty 0 4
 aaa authorization commands 15 default group tacacs+
```



Note

Use the default list for AAA authorization if you plan to use features such as dACL.

Before you begin

Ensure that the TACACS+ server address and group name are available.

Procedure

---

1. Enable AAA functionality.

   Example

   ```
   Device
   (config)# aaa new-model
   ```

2. Define the list of authentication methods at login using the **aaa authentication login {default** | *named_authentication_list*} **group** *AAA_group_name* command.

Example

```
Device(config)# aaa authentication login default group AAA_group_name
```

named_authentication_list refers to any name that is not greater than 31 characters.

AAA_group_name refers to the server group name. Define the **server_name** at the beginning of the configuration.

3. Create an authorization method list for web-based authorization using the **aaa authorization network {default** | *named_authorization_list*} **group** *AAA_group_name* command.

Example

```
Device(config)# aaa authorization network default group group1
```

4. Specify an AAA server.

Example

```
Device(config)# tacacs server yourserver
```

5. Configure the IP address for the TACACS server.

Example

```
Device(config-server-tacacs)# address ipv4 10.0.1.12
```

6. Specify a AAA server.

Example

```
Device(config)# tacacs-server host 10.1.1.1
```

Configure the HTTP or HTTPS server (GUI)

Set up HTTP and HTTPS access to enable secure web-based management of the device.

HTTP provides basic web access; HTTPS secures connections with SSL encryption. You can configure access ports, authentication, trust points, and session policies to meet your security requirements.

Before you begin

Confirm necessary certificates are available if using HTTPS trust points.

Procedure

---

1. Choose **Administration** > **Management** > **HTTP/HTTPS/Netconf**.
2. In the **HTTP/HTTPS Access Configuration** section, enable HTTP Access and enter the port that will listen for HTTP requests. The default port is 80. Valid values are 80, and ports between 1025 and 65535.

3. Enable **HTTPS Access** on the device and enter the designated port to listen for HTTPS requests. The default port is 1025. Valid values are 443 and ports between 1025 and 65535. Data exchanged over a secure HTTP connection is encrypted before transmission across the Internet. SSL encryption over HTTP enables secure configuration of the switch through a web browser.
4. Choose the **Personal Identity Verification** as enabled or disabled.
5. In the **HTTP Trust Point Configuration** section, enable **Enable Trust Point** to use Certificate Authority servers as trustpoints.
6. From the **Trust Points** drop-down list, choose a trust point.
7. In the **Timeout Policy Configuration** section, enter the HTTP timeout policy in seconds. Enter a value between one to 600 seconds.
8. Enter the number of minutes of inactivity allowed before the session times out. Enter a value between 180 and 1200 seconds.
9. Enter the server life time in seconds. Enter a value between one and 86400 seconds.
10. Enter the maximum number of requests the device can accept. Valid values range from one and 86400 requests.
11. Save the configuration.

The device is configured for HTTP or HTTPS access according to your specified settings.

Configure the HTTP server (CLI)

Enable HTTP or HTTPS server functionality on your device to support local web authentication.

Local web authentication requires the HTTP server to be enabled on the controller.

You can configure both HTTP and HTTPS servers to support device management and user authentication.

Both HTTP and HTTPS servers can be configured to support device management and user authentication.

Note that some browsers, such as the Apple psuedo-browser, do not open if you configure only the **ip http secure-server** command. You should also configure the **ip http server** command.

Procedure

---

1. Enable the HTTP server. The local web authentication feature uses the HTTP server to communicate with the hosts for user authentication.

   Example

   ```
   Device(config)# ip http server
   ```

2. Enable HTTPS.

   Example

   ```
   Device(config)# ip http secure-server
   ```

   You can configure custom authentication proxy web pages or specify a redirection URL for successful login.

> **Note**
>
> To ensure secure authentication when you enter the **ip http secure-server** command, the login page is always in HTTPS (secure HTTP) even if the user sends an HTTP request.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

The controller is now configured with the HTTP or HTTPS or both. This enables local web authentication and secure browser access for users.

Allow special characters for serial port

Allow Special Characters for Serial Port using commands.

Configure the serial port to support special characters for advanced device communication.

Use this procedure when the connected device or application must process special characters on the serial console port.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the primary terminal line number.

Example

```
Device(config)# line console 0
```

3. Configure the time to disconnect idle EXEC sessions using the **exec-timeout** *mins sec* command.

Example

```
Device(config-line)# exec-timeout 12 0
```

4. Configure login authentication checking using the **login authentication** *word* **default** command.

Example

```
Device(config-line)# login authentication NO_LOGIN
```

You can configure an authentication list with a name or use the default authentication list

5. Configure the character widths of EXEC command characters using the **exec-character-bit 7 8** command.

Example

```
Device(config-line)# exec-character-bit 8
```

6. Configure the stop bits for the console port using the **stopbits 1 1.5 2** command.

Example

```
Device(config-line)# stopbits 1
```

7. Return to privileged EXEC mode.

Example

```
Device(config-line)# end
```

The serial port is now configured to support special characters according to your settings.

Allow special characters for VTY port

Allow Special Characters for VTY Port using commands.

Enable the use of special (non-standard) characters in the banner text for Virtual Teletype (VTY) ports, which enhances customization options and supports global character sets.

You can allow special (non-standard) characters in the banner text for Virtual Teletype (VTY) ports. This lets you customize banners and support global character sets.

You may want to allow special characters in the banner displayed on the VTY port—for example, to include accented letters, non-English text, or symbols for branding or user communication.

Before you begin

Ensure you have determined the desired banner text and any required special characters.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Create a parameter map and enter parameter-map webauth configuration mode.

Example

```
Device(config)# parameter-map type webauth global
```

3. Create a custom banner using the **banner text** *text* command.

Example

```
Device(config-params-parameter-map)# banner text #Hêllö#
```

You can create a custom banner (of up to 400 characters) by entering *c <banner-text> c*, where *c* is a delimiting character.

If the banner text exceeds 400 characters (400 characters is about 250 words), you see an error message and the configuration is rejected. The parser limits each line to 254 characters (about 150 words), including CLI keywords. If you need more than 254 characters, split the banner text into multiple lines.

The webauth login page displays only the default banner strings, if banner command is not configured.

4. Return to privileged EXEC mode.

Example

```
Device(config-params-parameter-map)# end
```

You can now use special characters in the VTY port banner. When users connect through VTY, they see your custom banner.

HTTP and HTTPS access modes for web authentication

The table describes the various CLI combinations.

You can control HTTP and HTTPS access separately for device management (admin) and web authentication by configuring web authentication using CLI command combinations. The main commands involved are:

- **ip http server** : Enables HTTP access to the device for administration.
- **ip http secure-server** : Enables HTTPS access to the device for administration and is required for HTTPS access to web authentication.
- **webauth-http-enable** (under **parameter-map type webauth global)** : Allows HTTP access for web authentication without enabling HTTP access for administration. This command is not enabled by default.
- **secure-webauth-disable** (under **parameter-map type webauth global**): Disables HTTPS access for web authentication only. This command is not enabled by default.

CLI combinations

The table describes the possible CLI combinations.

Admin (Device Management)		Web Authentication		Required Configurations	
HTTP Access	**HTTPS Access**	**HTTP Access**	**HTTPS Access**	**Admin**	**Web Authentication**

No	Yes	Yes	Yes	`no ip http server`
				`ip http secure-server`
					`no ip http server`
					`ip http secure-server`
					`parameter-map type webauth global`
					`webauth-http-enable`

No	Yes	No	Yes	`no ip http server` `ip http secure-server`	`no ip http server` `ip http secure-server`

No	Yes	Yes	No	`no ip http server`
				`ip http secure-server`
					`no ip http server`
					`ip http secure-server`
					`parameter-map type webauth global`
					`webauth-http-enable`
					`secure-webauth-disable`

No	Yes	No	No	`no ip http server`
				`ip http secure-server`
					`no ip http server`
					`ip http secure-server`
					`parameter-map type webauth global`
					`secure-webauth-disable`

No	No	No	Yes	`no ip http server` `no ip http secure-server`	Not supported
No	No	Yes	No		

Admin (Device Management)		Web Authentication		Required Configurations	
HTTP Access	**HTTPS Access**	**HTTP Access**	**HTTPS Access**	**Admin**	**Web Authentication**

				`no ip http server`
				`no ip http secure-server`
					`no ip http server`
					`no ip http secure-server`
					`parameter-map type webauth global`
					`webauth-http-enable`

Yes	No	Yes	No	`ip http server` `no ip http secure-server`	`ip http server` `no ip http secure-server`

Yes	Yes	Yes	No	`ip http server`
				`ip http secure-server`
					`ip http server`
					`ip http secure-server`
					`parameter-map type webauth global`
					`secure-webauth-disable`

- To enable HTTP access for web authentication without also enabling HTTP administrative access, configure **webauth-http-enable** in **parameter-map type webauth global** mode. Do not use **ip http server** .
- HTTPS access for web authentication always requires **ip http secure-server** .
- To disable HTTPS access for web authentication, use the **secure-webauth-disable** command.
- These combinations allow granular control over how users can access the device for both management and web authentication purposes.

Limitations

These guidelines and limitations help you configure HTTP and HTTPS requests for web authentication.

- Enable HTTPS for device management before you enable HTTPS web authentication
- If **secure-webauth-disable** is configured and the client's initial request is `https://< >`, central web authentication cannot be performed.

Configure HTTP and HTTPS requests for web authentication (CLI)

Configure HTTP and HTTPS server parameters to support secure web authentication.

Perform this configuration when you need to enable or disable HTTP or HTTPS web authentication features on your device.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Set the HTTP server to default.

   Example

   ```
   Device(config)# no ip http server
   ```

4. Enable the HTTP server or the HTTP secure server using the **ip http {server | secure-server}** command.

   Example

   ```
   Device(config)# ip http server
   ```

5. Enable the global parameter map mode.

   Example

   ```
   Device(config)# parameter-map type webauth global
   ```

6. Disable HTTP secure server for web authentication.

   Example

   ```
   Device(config-params-parameter-map)# secure-webauth-disable
   ```

7. Enable HTTP server for web authentication.

   Example

   ```
   Device(config-params-parameter-map)# webauth-http-enable
   ```

The device is configured to send HTTP requests, HTTPS requests, or both to the web authentication module according to your settings.

Create Parameter Maps

Create a parameter map (GUI)

Define criteria-based policies to control device and user access within the local policy framework.

Use this task to create a parameter map by specifying match criteria and associating service templates. This enables dynamic policy application based on device type, user role, and other attributes.

Before you begin

Ensure you have the required policy details and service templates ready.

Procedure

1. Choose **Configuration** > **Security** > **Local Policy**.
2. Click **Add**.
3. Click **Policy Map**.
4. Enter **Policy Map Name**.
5. In the **Match Criteria List** settings, click **Add**.
6. In the **Add Match Criteria** settings, choose the service template from the **Service Template** drop-down list.
7. Choose the filters from **Device Type**, **User Role**, **User Name**, **OUI**, and **MAC Address** drop-down lists.
8. Click **Add Criteria**.
9. Click **Apply to Device**.

The parameter map is created and applied, enabling policy enforcement based on the selected criteria.

Configure local web authentication (GUI)

Enable local web authentication for your network using GUI.

Use this task to set authentication parameters, banner messages, and related settings for web authentication.

Procedure

1. Choose **Configuration** > **Security** > **Web Auth**.
2. On the **Web Auth** page, click **Add**.
3. In the **Create Web Auth Parameter** window that is displayed, enter a name for the parameter map.
4. In the **Maximum HTTP Connections** field, enter the maximum number of HTTP connections that you want to allow.
5. In the **Init-State Timeout** field, enter the timeout duration. The timer expires if the user does not enter valid credentials on the login page
6. Choose the type of Web Auth parameter.
7. Click **Apply to Device**.
8. On the **Web Auth** page, click the name of the parameter map.
9. In the **Edit WebAuth Parameter** window that is displayed, choose the required **Banner Type**.
   - If you choose **Banner Text**, enter the required banner text to be displayed.
   - If you choose **File Name**, specify the path of the file from which the banner text has to be picked up.
10. Enter the virtual IP addresses as required.
11. Set the status of **WebAuth Intercept HTTPS**, **Captive Bypass Portal**, and **Watch List Enable** as required.
12. In the **Watch List Expiry Timeout** field, enter the time in seconds after which the watch list should time out.

13. Set the status for **Disable Success Window**, **Disable Logout Window**, and **Login Auth Bypass for FQDN**, as required.
14. Check the **Sleeping Client Status** check box to enable authentication of sleeping clients. Specify the **Sleeping Client Timeout** in minutes. The valid range is 10 to 43,200 minutes.
15. Click the **Advanced** tab.
16. In the **Redirect for log-in** field, enter the name of the external server to which to send the login request.
17. In the **Redirect On-Success** field, enter the name of the external server to redirect after a successful login.
18. In the **Redirect On-Failure** field, enter the name of the external server to redirect after a login failure.
19. To configure external local web authentication, perform these steps:
    a) Under **Redirect to External Server** in the **Redirect Append for AP MAC Address** field, enter the AP MAC address.
    b) In the **Redirect Append for Client MAC Address** field, enter the client MAC address.
    c) In the **Redirect Append for WLAN SSID** field, enter the WLAN SSID.
    d) In the **Portal IPV4 Address** field, enter the IPv4 address of the portal to send redirects.
    e) In the **Portal IPV6 Address** field, enter the IPv6 address of the portal to send redirects if you are using an IPv6 address.
20. To configure customized local web authentication, perform these steps:
    a) Under **Customized Page**, specify the following pages:
        - **Login Failed Page**
        - **Login Page**
        - **Logout Page**
        - **Login Successful Page**
21. Click **Update & Apply**.

Local web authentication is enabled with your specified settings.

Configure internal local web authentication (CLI)

Set up internal local web authentication on your device using CLI.

Procedure

In the global configuration mode, create a parameter map

Example

```
Device(config)# parameter-map type webauth sample
```

Creates the parameter map.

The name of the parameter map must not exceed 99 characters.

You have configured internal local web authentication.

The device prompts clients to authenticate through the web portal. After authentication, the device grants full network access.

Configure the customized local web authentication (CLI)

Configure custom local web authentication workflow on a Cisco device by specifying custom portal pages using CLI.

Perform this procedure to offer users a personalized login, success, and failure experience during web authentication.

Before you begin

- Ensure you have valid HTML files for login, success, expired, and failure pages uploaded to device storage.
- Configure a virtual IP in the global parameter map to use the customized web authentication bundle.

Procedure

1. Configure the webauth type parameter.

   Example

   ```
   Device(config)# parameter-map type webauth parameter-map-name
   ```

   Configure a virtual IP in the global parameter map so the device uses the customized web authentication bundle.

2. Configures webauth sub-types using the **type** {**authbypass** | **consent** | **webauth** | **webconsent**} command.

   Example

   ```
   Device(config-params-parameter-map)# type webauth
   ```

   The types are **passthru**, **consent**, **webauth**, or **webconsent**.

3. Configure the customized login page using the **custom-page login device** *html-filename* command.

   Example

   ```
   Device(config-params-parameter-map)# custom-page login device
   bootflash:login.html
   ```

4. Configure the customized login expiry page using the **custom-page login expired device** *html-filename* command.

   Example

   ```
   Device(config-params-parameter-map)# custom-page login expired device
   bootflash:loginexpired.html
   ```

5. Configure the customized login success page using the **custom-page success device** *html-filename* command.

   Example

   ```
   Device(config-params-parameter-map)# custom-page success device
   bootflash:loginsuccess.html
   ```

6. Configure the customized login failure page using the **custom-page failure device** *html-filename* command.

Example

```
Device(config-params-parameter-map)# custom-page failure device
bootflash:loginfail.html
```

You have configured the customized local web authentication portal. Users can see the specified custom pages during authentication.

Configure the external local web authentication (CLI)

Configure the External Local Web Authentication using commands.

Enable external local web authentication on your device by configuring the required CLI parameters.

Use this procedure to set up external local web authentication settings, including parameter maps and redirect URLs, using the CLI.

Before you begin

Gather the parameter map name and redirect URLs.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the webauth type parameter.

   Example

   ```
   Device(config)# parameter-map type webauth sample
   ```

3. Configure web authentication subtypes by using the **type** {**authbypass** | **consent** | **webauth** | **webconsent**} command.

   Example

   ```
   Device(config-params-parameter-map)# type webauth
   ```

4. Configure the redirect URL for the login, failure, and success pages using the **redirect** [**for-login** | **on-failure** | **on-success**] *URL* command.

   Example

   ```
   Device(config-params-parameter-map)# redirect for-login
   http://www.cisco.com/login.html
   ```

> **Note**
>
> In the redirect URL, you need to press *Ctrl+v* and type *?* to configure the *?* character.
>
> The *?* character is commonly used in URL when ISE is configured as an external portal.

5. Configure the external portal IPv4 address using the **redirect portal** {**ipv4** | **ipv6**} *ip-address* command.

   Example

   ```
   Device(config-params-parameter-map)# redirect portal ipv4 23.0.0.1
   ```

   When using FQDN, use an IP address associated with the domain. If the domain resolves to more than one IP address, use the FQDN URL instead of an individual IP address.

6. Return to privileged EXEC mode.

   Example

   ```
   Device(config-params-parameter-map)# end
   ```

You have configured external local web authentication. Your device redirects users to the correct portal for authentication events, based on your parameter map and settings.

Configure the web authentication WLANs

Configure the Web Authentication WLANs using commands.

Configure a WLAN with web authentication security using CLI. Map authentication lists and parameter maps using the CLI.

Perform this task to set up a WLAN that uses web authentication, specifying key identifiers and settings through the CLI on your network device.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration sub-mode.

Example

```
Device(config)# wlan wlan-name wlan-id SSID-name
```

3. Disable WPA security.

   Example

   ```
   Device(config-wlan)# no security wpa
   ```

4. Enable web authentication for WLAN.

   Example

   ```
   Device(config-wlan)# security web-auth
   ```

5. Enable web authentication for WLAN using the **security web-auth** {**authentication-list** *authentication-list-name* | **parameter-map** *parameter-map-name*} command.

   Example

   ```
   Device(config-wlan)# security web-auth authentication-list webauthlistlocal
   ```

   ```
   Device(config-wlan)# security web-auth parameter-map sample
   ```

   Here is a description of some of the parameters.

   - **authentication-list** *authentication-list-name*: Sets the authentication list for IEEE 802.1x.
   - **parameter-map** *parameter-map-name*: Configures the parameter map.

   > **Note**
   >
   > When **security web-auth** is enabled, you can map the default **authentication-list** and global **parameter-map**. This is applicable for authentication-list and parameter-map that are not explicitly mentioned.

6. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wlan)# end
   ```

The WLAN is configured with web authentication security and the desired authentication and parameter settings.

Configure pre-auth web authentication ACL (GUI)

Configure Pre-Auth Web Authentication ACL using GUI.

Map a pre-authentication access control list (ACL) to a WLAN to control network access for users before authentication.

Use this procedure to enforce security policies by restricting network access to specified resources while users are in the pre-authenticated state on a wireless LAN.

Before you begin

Ensure that you have configured an ACL and a WLAN.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click the name of the WLAN.
3. In the **Edit WLAN** window, click the **Security** tab and then click the **Layer3** tab.
4. Click **Show Advanced Settings**.
5. In the **Preauthenticaion ACL** section, choose the appropriate ACL to map to the WLAN.
6. Click **Update & Apply to Device**.

When you connect to this WLAN, you must adhere to the access restrictions defined by the ACL until authentication is complete.

Configure pre-authentication web authentication ACL (CLI)

Configure Pre-Auth Web Authentication ACL using commands.

Set up a pre-authentication web authentication access control list (ACL) and apply it to a WLAN using CLI commands. This configuration ensures only authorized traffic passes through before web authentication occurs.

Use this task to control the network traffic permitted before clients authenticate on a WLAN. The ACL defines which clients and protocols are permitted through during the web authentication process.

Before you begin

- Gather the required network and client information.
- Determine the ACL rules (permit or deny) and relevant addresses.

Procedure

1. Enters global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create an ACL list using the **access-list** *access-list-number* {**deny** | **permit**} *hostname source-wildcard-bits* command.

Example

```
Device(config)# access-list 2 deny your_host 10.1.1.1 log
```

The *access-list-number* is a decimal number from one to 99, 100 to 199, 300 to 399, 600 to 699, 1300 to 1999, 2000 to 2699, or 2700 to 2799.

Enter **deny** or **permit** to specify whether to deny or permit if the conditions are matched.

The *source* is the source address of the network or host from which the packet is being sent specified as:

- The 32-bit quantity in dotted-decimal format.
- The keyword any as an abbreviation for *source* and *source-wildcard* of 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard.
- The keyword host as an abbreviation for *source* and *source-wildcard* of source 0.0.0.0.

(Optional) The *source-wildcard* applies wildcard bits to the source.

> **Note**
>
> The AP forwards traffic if it matches any of the ACLs applied in the pre-authentication ACLs. Use bidirectional rules for clients in web authentication pending state and for downstream traffic.

3. Create the WLAN.

Example

```
Device(config)# wlan mywlan 34 mywlan-ssid
```

profile-name is the WLAN name which can contain up to 32 alphanumeric characters.

wlan-id is the wireless LAN identifier. The valid range is from one to 512.

ssid-name is the SSID which can contain 32 alphanumeric characters.

4. Map the ACL to the web auth WLAN.

Example

```
Device(config-wlan)# ip access-group web name
```

access-list-name is the IPv4 ACL name or ID.

5. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

The ACL is configured and mapped to the WLAN. Only traffic that matches the permitted criteria passes before web authentication.

Configure the maximum web authentication request retries

Configure the Maximum Web Authentication Request Retries.

Set the maximum number of web authentication request retries to define how many times the system attempts authentication before it stops.

Configure this setting to adjust the tolerance for failed web authentication attempts, which can improve network security and user experience.

Before you begin

Ensure you are in global configuration mode on your device.

Procedure

1. Configure the maximum web authentication request retries.

   Example

   ```
   Device(config)# wireless security web-auth retries  number
   ```

   number is the maximum number of web authentication request retries. The valid range is 0-20.

2. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

You have configured the specified maximum number of web authentication request retries.

Configure a local banner in web authentication page (GUI)

Configure a Local Banner in Web Authentication Page using GUI.

Present a custom banner to users on the web authentication page to meet organizational messaging or compliance requirements.

Use this task to configure a banner that will display on the login page for web authentication sessions.

Before you begin

Procedure

1. Choose **Configuration** > **Security** > **Web Auth**.
2. In the **Webauth Parameter Map** tab, click the parameter map name. The **Edit WebAuth Parameter** window is displayed.
3. In the **General** tab and choose the required Banner Type:
   - If you choose **Banner Text**, enter the required banner text to be displayed.
   - If you choose **File Name**, specify the path of the file from which the banner text has to be picked up.
4. Click **Update & Apply**.

The specified banner displays on the web authentication page as configured.

Configure a local banner in web authentication page (CLI)

Configure a Local Banner in Web Authentication Page using commands.

Present a custom banner to users on the Web Authentication page to meet organizational messaging or compliance requirements.

Use this task to configure a banner that displays on the login page for web authentication sessions.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the web authentication parameters. Enters the parameter map configuration mode.

   Example

   ```
   Device(config)# parameter-map type webauth param-map
   ```

3. Enable the local banner using the **banner** *file* | *banner-text title* command.

   Example

   ```
   Device(config-params-parameter-map)# banner http C My Switch C
   ```

   Create a custom banner by entering *C banner-text C* (where *C* is a delimiting character), or *file* that indicates a file (for example, a logo or text file) that appears in the banner, or *title* that indicates the title of the banner.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-params-parameter-map)# end
   ```

The specified banner displays on the web authentication page as configured.

Configure Type WebAuth, Consent, or both

Configure Type WebAuth, Consent, or Both.

Configure a parameter map to use webauth type, consent, or webconsent login types.

Use this configuration to enable flexible guest access authentication for users on supported devices.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the Webauth type parameter.

   Example

   ```
   Device(config)# parameter-map type webauth webparalocal
   ```

3. Configure webauth type to Consent. You can configure the type as Webauth, Consent, or both (Webconsent).

   Example

   ```
   Device(config-params-parameter-map)# type consent
   ```

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-params-parameter-map)# end
   ```

5. Display the configuration details.

   Example

   ```
   Device(config)#show running-config | section parameter-map type webauth test
   ```

The parameter map is configured to use Webauth, Consent, or Webconsent types as specified and is ready for client authentication.

Configure preauthentication ACL

Configure Preauthentication ACL using commands.

Set up a preauthentication access control list (ACL) to restrict network access prior to client authentication.

Use this task when you need to limit which resources clients can reach before they have authenticated on a WLAN.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN you want to configure.

   Example

   ```
   Device(config)# wlan ramban
   ```

3. Disable the WLAN.

   Example

   ```
   Device(config-wlan)# shutdown
   ```

4. Configure the ACL to apply before authentication.

   Example

   ```
   Device(config-wlan)# ip access-group web preauthrule
   ```

5. Enable the WLAN again.

   Example

   ```
   Device(config)# no shutdown
   ```

6. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wlan)# end
   ```

7. Display the configuration details for the WLAN

   Example

   ```
   Device# show wlan name wlan-name
   ```

The preauthentication ACL is now active on the WLAN, restricting client access prior to authentication.

Configure trustpoint for local web authentication

Configure TrustPoint for Local Web Authentication using commands.

Configure a trustpoint on your controller to support secure local web authentication.

Set up a trustpoint so the controller can present a domain-specific certificate. Client browsers recognize and accept this certificate during redirection to the portal (for example, example.com).

Before you begin

Ensure that a certificate is installed on your controller.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create the parameter map.

   Example

   ```
   Device(config)# parameter-map type webauth global
   ```

3. Configure trustpoint for local web authentication.

   Example

   ```
   Device(config-params-parameter-map)# trustpoint trustpoint-name
   ```

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-params-parameter-map)# end
   ```

During local web authentication, the controller presents the specified domain-specific certificate. Clients' browsers then recognize and accept the portal redirection.

Configuration examples for local web authentication

-

Example: obtain a web authentication certificate

Obtaining Web Authentication Certificate example.

This example shows how to obtain web authentication certificate.

```
DeviceDevice# configure terminal
Device(config)# crypto pki import cert pkcs12
tftp://10.1.0.100/ldapserver-cert.p12 cisco
Device(config)# end
Device# show crypto pki trustpoints cert
 Trustpoint cert:
    Subject Name:
```

```
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
          Serial Number (hex): 00
    Certificate configured.
Device# show  crypto pki certificates cert
Certificate
  Status: Available
  Certificate Serial Number (hex): 04
  Certificate Usage: General Purpose
  Issuer:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
  Subject:
    Name: ldapserver
    e=rkannajr@cisco.com
    cn=ldapserver
    ou=WNBU
    o=Cisco
    st=California
    c=US
  Validity Date:
    start date: 07:35:23 UTC Jan 31 2012
    end   date: 07:35:23 UTC Jan 28 2022
  Associated Trustpoints: cert ldap12
  Storage: nvram:rkannajrcisc#4.cer

CA Certificate
  Status: Available
  Certificate Serial Number (hex): 00
  Certificate Usage: General Purpose
  Issuer:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
  Subject:
    e=rkannajr@cisco.com
    cn=sthaliya-lnx
    ou=WNBU
    o=Cisco
    l=SanJose
    st=California
    c=US
  Validity Date:
    start date: 07:27:56 UTC Jan 31 2012
    end   date: 07:27:56 UTC Jan 28 2022
  Associated Trustpoints: cert ldap12 ldap
  Storage: nvram:rkannajrcisc#0CA.cer
```

Example: display a web authentication certificate

Display a Web Authentication Certificate example.

This example shows how to display a web authentication certificate.

```
Device# show crypto ca certificate verb
     Certificate
     Status: Available
     Version: 3
     Certificate Serial Number (hex): 2A9636AC00000000858B
     Certificate Usage: General Purpose
     Issuer:
    cn=Cisco Manufacturing CA
    o=Cisco Systems
    Subject:
    Name: WS-C3780-6DS-S-2037064C0E80
    Serial Number: PID:WS-C3780-6DS-S SN:FOC1534X12Q
    cn=WS-C3780-6DS-S-2037064C0E80
    serialNumber=PID:WS-C3780-6DS-S SN:FOC1534X12Q
    CRL Distribution Points:
    http://www.cisco.com/security/pki/crl/cmca.crl
    Validity Date:
    start date: 15:43:22 UTC Aug 21 2011
    end   date: 15:53:22 UTC Aug 21 2021
    Subject Key Info:
    Public Key Algorithm: rsaEncryption
    RSA Public Key: (1024 bit)
    Signature Algorithm: SHA1 with RSA Encryption
    Fingerprint MD5: A310B856 A41565F1 1D9410B5 7284CB21
    Fingerprint SHA1: 04F180F6 CA1A67AF 9D7F561A 2BB397A1 0F5EB3C9
    X509v3 extensions:
    X509v3 Key Usage: F0000000
      Digital Signature
      Non Repudiation
      Key Encipherment
      Data Encipherment
    X509v3 Subject Key ID: B9EEB123 5A3764B4 5E9C54A7 46E6EECA 02D283F7
    X509v3 Authority Key ID: D0C52226 AB4F4660 ECAE0591 C7DC5AD1 B047F76C
    Authority Info Access:
    Associated Trustpoints: CISCO_IDEVID_SUDI
    Key Label: CISCO_IDEVID_SUDI
```

Example: choose the default web authentication login page

Choose the Default Web Authentication Login Page example.

This example shows how to choose a default web authentication login page.

```
Device# configure terminal
Device(config)# parameter-map type webauth test
This operation will permanently convert all relevant authentication commands
to their CPL control-policy equivalents. As this conversion is irreversible
and will
disable the conversion CLI 'authentication display [legacy|new-style]', you
are strongly advised to back up your current configuration before proceeding.
Do you wish to continue? [yes]: yes
Device(config)# wlan wlan50
Device(config-wlan)# shutdown
```

```
Device(config-wlan)# security web-auth authentication-list test
Device(config-wlan)# security web-auth parameter-map test
Device(config-wlan)# no shutdown
Device(config-wlan)# end
Device# show running-config | section wlan50
wlan wlan50 50 wlan50
 security wpa akm wpa2
 security wpa wpa1
 security wpa wpa1 ciphers aes
 security wpa wpa1 ciphers tkip
 security web-auth authentication-list test
 security web-auth parameter-map test
 session-timeout 1800
 no shutdown

Device# show running-config | section parameter-map type webauth test
parameter-map type webauth test
 type webauth
```

Example: choosing a CWA login page from an IPv4 external web server

Choose a Customized Web Authentication Login Page from an IPv4 External Web Server example.

This example shows how to choose a customized web authentication login page from an IPv4 external web server.

```
Device# configure terminal
Device(config)# parameter-map type webauth global
Device(config-params-parameter-map)# virtual-ip ipv4 192.0.2.1.
Device(config-params-parameter-map)# parameter-map type webauth test
Device(config-params-parameter-map)# type webauth
Device(config-params-parameter-map)# redirect for-login
http://9.1.0.100/login.html
Device(config-params-parameter-map)# redirect portal ipv4 9.1.0.100
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map
parameter-map type webauth global
virtual-ip ipv4 192.0.2.1.
parameter-map type webauth test
type webauth
redirect for-login http://10.1.0.100/login.html
redirect portal ipv4 10.1.0.100
security web-auth parameter-map rasagna-auth-map
security web-auth parameter-map test
```

Example: choose a customized web authentication login page from an IPv6 external web server

Choose a Customized Web Authentication Login Page from an IPv6 External Web Server example.

This example shows how to choose a customized web authentication login page from an IPv6 external web server.

```
Device# configure terminal
Device(config)# parameter-map type webauth global
Device(config-params-parameter-map)# virtual-ip ipv6 2001:DB8::/48
Device(config-params-parameter-map)# parameter-map type webauth test
Device(config-params-parameter-map)# type webauth
Device(config-params-parameter-map)# redirect for-login
http://9:1:1::100/login.html
```

```
Device(config-params-parameter-map)# redirect portal ipv6 9:1:1::100
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map
parameter-map type webauth global
virtual-ip ipv6 2001:DB8::/48
parameter-map type webauth test
type webauth
redirect for-login http://10:1:1::100/login.html
redirect portal ipv6 10:1:1::100
security web-auth parameter-map rasagna-auth-map
security web-auth parameter-map test
```

Example: assigning login, login failure, and logout pages per WLAN

Assign Login, Login Failure, and Logout Pages per WLAN example.

This example shows how to assign login, login failure and logout pages per WLAN.

```
Device# configure terminal
Device(config)# parameter-map type webauth test
Device(config-params-parameter-map)# custom-page login device
flash:loginsantosh.html
Device(config-params-parameter-map)# custom-page login expired device
flash:loginexpire.html
Device(config-params-parameter-map)# custom-page failure device
flash:loginfail.html
Device(config-params-parameter-map)# custom-page success device
flash:loginsucess.html
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map type webauth test
 parameter-map type webauth test
 type webauth
 redirect for-login http://10.1.0.100/login.html
 redirect portal ipv4 10.1.0.100
 custom-page login device flash:loginsantosh.html
 custom-page success device flash:loginsucess.html
 custom-page failure device flash:loginfail.html
 custom-page login expired device flash:loginexpire.html
```

Example: configure preauthentication ACL

Configuring Preauthentication ACL example.

This example shows how to configure preauthentication ACL.

```
Device# configure terminal
Device(config)# wlan fff
Device(config-wlan)# shutdown
Device(config-wlan)# ip access-group web preauthrule
Device(config-wlan)# no shutdown
Device(config-wlan)# end
Device# show wlan name fff
```

Example: configure webpassthrough

Configure Webpassthrough example.

This example shows how to configure webpassthrough.

```
Device# configure terminal
Device(config)# parameter-map type webauth webparalocal
Device(config-params-parameter-map)# type consent
Device(config-params-parameter-map)# end
Device# show running-config | section parameter-map type webauth test
 parameter-map type webauth test
 type webauth
 redirect for-login http://10.1.0.100/login.html
 redirect portal ipv4 10.1.0.100
```

Verify web authentication type

Verify Web Authentication Type.

To verify the web authentication type, run the command:

```
Device# show parameter-map type webauth all
Type Name
--------------------------------
Global global
Named webauth
Named ext
Named redirect
Named abc
Named glbal
Named ewa-2
```

```
Device# show parameter-map type webauth global
Parameter Map Name : global
Banner:
Text : CisCo
Type : webauth
Auth-proxy Init State time : 120 sec
Webauth max-http connection : 100
Webauth logout-window : Enabled
Webauth success-window : Enabled
Consent Email : Disabled
Sleeping-Client : Enabled
Sleeping-Client timeout : 60 min
Virtual-ipv4 : 10.0.2.1.
Virtual-ipv4 hostname :
Webauth intercept https : Disabled
Webauth Captive Bypass : Disabled
Webauth bypass intercept ACL :
Trustpoint name :
HTTP Port : 80
Watch-list:
Enabled : no
Webauth login-auth-bypass:
```

```
Device# show parameter-map type webauth name global
Parameter Map Name : global
Type : webauth
Auth-proxy Init State time : 120 sec
Webauth max-http connection : 100
Webauth logout-window : Enabled
Webauth success-window : Enabled
```

```
Consent Email : Disabled
Sleeping-Client : Disabled
Webauth login-auth-bypass:
```

External Web Authentication

Explains external web authentication mechanisms for WLAN clients, providing instructions to configure single or multiple WebAuth servers and wired guest EWA using CLI procedures.

Configure EWA with single WebAuth server address and default ports (80/443) (CLI)

The task file describes how to configure External Web Authentication (EWA) with single WebAuth server address and default ports (80/443).

Configure External Web Authentication (EWA) on your device to use a single WebAuth server address with default ports (80 or 443).

Use this procedure to redirect guest WLAN clients to a specific WebAuth portal using default HTTP or HTTPS ports.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Define the authentication method at login.

   Example

   ```
   Device(config)# aaa authentication login WEBAUTH local
   ```

3. Create the parameter map using the **parameter-map type webauth** *parameter-map-name* command.

   Example

   ```
   Device(config)# parameter-map type webauth ISE-Ext-Webauth_IP
   ```

   The *parameter-map-name* must not exceed 99 characters.

4. Configure the webauth type parameter.

   Example

   ```
   Device(config-params-parameter-map)# type webauth
   ```

5. Configure the URL string for redirect during login using the **redirect for-login** *URL-String*command.

   Example

   ```
   Device(config-params-parameter-map)#  redirect for-login
   https://192.168.0.98:443/portal/PortalSetup.action?portal=ad64b062-1098-11e7-8591-005056891b52
   ```

6. Configure the external portal IPv4 address.

   Example

   ```
   Device(config-params-parameter-map)# redirect portal ipv4 192.168.0.98
   ```

7. Return to global configuration mode.

   Example

   ```
   Device(config-params-parameter-map)# exit
   ```

8. Configure a WLAN using the **wlan** *wlan-name wlan-id SSID-name* command.

   Example

   ```
   Device(config)#  wlan EWLC3-GUEST 3 EWLC3-GUEST
   ```

9. Disable adaptive 11r.

   Example

   ```
   Device(config-wlan)# no security ft adaptive
   ```

10. Disable WPA security.

    Example

    ```
    Device(config-wlan)# no security wpa
    ```

11. Disable WPA2 security.

    Example

    ```
    Device(config-wlan)# no security wpa wpa2
    ```

12. Disable WPA2 ciphers for AES.

    Example

    ```
    Device(config-wlan)# no security wpa wpa2 ciphers aes
    ```

13. Disable security AKM for dot1x.

    Example

    ```
    Device(config-wlan)# no security wpa akm dot1x
    ```

14. Enable web authentication for WLAN.

Example

```
Device(config-wlan)# security web-auth
```

15. Enable authentication list for dot1x security using the **security web-auth authentication-list** *authenticate-list-name* command.

Example

```
Device(config-wlan)# security web-auth authentication-list WEBAUTH
```

16. Configure the parameter map using the **security web-auth parameter-map** *parameter-map-name* command.

Example

```
Device(config-wlan)# security web-auth parameter-map ISE-Ext-Webauth_IP
```

> **Note**
>
> If parameter map is not associated with a WLAN, the configuration uses the global parameter map.

17. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

Once enabled for the configured WLAN, web authentication redirects clients to the specified WebAuth portal using default ports.

Configure EWA with multiple web servers and/or ports different than default (80/443)

Configure EWA with Multiple Web Servers and/or Ports Different than Default (80/443) using commands.

Configure an External Web Authentication (EWA) workflow to support multiple web servers and custom port numbers using CLI.

Enable guest access using EWA if there are multiple external web servers or if web authentication must use ports other than the default ports of 80 or 443.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Define an extended IPv4 access list using a name, and enters access-list configuration mode.

Example

```
Device(config)# ip access-list extended preauth_ISE_Ext_WA
```

3. Permit access from any host to the first external web server port number 8443.

Example

```
Device(config)# access-list-number permit tcp any host
external_web_server_ip_address1 eq port-number
```

Example

```
Device(config)# 10 permit tcp any host 192.168.0.98 eq 8443
```

4. Permit access from any host to the second external web server port number 8443.

Example

```
Device(config)# access-list-number permit tcp any host
external_web_server_ip_address2 eq port-number
```

Example

```
Device(config)# 10 permit tcp any host 192.168.0.99 eq 8443
```

5. Permit DNS UDP traffic.

Example

```
Device(config)# access-list-number permit udp any eq domain
```

Example

```
Device(config)# 20 permit udp any any eq domain
```

6. Permit DHCP traffic using the **bootpc**.

Example

```
Device(config)# 30 permit udp any any eq bootpc
```

Example

```
Device(config)# access-list-number permit udp any any eq bootpc
```

7. Permit DHCP traffic using **bootps**.

Example

```
Device(config)# access-list-number permit udp any any eq bootps
```

Example

```
Device(config)# 40 permit udp any any eq bootps
```

8. Permit the access from the first external web server port 8443 to any host.

Example

```
Device(config)# access-list-number permit tcp any host
external_web_server_ip_address1 eq port-number any
```

Example

```
Device(config)# 50 permit tcp host 192.168.0.98 eq 8443 any
```

9. Permit the access from the second external web server port 8443 to any host.

Example

```
Device(config)# access-list-number permit tcp any host
external_web_server_ip_address2 eq port-number any
```

Example

```
Device(config)# 50 permit tcp host 192.168.0.99 eq 8443 any
```

10. Permit DNS TCP traffic.

Example

```
Device(config)# access-list-number permit tcp any eq domain
```

Example

```
Device(config)# 60 permit tcp any any eq domain
```

11. Deny all the other traffic.

Example

```
Device(config)# access-list-number deny ip any any
```

Example

```
Device(config)# 70 deny ip any any
```

12. Create the WLAN using the **wlan** *wlan-name wlan-id ssid* command.

Example

```
Device(config)# wlan wlan-name wlan-id ssid
```

Example

```
Device(config)# wlan EWLC3-GUEST 3 EWLC3-GUEST
```

13. Configure the IPv4 WLAN web ACL

Example

```
Device(config-wlan)# ip access-group web name
```

The variable *name* specifies the user-defined IPv4 ACL name

14. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

This configuration allows EWA with multiple external web servers and ports. It supports DNS or DHCP traffic and blocks unauthorized traffic.

Configure wired guest EWA with multiple web servers, ports different than default (80/443)

Configure Wired Guest EWA with Multiple Web Servers and/or Ports Different than Default (80/443) using commands.

Configure Wired Guest External Web Authentication (EWA) when using multiple web servers or ports other than the default ports of 80 or 443 using CLI.

Wired Guest LAN profiles do not allow manual ACL assignment directly. To support multiple web servers or custom ports, use the bypass ACL in the global parameter map.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Define an extended IPv4 access list using a name, and enter access-list configuration mode.

Example

```
Device(config)# ip access-list extended BYPASS_ACL
```

3. Allow the traffic to switch centrally.

Example

```
Device(config)# 10 deny ip any host 192.168.0.45
```

4. Allow the traffic to switch centrally

Example

```
Device(config)# access-list-number  deny ip any host hostname
```

Example

```
Device(config)# 20 deny ip any host 4.0.0.1
```

5. Creates a parameter map and enter parameter-map webauth configuration mode.

Example

```
Device(config)# parameter-map type webauth global
```

6. Create a WebAuth bypass intercept using the ACL name.

Example

```
Device(config-params-parameter-map)# webauth-bypass-intercept BYPASS_ACL
```

> **Note**
>
> You cannot manually apply an ACL to the wired guest profile to configure external web authentication with multiple IP addresses or different ports. To work around this, use the bypass ACL in the wired guest profile.

7. Return to privileged EXEC mode.

Example

```
Device(config-params-parameter-map)# end
```

The wired guest profile uses the bypass ACL, enabling external web authentication with multiple web servers or custom ports.

Authenticating sleeping clients

Details authentication strategies for sleeping clients on WLANs, covering mobility guidelines, restrictions, and step-by-step procedures for configuration using GUI and CLI interfaces.

A sleeping client is a wireless device that

- has completed web authentication and been granted guest access
- is allowed to enter sleep mode and wake up without reauthenticating through the login page, and
- the controller stores sleeping client information for a configurable duration before requiring reauthentication.

The valid range is 10 minutes to 43200 minutes, with the default being 720 minutes. You can also configure this duration on WebAuth parameter map that is mapped to a WLAN. The sleeping client timer is activated when conditions such as idle timeout, session timeout, WLAN disabling, or AP nonoperational status occur.

> ⚠ **Caution**
>
> If the MAC address of a client in sleep mode is spoofed, a fake device, such as a laptop, can be mistakenly authenticated.

Feature History

Feature Name	Release	Description
Webauth Sleeping Client Support	Cisco IOS XE 17.1.1s	The web authentication sleeping clients feature supports multiple combinations of authentications for a given client, which are configured on the WLAN profile.

Scenarios where sleeping clients do not need reauthentication

- Suppose there are two controller s in a mobility group. A client that is associated with one controller goes to sleep and then wakes up and gets associated with the other controller .
- Suppose there are three controller s in a mobility group. A client that is associated with the second controller that is anchored to the first controller goes to sleep, wakes up, and gets associated with the third controller .
- A client sleeps, wakes up and gets associated with the same or different export foreign controller that is anchored to the export anchor.

Guidelines for mobility scenarios

- This feature is supported on the FlexConnect scenario with local switching and central authentication.
- L2 roaming in the same subnet is supported.
- Anchor sleeping timer is applicable.
- The sleeping client information is shared between multiple auto anchors when a sleeping client moves from one anchor to another.

Restrictions on authenticating sleeping clients

Authenticating Sleeping Clients restrictions.

- If the MAC address of a client in sleep mode is spoofed, a fake device, such as a laptop, can be mistakenly authenticated.
- The sleep client feature works only for WLAN configured with WebAuth security.
- You can configure the sleeping clients only on a per WebAuth parameter-map basis.
- The authentication of sleeping clients feature is supported only on WLANs that have Layer 3 security enabled.
- When Layer 3 security is enabled, the Authentication, Passthrough, and On MAC Filter failure web policies are supported. The Splash Page Web Redirect web policy is not supported.

- The central web authentication of sleeping clients is not supported.
- The authentication of sleeping clients feature is not supported on guest LANs and remote LANs.
- A guest access sleeping client that has a local user policy is not supported. In this case, the WLAN-specific timer is applied.

Configure authentication for sleeping clients (GUI)

Configure Authentication for Sleeping Clients using GUI.

Enable authentication for sleeping clients to ensure devices can maintain secure network access even after entering sleep mode.

Sleeping clients, such as laptops or mobile devices, periodically enter low-power states. You configure authentication to ensure they reconnect securely and seamlessly when they wake.

Procedure

1. Choose **Configuration** > **Security** > **Web Auth**.
2. In the **Webauth Parameter Map** tab, click the parameter map name. The **Edit WebAuth Parameter** window is displayed.
3. Check **Sleeping Client Status** check box.
4. Click **Update & Apply to Device**.

Authentication for sleeping clients is now enabled. Devices entering sleep mode will reconnect securely without manual intervention.

Configure authentication for sleeping clients (CLI)

Configure Authentication for Sleeping Clients using commands.

Configure authentication parameters to manage sleeping wireless clients and control their session persistence on the device.

Sleeping clients are wireless devices that temporarily disconnect due to power-saving or roaming behaviors. Configuring authentication for sleeping clients ensures they are recognized and handled appropriately during these periods.

Procedure

1. Create a parameter map and enters parameter-map webauth configuration mode.

   Example

   ```
   Device(config)# parameter-map type webauth global
   ```

2. Configure the sleeping client timeout to 100 minutes.

   Example

   ```
   Device(config-params-parameter-map)# sleeping-client timeout 100
   ```

   Valid range is between 10 minutes and 43200 minutes.

> **Note**
>
> If you do not use the timeout keyword, the sleeping client is configured with the default timeout value of 720 minutes.

3. Exit parameter-map webauth configuration mode and returns to privileged EXEC mode.

   Example

   ```
   Device# end
   ```

4. Optional: Show the MAC address of the clients and the time remaining in their respective sessions.

   Example

   ```
   Device# show wireless client sleeping-client
   ```

5. Optional: Delete sleeping client entries from the sleeping client cache.
   - **clear wireless client sleeping-client**–Deletes all sleeping client entries from the sleeping client cache.
   - **clear wireless client sleeping-client mac-address** *mac-addr*–Deletes the specific MAC entry from the sleeping client cache.

   Example

   ```
   Device# clear wireless client sleeping-client
   mac-address 00e1.e1e1.0001
   ```

The device displays the global configuration prompt, allowing you to make configuration changes.

Sleeping Clients With Multiple Authentication

Describes multi-authentication for sleeping clients, outlining mobility support, supported authentication combinations, WLAN configuration for Dot1x, MAC bypass, PSK and LWA, and verification procedures.

Mobility support for sleeping clients

Sleeping Clients Mobility Support.

Feature Name	Release	Description
Mobility support for guest and non guest sleeping clients.	Cisco IOS XE 17.1.1s	Mobility support for guest and non guest sleeping clients.

Supported combinations of multiple authentications

Multiple Authentications Supported Combinations.

Multiple authentication feature supports sleeping clients configured in the WLAN profile.

The table outlines the supported combination of multiple authentications.

Table 113: Supported combinations of multiple authentications

Layer 2	Layer 3	Supported
MAB	LWA	Yes
MAB Failure	LWA	Yes
Dot1x	LWA	Yes
PSK	LWA	Yes

Configure WLAN for 802.1X and local web authentication

Configure WLAN for Dot1x and Local Web Authentication using commands.

Configure a WLAN with both 802.1X authentication and local web authentication on a controller using CLI.

Use this procedure when you need to secure a WLAN such that users can authenticate with their credentials (802.1X) or are redirected to a local web authentication portal. This scenario is common in enterprise or campus environments where both secure and user-friendly authentication methods are required.

Before you begin

Enter the global configuration mode.

Procedure

1. Enter WLAN configuration submode.

   Example

   ```
   Device(config)# wlan wlan-test 3 ssid-test
   ```

   - *profile-name*: Profile name of the configured WLAN.
   - *wlan-id*: Wireless LAN identifier. Range is from one to 512.
   - *SSID_Name*: SSID, which can contain up to 32 alphanumeric characters.

2. Enable security authentication list for dot1x security. The configuration is similar for all dot1x security WLANs.

   Example

   ```
   Device(config-wlan)#  security dot1x authentication-list default
   ```

3. Configure web authentication.

Example

```
Device(config-wlan)#  security web-auth
```

4. Enable authentication list for dot1x security.

Example

```
Device(config-wlan)#  security web-auth authentication-list default
```

5. Map the parameter map.

Example

```
Device(config-wlan)# security web-auth parameter-map global
```

> **Note**
>
> If the parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.

6. Enable WLAN.

Example

```
Device(config-wlan)# no shutdown
```

The new WLAN is configured with both dot1x and local web authentication enabled. Users connecting to this SSID can authenticate using their credentials or through the local web portal, according to controller policies.

Configure a WLAN for MAC authentication bypass and LWA

Configure a WLAN for MAC Authentication Bypass and Local Web Authentication using commands.

Set up a WLAN so that client devices are authenticated using MAC authentication bypass and provided local web authentication.

Use this task to configure a secure SSID that allows MAC-based device authentication followed by a web portal login, ensuring flexible and layered network access control.

Before you begin

Prepare the list name for MAC filtering and ensure the web authentication parameter map is defined (or note that the global map will be used).

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enter WLAN configuration submode.

Example

```
Device(config)# wlan wlan-test 3 ssid-test
```

- *profile-name*: Profile name of the configured WLAN.
- *wlan-id*: Wireless LAN identifier. Range is from 1 to 512.
- *SSID_Name*: SSID, which can contain up to 32 alphanumeric characters.

3. Set the MAC filtering parameters using the **mac-filtering** *list-name*

Example

```
Device(config-wlan)#  mac-filtering cat-radius
```

4. Disable security AKM for dot1x.

Example

```
Device(config-wlan)#  no security wpa akm dot1x
```

5. Disable the WPA2 cipher.

Example

```
Device(config-wlan)# no security wpa wpa2 ciphers aes
```

aes–Excryption type that specifies WPA/AES support.

6. Map the parameter map.

Example

```
Device(config-wlan)# security web-auth parameter-map global
```

> **Note**
>
> If parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.

7. Enable WLAN.

Example

```
Device(config-wlan)# no shutdown
```

The WLAN is configured to use MAC Authentication Bypass with Local Web Authentication.

Configure a WLAN for local web authentication and MAC filtering

Configure a WLAN for Local Web Authentication and MAC Filtering using commands.

Establish a WLAN profile with local web authentication, including MAC address filtering for enhanced security.

Use this task to set up a WLAN on your Cisco device that requires users to authenticate via a local web portal and applies MAC address filtering as an additional security measure.

Before you begin

- Know the desired SSID name, WLAN ID, and profile name.
- Prepare the MAC filter list name.
- Confirm availability of an authentication parameter map (if needed).

Procedure

1. Enters WLAN configuration submode using the **wlan** *profile-name wlan-id SSID_name* command.

   Example

   ```
   Device(config)# wlan wlan-test 3 ssid-test
   ```

   - *profile-name*: Profile name of the configured WLAN.
   - *wlan-id*: Wireless LAN identifier. Range is from one to 512.
   - *SSID_Name*: SSID, which can contain up to 32 alphanumeric characters.

2. Set the MAC filtering parameters.

   Example

   ```
   Device(config-wlan)#  mac-filtering cat-radius
   ```

3. Disable security Authenticated Key Management (AKM) for dot1x.

   Example

   ```
   Device(config-wlan)#  no security wpa akm dot1x
   ```

4. Disable the WPA2 cipher.

   Example

   ```
   Device(config-wlan)# no security wpa wpa2 ciphers aes
   ```

aes: Excryption type that specifies WPA/AES support.

5. Configure the fallback policy with MAC filtering and web authentication.

Example

```
Device(config-wlan)# security web-auth on-macfilter-failure wlan-id
```

6. Map the parameter map.

Example

```
Device(config-wlan)# security web-auth parameter-map global
```

> **Note**
>
> If the parameter map is not associated with a WLAN, the configuration is considered from the global parameter map

7. Enable WLAN.

Example

```
Device(config-wlan)# no shutdown
```

The WLAN is now configured for local web authentication with MAC filtering, ready for users to connect and authenticate through the portal.

Configure a PSK + LWA in a WLAN

Configure a PSK + LWA in a WLAN using commands.

Set up a WLAN with both a preshared key (PSK) and local web authentication (LWA) using CLI commands.

Use this task to enable secure access for clients on a WLAN using both a PSK and web authentication mechanisms.

Before you begin

- Decide the profile name, WLAN ID, and SSID for the new WLAN.
- Create required authentication lists and parameter maps if not already present.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enter the WLAN configuration sub-mode.

Example

```
Device(config)# wlan wlan-name wlan-id SSID-name
```

3. Disables security AKM for dot1x.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

4. Enables web authentication for a WLAN.

   Example

   ```
   Device(config-wlan)# security web-auth
   ```

5. Disable the WPA2 cipher.

   Example

   ```
   Device(config-wlan)# no security wpa wpa2 ciphers aes
   ```

   aes: Excryption type that specifies WPA/AES support.

6. Configure the preshared key on a WLAN.

   Example

   ```
   Device(config-wlan)# security wpa psk set-key ascii 0 1234567
   ```

7. Configure PSK support.

   Example

   ```
   Device(config-wlan)#security wpa akm psk
   ```

8. Enable the authentication list for dot1x security.

   Example

   ```
   Device(config-wlan)# security web-auth authentication-list default
   ```

9. Map the parameter map.

   Example

   ```
   Device(config-wlan)# security web-auth parameter-map global
   ```

> **Note**
>
> If the parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.

The WLAN is configured to use both a preshared key and local web authentication for client access.

Configure a sleeping client

Configure a Sleeping Client using commands.

Configure parameters that determine how long a client device remains in "sleeping" state before timing out, using CLI.

Sleeping clients are devices that temporarily disconnect from the network but retain their association for a defined period, allowing smooth reconnection. You may wish to adjust the sleeping client timeout to optimize performance or manage resource use.

Before you begin

Identify the parameter map (by name or use the global map) you wish to modify.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a parameter map and enters *parameter-map-name* configuration mode using the **parameter-map type webauth** {*parameter-map-name* | **global**} command.

   Example

   ```
   Device(config)# parameter-map type webauth MAP-2
   ```

   The specific configuration commands supported for a **global** parameter map defined with the global keyword differ from the commands supported for a named parameter map defined with the *parameter-map-name* argument.

3. Configure the sleeping client timeout, in minutes using the **sleeping client** [**timeout** *time*] command.

   Example

   ```
   Device(config-params-parameter-map)#  sleeping-client timeout 60
   ```

   The available range for the *time* argument is from 10 to 43200.

> **Note**
>
> If you do not use the **timeout** keyword, the sleeping client is configured with the default timeout value of 720 minutes.

The sleeping client timeout is configured for the specified parameter map.

Verify a sleeping client configuration

Verify a Sleeping Client Configuration.

To verify a sleeping client configuration, use the command:

```
Device# show wireless client sleeping-client
Total number of sleeping-client entries: 1

MAC Address                    Remaining time (mm:ss)
------------------------------------------------------------
2477.031b.aa18                 59:56
```

Multiple Authentication Combinations with 802.1X and LWA

Explains multi-authentication integration of 802.1X and Local Web Authentication, outlining information, limitations, CLI configuration procedures, verification steps, and keyref reuse approaches for policy retention.

Multiauthentication combinations with 802.1X authentication and local web authentication

Multiauthentication Combination with 802.1X Authentication and Local Web Authentication details.

A multiauthentication combination is a network access control strategy that

- allows multiple authentication methods (such as 802.1X and web authentication) to be used in sequence
- enables enforcement of security and user consent policies in wireless networks, and
- supports policy merging to maintain consistent access controls throughout device connections.

Expanded Explanation

In a wireless setup, for example, in a university, clients authenticate through 802.1X authentication. Because the 802.1X (dot1X) authentication process is secure and does not require user intervention, the end-users are unaware of the network that their devices are connected to. This could lead to serious concerns if they connect to the university's wireless network and post inappropriate content or access restricted content.

To avoid this situation, web authentication (webauth) and 802.1X authentication are configured in the network. End-user consent is used as a part of webauth to inform users that they are connected to the university's Wi-Fi network.

When the end-users accept the credentials for consent, AAA policies are not applied. The AAA policies that were applied earlier are deleted, resulting in a VLAN change and client disconnection.

A new command is introduced in Cisco IOS XE Dublin 17.11.1 to fix this issue. When you run the **consent activation-mode merge** command, the policy that is applied through consent is merged with the policy applied for 802.1X or MAC

Authentication Bypass (MAB) authentication, thereby allowing clients to access the network. This command is available in parameter-map mode, which is configured with **type consent** command.

Feature History

Release	Feature	Feature Information
Cisco IOS XE Dublin 17.11.1	Multiauthentication Combination of 802.1X and Local Web Authentication	This feature supports the merging of applied policies during multiauthentication of 802.1X or MAC authentication bypass (MAB) and local web authentication (LWA).

Limitations for multi authentication combination of 802.1X and local web authentication

Multi Authentication Combination of 802.1X and Local Web Authentication limitations.

- It is not possible to configure this feature on the controller GUI.
- SNMP is not supported.
- When the **consent activation-mode merge** command is not configured on the webauth parameter map, the default activation mode is Replace. This means that the user profile for consent replaces all the user profile policies that were previously applied.

Enable the multiauthentication combinations of 802.1X authentication and local web authentication (CLI)

Enable the Multiauthentication Combination of 802.1X Authentication and Local Web Authentication using commands.

Configure multiauthentication by combining 802.1X authentication and local web authentication (LWA) on the device using CLI commands.

Use this procedure when you want to allow users to authenticate through both 802.1X and browser-based local web authentication. This is commonly needed for networks supporting multiple authentication types on a single interface.

Before you begin

Ensure that you have working knowledge of multiauthentication concepts, LWA (consent), and AAA override.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the webauth type parameter. Enters the parameter map configuration mode.

   Example

   ```
   Device(config)# parameter-map type webauth parameter-map1
   ```

3. Configure the type as **consent**.

Example

```
Device(config-params-parameter-map)# type consent
```

4. Enable policy activation mode and merges the previous policy using the **[no] consent {activation-mode merge | email}** command.

   Example

   ```
   Device(config-params-parameter-map)# consent activation-mode merge
   ```

   Run the **no** form of this command to disable the feature.

Multiauthentication combining 802.1X authentication and local web authentication is enabled on the device.

Verify multiauthentication combination with 802.1X authentication and local web authentication

Verify Multiauthentication Combination with 802.1X Authentication and Local Web Authentication.

To verify the multiauthentication combination with 802.1X authentication and LWA, run the command:

```
Device# show parameter-map type webauth lwa-consent
Parameter Map Name                : lwa_consent
  Banner Title                    : Consent Title
  Banner Text                     : Please accept the consent
  Type                            : consent
  Auth-proxy Init State time      : 300 sec
  Webauth max-http connection     : 200
  Webauth logout-window           : Enabled
  Webauth success-window          : Enabled
  Consent Email                   : Disabled
  Activation Mode                 : Merge
  Sleeping-Client                 : Disabled
  Webauth login-auth-bypass:
```

Title

Device

120 Central Web Authentication

Topics:

- Central web authentication

Central web authentication

Central Web Authentication details.

A central web authentication is a wireless security mechanism that

- delegates the user login webpage and credential processing to a centralized authentication server such as Cisco ISE
- initiates user redirection and authentication at Layer 2, often in combination with MAC filtering or 802.1X authentication, and
- uses RADIUS server attributes so that network controllers redirect web traffic to the authentication portal. This approach streamlines access and reduces delays.

- **ISE (Identity Services Engine)**: Cisco's centralized network access policy platform that delivers authentication, authorization, accounting, and guest access functionality.
- **ACL (Access control list)**: A set of rules used to control network traffic and enforce security policies, determining which clients can access which network resources.
- **Change of Authorization (CoA)**: A mechanism to reapply or update authorization parameters dynamically without requiring a new login.

> **Note**
>
> Currently, Cisco Identity Services Engine (ISE) supports only one Attribute-Value Pair (AVP) for Access Control Lists (ACLs), either IPv4 or IPv6. This limitation prevents the simultaneous return of both ACLs within the same authentication profile. As a result, dual-stack clients (devices using both IPv4 and IPv6) can operate within a Central Web Authentication (CWA) flow, but only under specific configurations. To avoid issues, it is recommended to configure a default IPv6 ACL with a 'deny all' rule. This ensures that clients do not encounter connectivity problems while the limitation is addressed through a future feature enhancement

Prerequisite

Cisco ISE

Comparison with other authentication methods

Central web authentication (CWA) lets you manage web-based access for wireless clients through a dedicated authentication portal (typically Cisco ISE). Unlike traditional local web authentication, CWA performs core security and redirection at Layer 2, working with authentication mechanisms such as MAC filtering or 802.1X. The RADIUS server sends specific attributes to the controller, which then redirects web traffic for portal-based login. This approach helps streamline authentication, reduce delays, and improve policy enforcement.

There are multiple methods of web authentication used in wireless environments.

- **Local web authentication (LWA)**: Local web authentication (LWA) is Layer 3 security configured on the controller. The authentication page and pre-authentication ACL are configured locally. The controller intercepts HTTP(S) traffic from the client and redirects the client to an internal authentication web page. The controller authenticates credentials locally or through RADIUS or LDAP.
- **External web authentication (EWA)**: Also configured as Layer 3 security on the controller. The controller intercepts HTTP(S) traffic and redirects the client to a login page hosted on an external web server. Credentials are authenticated by the controller locally or through a RADIUS or LDAP server. The pre-authentication ACL is statically configured on the controller.

- **Central web authentication (CWA)**: Typically configured as Layer 2 security, with the redirection URL and pre-authentication ACL residing on Cisco ISE. During Layer 2 authentication, Cisco ISE pushes the redirection attributes to the controller. The controller redirects all web traffic from the client to the Cisco ISE login page. Cisco ISE then validates the credentials entered by the client through HTTPS and authenticates the user.

This table compares different types of authentication methods used in wireless environments.

Table 114: Difference from other authentication types

Method	Layer	Portal	Credential processing	Key attributes
Local web authentication	Layer 3	Controller	Locally or through RADIUS or LDAP	Controller intercepts traffic
External web authentication	Layer 3	External server	Locally or through RADIUS or LDAP	Static ACLs and external portal
Central web authentication	Layer 2	Cisco ISE	Through Cisco ISE (HTTPS)	Cisco ISE

Analogy: concert tickets

Imagine a concert venue with several ways to check tickets and admit guests. You can have ticket booths at every entrance (local authentication), ticket checkers who send guests to a special desk outside (external authentication), or one main VIP booth at the heart of the venue that handles everyone's tickets and access (central authentication). Let's use this concert analogy to understand central web authentication and other methods.

At your concert venue, central web authentication (CWA) is what happens when, instead of letting every entrance or gate have their own ticket booth, you create one exclusive VIP booth—like Cisco ISE—that manages all ticketing for everyone. Instead of waiting until a guest actually tries to enter through a particular door (the way local booth might do), the venue's security starts checking guests' tickets as soon as they enter the red carpet. The VIP booth can give the gatekeepers special instructions: "If you don't recognize someone's ticket, redirect them straight to me!" This means the main ticketing process is handled efficiently and quickly by one central authority.

Let's look at all the ticketing strategies you could use at your concert:

Local Ticket Booth (Local Web Authentication, LWA): Every entrance has its own mini ticket booth and rules. Guards at the door check tickets and can ask guests for their info. Ticket validation is handled locally at each gate, sometimes via a backstage manager or external system.

External Ticket Desk (External Web Authentication, EWA): Instead of ticket checks at the gate, guests are sent to a desk outside the stadium. The desk is run by another company. The security at the entrance gates redirects guests and the validation can still interact with the backstage manager if needed. Rules for who gets through are set upfront.

VIP Central Ticket Booth (Central Web Authentication, CWA): The gates just check basic details (like guest's wristband color), and anyone who isn't recognized is sent straight to the main VIP booth (Cisco ISE) to have their ticket or credentials checked and get access granted for the whole event.

How to configure ISE

Configuring ISE.

To configure ISE, follow these tasks:

1. Create an authorization profile.
2. Create an authentication rule.

3. Create an authorization rule.

Create an authorization profile

Create an Authorization Profile.

Define an authorization profile for central web authentication with required redirect attributes.

Before you begin

Ensure you have the required access and know the ACL name for redirection.

Procedure

1. Click **Policy**, and click **Policy Elements**.
2. Click **Results**.
3. Expand **Authorization**, and click **Authorization Profiles**.
4. Click **Add** to create a new authorization profile for central web authentication.
5. In the **Name** field, enter a name for the profile. For example, CentralWebAuth.
6. Choose **ACCESS_ACCEPT** from the **Access Type** drop-down list.
7. Check the **Web Redirection (CWA, MDM, NSP, CPP)** check box, and choose **Centralized Web Auth** from the drop-down list.
8. In the **ACL** field, enter the name of the ACL that defines the traffic to be redirected. For example, redirect.
9. In the **Value** field, choose the default or customized values.

   This field defines whether the Cisco ISE displays the default or a custom web portal that the Cisco ISE admin created.

10. Click **Save**.

The authorization profile is created and is available for use in your central web authentication policies.

Create an authentication rule

Create an Authentication Rule.

Define an authentication rule to control network access based on specified conditions.

Use authentication rules to specify how users or devices are authenticated in your policy system. This rule determines which authentication methods and identity sources are applied when network access is requested.

Procedure

1. In the **Policy** > **Authentication** page, click **Authentication**.
2. Enter a name for your authentication rule. For example, MAB.
3. In the If condition field, select the plus (+) icon.
4. Choose **Compound condition**, and choose **Wireless_MAB**.
5. Click the arrow next to **and ...** in order to expand the rule further.
6. Click the + icon in the **Identity Source** field, and choose **Internal endpoints**.
7. Choose **Continue** from the **If user not found** drop-down list.

   A device can be authenticated even if its MAC address is not known when you select this option.

8. Click **Save**.

Your new authentication rule is added to the authentication policy.

Create an authorization rule

Create an Authorization Rule.

Define and enforce policies to control access to network resources for users and devices.

Use authorization rules to allow or deny access based on user, device, or authentication attributes in Cisco ISE.

You can configure many rules in the authorization policy, such as the *MAC not known*.

Procedure

1. Click **Policy** > **Authorization**.
2. In the **Rule Name** field, enter a name. For example: *Mac not known*.
3. In the **Conditions** field, click the plus (+) icon.
4. Choose **Compound Conditions**, and choose **Wireless_MAB**.
5. From the settings icon, select **Add Attribute/Value** from the options.
6. In the **Description** field, choose **Network Access** > **AuthenticationStatus** as the attribute from the drop down list.
7. Choose the **Equals** operator.
8. From the right hand field, choose **UnknownUser**.
9. In the **Permissions** field, choose the name of the authorization profile you had created earlier.

   The Cisco ISE continues even though the user (or MAC) is not known.

   Unknown users now see the login page. When they enter their credentials, the Cisco ISE sends another authentication request. To support guest users, configure a rule to check if the user is in the guest group. For instance, if UseridentityGroup Equals Guest is used, all guests are included in this group.

10. In the **Conditions** field, click the plus (+) icon.
11. Choose **Compound Conditions**, and choose to create a new condition.

    Place the new rule so it is evaluated prior to the *MAC not known* rule.

12. From the settings icon, select **Add Attribute/Value** from the options.
13. In the **Description** field, choose **Network Access** > **UseCase** as the attribute from the drop-down list.
14. Choose the **Equals** operator.
15. From the right hand field, choose **GuestFlow**.
16. In the **Permissions** field, click the plus (+) icon to select a result for your rule.

    You can choose **Standard** > **PermitAccess** option or create a custom profile to return the attributes that you like.

    When the user is authorized on the login page, the Cisco ISE triggers a Change of Authorization (COA). This process restarts Layer 2 authentication. If the user is identified as a guest, the user is authorized.

Cisco ISE evaluates user or device access requests against these rules and enforces the appropriate policy. Unknown users see the login page and, if identified as guests, receive guest access upon authentication.

How to configure central web authentication on the controller

Configure Central Web Authentication on the Controller.

To configure central web authentication on the controller, proceed with these tasks.

1. Configure WLAN.
2. Configure policy profile.
3. Configure redirect ACL.
4. Configure AAA for central web authentication.
5. Configure redirect ACL in Flex profile.

Configure WLAN (GUI)

Configure a WLAN using the GUI.

Set up a new WLAN on your wireless controller using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. In the **WLANs** window, click the name of the **WLAN** or click **Add** to create a new one.
   The **Add/Edit WLAN** window is displayed.
3. In the **Add/Edit WLAN** window, click the **General** tab to configure these parameters.
   - In the **Profile Name** field, enter or edit the name of the profile.
   - In the **SSID** field, enter or edit the SSID name.

     The SSID name is alphanumeric and can be up to 32 characters in length.
   - In the **WLAN ID** field, enter or edit the ID number. The valid range is between one and 512.
   - Select the **802.11** radio band from the **Radio Policy** drop-down list.
   - Using the **Broadcast SSID** toggle button, change the status to either **Enabled** or **Disabled**.
   - Using the **Status** toggle button, change the status to either **Enabled** or **Disabled**.
4. Click the **Security** tab, and then select the **Layer 2** tab to configure these parameters:
   - Select **None** from the **Layer 2 Security Mode** drop-down list. This setting disables Layer 2 security.
   - Enter the **Reassociation Timeout** value, in seconds. This value specifies the duration before a fast transition reassociation times out.
   - Check the **Over the DS** check box to enable Fast Transition over a distributed system.
   - Choose **OWE**. Opportunistic Wireless Encryption (OWE) provides data confidentiality with encryption over the air between an AP radio and a wireless client. OWE Transition Mode ensures backwards compatibility.
   - Choose Fast Transition (802.11r), the IEEE standard for fast roaming. This standard allows the initial handshake with a new AP to occur before the client roams to the target AP. This method is known as Fast Transition.
   - Check the check box to enable MAC filtering in the WLAN.
   - Check the **Lobby Admin Access** check box to enable Lobby Admin access.

5. Click **Save & Apply to Device**.

Configure WLAN (CLI)

Set up a new WLAN on your wireless controller using the CLI.

Before you begin

Enable MAC filtering for Layer 2 authentication to download the redirect URL and ACL.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN.

   Example

   ```
   Device(config)# wlan wlan-name wland-id SSID-name
   ```

   - **wlan-name** is the name of the configured WLAN.
   - **wlan-id** is the WLAN identifier. The range is one to 512.
   - **SSID-name** is the SSID name which can have up to 32 alphanumeric characters.

   If you have already created and configured the WLAN, use the **wlan** *wlan-name* command.

3. Enable MAC filtering on a WLAN.

   Example

   ```
   Device(config-wlan)# mac-filtering name
   ```

   If the authentication list is not already configured, the default authentication list is used when configuring MAC filtering.

4. Disable WPA security.

   Example

   ```
   Device(config-wlan)# no security wpa
   ```

5. Enable the WLAN.

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

6. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

The new WLAN is available and active on the controller.

After completing the WLAN configuration, if the changes are not pushed to all the APs, a syslog message appears.

```
2021/01/06 16:20:00.597927186 {wncd_x_R0-4}{1}: [wlanmgr-db] [20583]: UUID:
0, ra: 0, TID: 0
(note): Unable to push WLAN config changes to all APs, cleanup required for
WlanId: 2, profile: wlan1 state: Delete pending
```

If this syslog message appears for more than six minutes, reload the controller

If the controller does not reload and the syslog message continues to appear, collect the archive logs and wncd core file. Then, raise a case by clicking the link: Support Case Manager.

```
Device# config terminal
            Device(config)# wlan wlanProfileName 1 ngwcSSID
            Device(config-wlan)# mac-filtering default
            Device(config-wlan)# no security wpa
            Device(config-wlan)# no shutdown
            Device(config-wlan)# end
```

Configure policy profile (CLI)

Define and activate a policy profile to control WLAN behavior using the CLI.

Use the CLI to set up policy profiles that govern AAA, NAC, and VLAN assignment for wireless clients on Cisco devices. Apply this procedure when building new WLAN policies or adjusting network admission controls.

Before you begin

- Both NAC and AAA override must be configured in the policy profile for proper CWA operation.
- You need a AAA override to apply policies coming from the AAA or ISE servers. When a redirect URL and redirect ACL is received from the ISE server, NAC is used to trigger the Central Web Authentication (CWA).
- The default policy profile is used for APs not associated with a specific policy.
- Clients may experience VLAN assignment during the final RADIUS access-accept following successful authentication.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the WLAN.

Example

```
Device(config)# wlan wlan-name wland-id SSID-name
```

- **wlan-name** is the name of the configured WLAN.
- **wlan-id** is the WLAN identifier. The range is one to 512.
- **SSID-name** is the SSID name which can have up to 32 alphanumeric characters.

If you have already created and configured the WLAN, use the **wlan** *wlan-name* command.

3. Set the policy profile.

Example

```
Device(config)# wireless profile policy default-policy-profile
```

4. Map the VLAN to a policy profile.

Example

```
Device(config-wireless-policy)# vlan vlan-id
```

If vlan-id is not specified, the default native vlan one is applied. The valid range for vlan-id is one to 4096. Management VLAN is applied if no VLAN is configured on the policy profile.

5. Configure AAA override to apply policies coming from the AAA or ISE servers.

Example

```
Device(config-wireless-policy)# aaa-override
```

6. Configure Network Access Control in the policy profile. NAC is used to trigger the Central Web Authentication (CWA).

Example

```
Device(config-wireless-policy)# nac
```

7. Enable the WLAN.

Example

```
Device(config-wireless-policy)# no shutdown
```

8. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

The policy profile is configured and active, applying the desired AAA, NAC, and VLAN parameters to WLAN clients.

```
Device# configure terminal
Device(config)# wireless profile policy default-policy-profile
Device(config-wireless-policy)# vlan 41
Device(config-wireless-policy)# aaa-override
Device(config-wireless-policy)# nac
Device(config-wireless-policy)# no shutdown
Device(config-wireless-policy)# end
```

Configure a policy profile (GUI)

Configure a Policy Profile using GUI.

Create and apply a new policy profile using the GUI.

Use this task to define access and traffic management settings by configuring a policy profile in the system's GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. On the **Policy Profile** page, click **Add**.
3. In the **Add Policy Profile** window, in General Tab, enter a name and description for the policy profile.
4. To enable the policy profile, set **Status** as Enabled.
5. Use the slider to enable or disable **Passive Client** and **Encrypted Traffic Analytics** .
6. Optional: In the **CTS Policy** section, choose the appropriate status for the following:
   - Inline Tagging—a transport mechanism using which a embedded wireless controller, controller, or AP understands the source SGT.
   - SGACL Enforcement
7. Specify a default **SGT**. The valid range is from two to 65519.
8. In the **WLAN Switching Policy** section, choose as required:
   - Central Switching
   - Central Authentication
   - Central DHCP
   - Central Association Enable
   - Flex NAT/PAT
9. Click **Save & Apply to Device**.

The new policy profile is created and applied to the device using your configured settings.

Create redirect ACL

Create Redirect ACL using commands.

Create a redirect access control list (ACL). Direct unauthenticated HTTP and HTTPS traffic to the Cisco Identity Service Engine login page. Allow traffic to Cisco ISE and block other traffic

The redirect ACL is a punt ACL that must be predefined on the controller. For FlexConnect local switching, the ACL must be predefined on the AP. The Authentication, authorization, and accounting (AAA) server server returns only the name of the ACL, not its definition.

The redirect ACL defines traffic matching "deny" statements, allowing this traffic to pass through the data plane. Traffic matching "permit" statements is sent to the control plane for further processing, which includes web interception and redirection.

The redirect ACL includes implicit statements that allow DHCP and DNS traffic to all IP addresses, similar to LWA.

It also ends with an implicit deny statement in the security ACL.

Before you begin

Ensure you have the IP address of your Cisco ISE deployment.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create an extended access list.

   Example

   ```
   Device(config)# ip access-list extended redirect
   ```

   If users are not authenticated, HTTP and HTTPS browsing fails because Cisco ISE uses a redirect ACL (named **`redirect`**).

3. Permit traffic to Cisco ISE and block all other traffic.

   Example

   ```
   Device(config)# deny ip any host ISE-IP-address
   ```

4. Permit traffic to Cisco ISE and all other traffic is blocked.

   Example

   ```
   Device(config)# deny ip host ISE-IP-add any
   ```

   > **Note**
   >
   > This ACL is applicable for both local and flex mode.

5. Redirect all HTTP or HTTPS access to the ISE login page. Port number 80 is used for HTTP, and port number 443 is used for HTTPS, configured with the **permit TCP any any eq `web address/port-number`** command.

Example
In case of HTTP:

```
Device(config)# permit TCP any any eq www
```

```
Device(config)# permit TCP any any eq 80
```

Example
In case of HTTPS:

```
Device(config)# permit TCP any any eq 443
```

To allow traffic to Cisco ISE, configure Cisco ISE before the HTTP or HTTPS ACE entry.

6. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

HTTP

```
Device# configure terminal
Device(config)# ip access-list extended redirect
Device(config)# deny ip any host 123.123.134.112
Device(config)# deny ip host 123.123.134.112 any
Device(config)# permit TCP any any eq www
Device(config)# permit TCP any any eq 80
Device(config)# end
```

HTTPS

```
Device# configure terminal
Device(config)# ip access-list extended redirect
Device(config)# deny ip any host 123.123.134.112
Device(config)# deny ip host 123.123.134.112 any
Device(config)# permit TCP any any eq 443

Device(config)# end
```

Configure AAA for CWA

Configure AAA for Central Web Authentication using commands.

Set up AAA to support Central Web Authentication using CLI.

Central Web Authentication requires proper AAA configuration to enable Change of Authorization (CoA) functions using a RADIUS (Remote Authentication Dial-In User Service) server.

Before you begin

Ensure AAA authentication is configured on your device. For more details, see the *Configuring AAA Authentication* documentation.

Before you begin

Procedure

1. Configure the Change of Authorization (CoA) on the controller.

   Example

   ```
   Device(config)# aaa server radius dynamic-author
   ```

2. Specify a RADIUS client and configure the RADIUS key that is shared between the device and the RADIUS client.

   Example

   ```
   Device(config-locsvr-da-radius)# client ISE-IP-add server-key
   radius-shared-secret
   ```

   ISE-IP-add is the IP address of the RADIUS client.

   server-key is the radius client server-key.

   radius-shared-secret can have these values:

   - **0**–Specifies unencrypted key.
   - **6**–Specifies encrypted key.
   - **7**–Specifies HIDDEN key.
   - **Word**–Unencrypted (cleartext) server key.

   When configuring WSMA data in GUI, ensure that the RADIUS shared secret does not exceed 240 characters.

AAA is configured to support Central Web Authentication with dynamic CoA, enabling secure device communication with the RADIUS server.

```
Device# config terminal
Device(config)# aaa server radius dynamic-author
Device(config-locsvr-da-radius)# client 123.123.134.112 server-key 0 SECRET
Device(config-locsvr-da-radius)# end
```

Configure redirect ACL in flex profile (GUI)

Configure Redirect ACL in Flex Profile using GUI.

Set up a redirect ACL within a flex profile so that APs enforce specified access control and web authentication policies.

The redirect ACL definition must be sent to the AP in the FlexConnect profile. The redirect ACL associated with an AP must be configured in the FlexConnect profile that hosts the client. If an AP is not configured with a FlexConnect profile, it is associated with the default FlexConnect profile.

Before you begin

- Identify the FlexConnect profile to be updated or verify if a new Flex profile needs to be created.
- Prepare the ACL details, central web authentication requirements, and any preauthentication URL filters.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Flex**.
2. On the **Flex Profile** page, click the name of the FlexConnect profile or click **Add** to create a new FlexConnect profile.
3. In the **Add/Edit Flex Profile** window that is displayed, click the **Policy ACL** tab.
4. Click **Add** to map an ACL to the FlexConnect profile.
5. Choose the ACL name, enable central web authentication, and specify the preauthentication URL filter.
6. Click **Save**.
7. Click **Update & Apply to Device**.

The AP applies the configured redirect ACL parameters for client traffic as specified in the flex profile.

Configure redirect ACL in flex profile (CLI)

Configure Redirect ACL in Flex Profile using commands.

Set up a redirect ACL within a flex profile so that APs enforce specified access control and web authentication policies.

The redirect ACL definition must be sent to the AP in the FlexConnect profile. The redirect ACL associated with an AP must be configured in the FlexConnect profile that hosts the client. If an AP is not configured with a FlexConnect profile, it is associated with the default FlexConnect profile.

Note

When the ACL is pushed down to the APs, the permission must change from **deny** to **permit** or the other way around. This change does not occur if the ACL contains an object group, causing the ACL not to be fully translated, which may cause the redirection to fail.

Before you begin

- Ensure you have the required ACL policy created on the controller.
- Confirm that AP is operating in FlexConnect mode.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a new flex policy.

   Example

   ```
   Device(config)# wireless profile flex default-flex-profile
   ```

   The default flex profile name is **`default-flex-profile`**.

3. Configure an ACL policy.

   Example

```
Device(config-wireless-flex-profile)# acl-policy
                        acl-policyname
```

4. Configure central web authentication.

   Example

```
Device(config-wireless-flex-profile-acl)# central-webauth
```

5. Return to privileged EXEC mode.

   Example

```
Device(config-wireless-flex-profile-acl)# end
```

The AP in flex profile has the redirect ACL configured. The ACL is now applied and used for client redirection during centralized web authentication.

Troubleshoot CWA

Init-State timer running out

Issue: The client devices are deauthenticated by the controller if users fail to enter their credentials in a limited time interval. The clients are deauthenticated after three times the time configured for the init-state timeout in the controller.

Explanation: This is expected behavior. For central web authentication, the init-state timeout does not apply directly. The reap timer—three times the init-state timeout plus five seconds (3 × init-state timeout + 5)—determines when your device deauthenticates in seconds.

For example, if you have configured the init-state timeout as 10 seconds, then the client devices are deauthenticated if users fail to enter their credentials after 35 seconds; that is (3*10 + 5) = 35 seconds.

For example, if you set the init-state timeout to 10 seconds, your device deuathenticates after 35 seconds if you do not enter your credentials. (This calculation is: 3 × 10 + 5 = 35 seconds.)

121 Private Shared Key

Topics:

- Private preshared keys

Private preshared keys

Introduces Private PSK authentication, detailing system limitations, identity PSK operation, and step-by-step procedures for configuring and applying PSKs on WLANs and policy profiles via GUI and CLI, including verification methods for secure wireless access.

A private preshared key (PSK) is a security protocol for network authentication that

- enables devices to securely connect to wireless local area networks (WLANs),
- assigns unique keys to individual devices or user groups, reducing the risk of unauthorized access, and
- ensures compatibility with systems where advanced authentication methods like 802.1X are unsupported.

Understanding identity PSKs

Identity PSKs are unique PSKs created for individuals or groups of users on the same SSID. They provide the same simplicity of PSK while enabling features such as:

- Stronger security for IoT by supporting devices that do not use 802.1X.
- Easy revocation of access for a specific device or individual without affecting others.
- Scalable management of thousands of keys distributed through the AAA server.

These features make Identiy PSKs ideal for IoT, BYOD, and guest deployments.

Limitations

SSID Preshared Key Special Characters

- Special characters, such as '<' and '>' are supported in SSID Preshared key.
- Avoid using the percentage symbol (%) in a PSK. Some client and access point combinations may interpret the percentage symbol as a special character, which can prevent the client from joining the PSK WLAN. If clients cannot join the WLAN, replace the symbol with another supported ASCII character and retry.

SSID Preshared Key Whitespace Support

PSK supports whitespace in passwords (before or after or in-between) within double quotes only; single quotes for whitespaces are not supported.

How identity PSK authentication works

Identity Pre-Shared Key (IPSK) authentication allows a unique pre-shared key to be assigned to each device or user based on their MAC address, enhancing network security and control. You collaborate with multiple network components to authorize, authenticate, and provision device access.

Summary

The key components involved in the process are:

- Client device: Sends an association request to connect to the access point.
- AP: Broadcasts the SSID and forwards association requests to the controller.
- Embedded Wireless Controller: Constructs RADIUS requests and relays them to the authentication server; manages computation of PSKs and applies received configuration parameters.
- RADIUS (AAA) server: Authenticates client MAC addresses, authorizes access, and returns passphrases or additional parameters as needed.

Workflow

Here are the stages you will follow for Identity PSK authentication:

1. The client device attempts to connect to the wireless network by sending an association request to the SSID broadcast by the access point.
2. The access point forwards the association request to the Embedded Wireless Controller.
3. The controller creates a RADIUS authentication request packet containing the client's MAC address and relays it to the RADIUS (AAA) server.
4. The RADIUS server performs authentication and determines whether the client is authorized.
   - If authorized, the server sends an ACCESS-ACCEPT response; otherwise, it sends an ACCESS-REJECT.
5. If the client is authorized, the RADIUS server includes the AV-pair passphrase (the identity PSK) in the response, as well as optional parameters such as username, VLAN, and Quality of Service (QoS).
6. Upon receiving the response, the controller uses the passphrase to compute the Pairwise Master Key (PMK) and apply any additional attributes for the client's session.
7. If you have multiple devices, the same passphrase can apply to all of them.
8. If a configured PSK is shorter than 15 characters (when Federal Information Processing Standard, FIPS, is enabled), the controller allows the WLAN configuration but displays the following console warning:

> **Note**
>
> AP is allowed to join but corresponding WLAN will not be pushed to the access point.

Result

Your device will either be granted access to the network with its assigned unique PSK and relevant settings, or access will be denied based on the authentication outcome. This information could be better presented in a separate paragraph or subsection to distinctively address the outcome of the process.

Configure a PSK in a WLAN

Configure a PSK in a WLAN using commands.

Set up security for a pre-shared key (PSK) in a WLAN environment using command-line instructions.

Before you begin

- Configure security for a pre-shared key (PSK) in a WLAN.
- If there is no override from the AAA server, the system uses the WLAN value for authentication.
- In Federal Information Processing Standard (FIPS) and common criteria mode, ensure that the PSK WLAN uses at least 15 ASCII characters.

Procedure

1. Configure terminal

Example

```
Device# configure terminal
```

Enters global configuration mode.

2. Configure the WLAN and the SSID

Example

```
Device(config)# wlan test-profile 4 abc
```

Configures the WLAN and SSID.

3. Disable security AKM for 802.11

Example

```
Device(config-wlan)# no security wpa akm dot1x
```

Disables security AKM for 802.11x.

4. Configure the security type PSK

Example

```
Device(config-wlan)# security wpa akm psk
```

Configures the security type PSK.

5. Configure PSK authenticated key management (AKM) shared key

Example

```
Device(config-wlan)# security wpa akm psk set-key asci 0
```

Configures the PSK authenticated key management (AKM) shared key.

> **Note**
>
> You must set the **psk set-key** before configuring AKM PSK.

6. Configure PSK support

Example

```
Device(config-wlan)# security wpa akm psk
```

Configures PSK support.

7. Configure multi-preshared key (MPSK) support

Example

```
Device(config-wlan)# security wpa wpa2 mpsk
```

Configures multi-preshared key (MPSK) support.

> **Note**
>
> Enable AKM PSK to use MPSK.

8. Specify MAC filtering in WLAN

Example

```
Device(config-wlan)# mac-filtering test1
```

Specifies MAC filtering in a WLAN.

The WLAN is configured to use a pre-shared key for client authentication.

Configure a PSK in a WLAN using GUI

Configure a PSK in a WLAN using GUI.

Configure a pre-shared key (PSK) for a WLAN.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. On the **Wireless Networks** page, click **Security** tab.
3. In the **Layer 2** window that is displayed, go to the **WPA Parameters** section.
4. From the **Auth Key Mgmt** drop-down, select the PSK format and type.
5. Enter the Pre-Shared Key in hexadecimal characters.
   - If you selected the PSK format as HEX, the key length must be exactly 64 characters.
   - If you selected the PSK format as ASCII, ensure the key length ranges from 8 to 63 characters.

   After configuring the key, these details are not visible due to security reasons. Even if you click on the eye icon next to the Pre-Shared Key box, the details remain hidden.
6. Click **Save & Apply to Device**.

The PSK is configured and applied to the selected WLAN.

What to do next

Check the WLAN connectivity to ensure successful configuration.

Apply a policy profile to a WLAN (GUI)

Apply a policy profile to a WLAN using GUI.

This task helps you configure WLAN by applying policy profiles for better network management.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. On the **Manage Tags** page, click **Policy** tab.
3. Click **Add** to view the **Add Policy Tag** window.
4. Enter the policy tag name and its description.
5. Click **Add** to map WLAN and policy.
6. Choose the WLAN profile to map with the appropriate policy profile, then click the tick icon.
7. Click **Save & Apply to Device**.

Once the task is completed, you successfully map the WLAN to the chosen policy profile

What to do next

Verify the changes by checking the applied profiles under Wireless Local Area Network (WLAN) settings.

Monitor network performance to confirm the policy effectiveness.

Apply a policy profile to a WLAN using CLI

Apply a policy profile to a WLAN using commands.

The goal of this task is to effectively apply a policy profile to a WLAN using CLI commands, which helps manage network configurations efficiently.

Procedure

1. Enter global configuration mode

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. Configure the default policy profile

   Example

   ```
   Device(config)# wireless profile policy policy-iot
   ```

   Configures the default policy profile.

3. Configure AAA override to apply policies coming from the AAA server

Example

```
Device(config-wireless-policy)# aaa-override
```

Configures AAA override to apply policies coming from the AAA server or ISE, the Cisco Identity Services Engine server.

Once you apply the policy profile, configure the WLAN with the specified policy settings to enable the expected network management and security features.

What to do next

You should verify the applied configuration by checking the WLAN settings to ensure all policies are correctly enforced.

- Confirm connectivity and performance with the new settings.

Verify a private PSK

Verify a private PSK.

To verify the configuration of a WLAN and a client, use the following **show** commands

Verify wireless client configuration

```
Device# show wlan id 2


WLAN Profile Name     : test_ppsk
================================================
Identifier                                       : 2
Network Name (SSID)                              : test_ppsk
Status                                           : Enabled
Broadcast SSID                                   : Enabled
Universal AP Admin                               : Disabled
Max Associated Clients per WLAN                  : 0
Max Associated Clients per AP per WLAN           : 0
Max Associated Clients per AP Radio per WLAN     : 0
Number of Active Clients                         : 0
Exclusionlist Timeout                            : 60
CHD per WLAN                                     : Enabled
Interface                                        : default
Multicast Interface                              : Unconfigured
WMM                                              : Allowed
WifiDirect                                       : Invalid
Channel Scan Defer Priority:
  Priority (default)                             : 4
  Priority (default)                             : 5
  Priority (default)                             : 6
Scan Defer Time (msecs)                          : 100
Media Stream Multicast-direct                    : Disabled
CCX - AironetIe Support                          : Enabled
CCX - Diagnostics Channel Capability             : Disabled
Peer-to-Peer Blocking Action                     : Disabled
Radio Policy                                     : All
DTIM period for 802.11a radio                    : 1
DTIM period for 802.11b radio                    : 1
Local EAP Authentication                         : Disabled
Mac Filter Authorization list name               : test1
```

```
Accounting list name                              : Disabled
802.1x authentication list name                   : Disabled
Security
    802.11 Authentication                         : Open System
    Static WEP Keys                               : Disabled
    802.1X                                        : Disabled
    Wi-Fi Protected Access (WPA/WPA2)             : Enabled
        WPA (SSN IE)                              : Disabled
        WPA2 (RSN IE)                             : Enabled
            TKIP Cipher                           : Disabled
            AES Cipher                            : Enabled
        Auth Key Management
            802.1x                                : Disabled
            PSK                                   : Enabled
            CCKM                                  : Disabled
            FT dot1x                              : Disabled
            FT PSK                                : Disabled
            PMF dot1x                             : Disabled
            PMF PSK                               : Disabled
    CCKM TSF Tolerance                            : 1000
    FT Support                                    : Disabled
        FT Reassociation Timeout                  : 20
        FT Over-The-DS mode                       : Enabled
    PMF Support                                   : Disabled
        PMF Association Comeback Timeout          : 1
        PMF SA Query Time                         : 200
    Web Based Authentication                      : Disabled
    Conditional Web Redirect                      : Disabled
    Splash-Page Web Redirect                      : Disabled
    Webauth On-mac-filter Failure                 : Disabled
    Webauth Authentication List Name              : Disabled
    Webauth Parameter Map                         : Disabled
    Tkip MIC Countermeasure Hold-down Timer       : 60
Call Snooping                                     : Disabled
Passive Client                                    : Disabled
Non Cisco WGB                                     : Disabled
Band Select                                       : Disabled
Load Balancing                                    : Disabled
Multicast Buffer                                  : Disabled
Multicast Buffer Size                             : 0
IP Source Guard                                   : Disabled
Assisted-Roaming
    Neighbor List                                 : Disabled
    Prediction List                               : Disabled
    Dual Band Support                             : Disabled
IEEE 802.11v parameters
    Directed Multicast Service                    : Disabled
    BSS Max Idle                                  : Disabled
        Protected Mode                            : Disabled
    Traffic Filtering Service                     : Disabled
    BSS Transition                                : Enabled
        Disassociation Imminent                   : Disabled
            Optimised Roaming Timer               : 40
            Timer                                 : 200
    WNM Sleep Mode                                : Disabled
802.11ac MU-MIMO                                  : Disabled
```

Commands for PSK verification

```
Device# show wireless client mac-address a886.adb2.05f9 detail


Client MAC Address : a886.adb2.05f9
Client IPv4 Address : 9.9.58.246
Client Username : A8-86-AD-B2-05-F9
AP MAC Address : c025.5c55.e400
AP Name: saurabh-3600
AP slot : 1
Client State : Associated
Policy Profile : default-policy-profile
Flex Profile : default-flex-profile
Wireless LAN Id : 6
Wireless LAN Name: SSS_PPSK
BSSID : c025.5c55.e40f
Connected For : 280 seconds
Protocol : 802.11n - 5 GHz
Channel : 60
Client IIF-ID : 0xa0000001
Association Id : 1
Authentication Algorithm : Open System
Client CCX version : No CCX support
Session Timeout : 320 sec (Remaining time: 40 sec)
Input Policy Name  :
Input Policy State : None
Input Policy Source : None
Output Policy Name  :
Output Policy State : None
Output Policy Source : None
WMM Support : Enabled
U-APSD Support : Enabled
  U-APSD value : 0
  APSD ACs    : BK, BE, VI, VO
Fastlane Support : Disabled
Power Save : OFF
Current Rate : m22
Supported Rates : 9.0,18.0,36.0,48.0,54.0
Mobility:
  Move Count                  : 0
  Mobility Role               : Local
  Mobility Roam Type          : None
  Mobility Complete Timestamp : 09/27/2017 16:32:25 IST
Policy Manager State: Run
NPU Fast Fast Notified : No
Last Policy Manager State : IP Learn Complete
Client Entry Create Time : 280 seconds
Policy Type : WPA2
Encryption Cipher : CCMP (AES)
Authentication Key Management : PSK
AAA override passphrase: Yes
Management Frame Protection : No
Protected Management Frame - 802.11w : No
EAP Type : Not Applicable
VLAN : 58
Access VLAN : 58
Anchor VLAN : 0
WFD capable : No
Manged WFD capable : No
Cross Connection capable : No
Support Concurrent Operation : No
```

```
Session Manager:
  Interface        : capwap_90000005
  IIF ID           : 0x90000005
  Device Type      : Apple-Device
  Protocol Map     : 0x000001
  Authorized       : TRUE
  Session timeout  : 320
  Common Session ID: 1F3809090000005DC30088EA
  Acct Session ID  : 0x00000000
  Auth Method Status List
        Method : MAB
                 SM State        : TERMINATE
                 Authen Status   : Success
  Local Policies:
        Service Template : wlan_svc_default-policy-profile (priority 254)
                 Absolute-Timer   : 320
                 VLAN             : 58
  Server Policies:
  Resultant Policies:
                 VLAN             : 58
                 Absolute-Timer   : 320
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 0
Fast BSS Transition Details :
  Reassociation Timeout : 0
11v BSS Transition : Not implemented
FlexConnect Data Switching : Local
FlexConnect Dhcp Status : Local
FlexConnect Authentication : Central
FlexConnect Central Association : No
Client Statistics:
  Number of Bytes Received : 59795
  Number of Bytes Sent : 21404
  Number of Packets Received : 518
  Number of Packets Sent : 274
  Number of EAP Id Request Msg Timeouts :
  Number of EAP Request Msg Timeouts :
  Number of EAP Key Msg Timeouts :
  Number of Policy Errors : 0
  Radio Signal Strength Indicator : -32 dBm
  Signal to Noise Ratio : 58 dB
Fabric status : Disabled
```

122 Multi-Preshared Key

Topics:

- Multi-preshared key
- Procedures

Multi-preshared key

Introduces the core concepts of Multi-Preshared Key, outlining operational details and supported features, and details applicable restrictions impacting deployment and authentication scenarios.

A multi-preshared key (multi-PSK) is a wireless security feature that

- Allows multiple pre-shared keys (PSKs) to be configured for a single SSID
- Enables any configured PSK to grant access to the same wireless network
- Improves network flexibility by supporting concurrent user groups or devices with different credentials.

PSK: A pre-shared key is a password or passphrase used to authenticate clients on a wireless network.

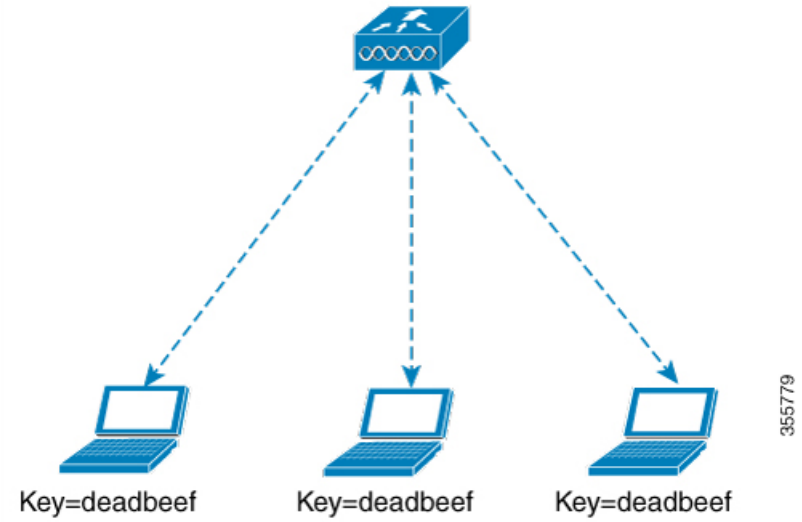
Supporting analogy: keycards to the same door

Having multi-PSk keys for a single SSID is like giving different colored keycards to various teams. Each keycard opens the same door, but you can manage which card goes to which team. This approach increases flexibility and security compared to using one generic key.

Comparing traditional PSK with

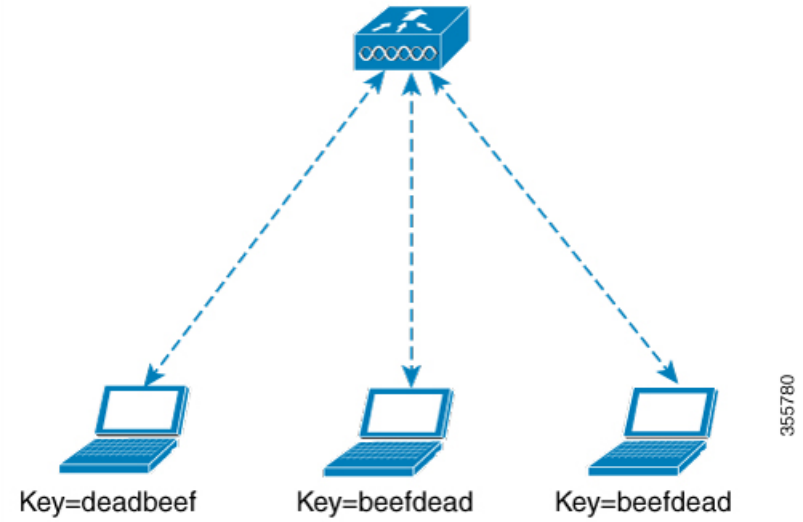
In a traditional PSK, all the clients joining the network use the same password as shown in the figure.

Figure 16: Traditional PSK



But with multi-PSK, client can use any of the configured pre-shared keys to connect to the network as shown in the figure.

Figure 17: Multi-PSK



In the Multi-PSK example, two passwords—deadbeef and beefdead—are configured for the same SSIDIn this scenario, clients can connect to the network using either of the passwords.

Note

- Each SSID in a Multi-PSK supports five PSKs.
- Multi-PSK is different from iPSK. In iPSK, the PSK password comes from ISE authorization policy, so MAB is required. Multi-PSK uses a pool of passwords locally configured in WLAN, so Identity Service Engine is not used.

Feature	Traditional PSK	Multi-PSK	Identity PSK (iPSK)
Number of PSKs per SSID	One shared key	Multiple keys (up to five)	Unique key per user or per group
Use case flexibility	All users share one credential	Separate keys for groups or devices	Per-user or per-group credentials for high granularity
Example	All staff share same key	Staff, guests, IoT devices have different keys	Each staff member, contractor, or device gets its own key
Key management	Single change affects all users	Changes can target specific groups or devices	Changes can target specific users or devices; policy-driven from ISE
Security granularity	Lowest – one compromise affects all	Better – compromise isolated to that PSK group	Highest – compromise isolated to individual user or device and fully policy-based

Restrictions

Multi-PSK restrictions.

- In central authentication flex mode, the standalone AP allows client join with the highest priority PSK (priority 0 key). New clients that do not use the highest priority PSK are rejected during the standalone mode.
- Multi-PSK does not support local authentication.
- Avoid using the percentage symbol (%) in a PSK. Some client and access point combinations may interpret the percentage symbol as a special character, which can prevent the client from joining the PSK WLAN. If clients cannot join the WLAN, replace the symbol with another supported ASCII character and retry.

Procedures

Guides configuration of Multi-Preshared Key using both GUI and CLI, and provides verification steps to ensure proper deployment and functionality.

Configure a multi-preshared key (GUI)

Configure Multi-Preshared Key using GUI.

Configure a WLAN to use a multi-PSK through the controller's GUI.

Before you begin

Know the required security settings such as Wired Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA), PSK, and so on for your WLAN.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. On the **Wireless Networks** page, click the name of the WLAN.
3. In the **Edit WLAN** window, click the **Security** tab.
4. In the **Layer2** tab, choose the **Layer2 Security Mode** from the options:
   - None: No Layer 2 security
   - 802.1X: WEP 802.1X data encryption type
   - WPA + WPA2: Wi-Fi Protected Access
   - Static WEP: Static WEP encryption parameters
   - Static WEP+802.1X: Both Static WEP and 802.1X parameters

Parameters	Description
802.1X	
WEP Key Size	Choose the key size. The available values are: • None • 40 bits • 104 bits
WPA + WPA2	

Parameters	Description
Protected Management Frame	Possible values are: • Disabled • Optional • Required
WPA Policy	Check the check box to enable WPA policy.
WPA Encryption	Choose the WPA encryption standard. A WPA encryption standard must be specified if you have enabled WPA policy.
WPA2 Policy	Check the check box to enable WPA2 policy.
WPA2 Encryption	Choose the WPA2 encryption standard. A WPA encryption standard must be specified if you have enabled WPA policy.
Auth Key Mgmt	Possible rekeying mechanism options: • 802.1X • FT + 802.1X • PSK: You must specify the PSK format and a preshared key • Cisco Centralized Key Management: You must specify a Cisco Centralized Key Management Timestamp Tolerance value • 802.1X + Cisco Centralized Key Management: You must specify a Cisco Centralized Key Management Timestamp Tolerance value • FT + 802.1X + Cisco Centralized Key Management: You must specify a Cisco Centralized Key Management Timestamp Tolerance value
Static WEP	
Key Size	Possible key size options: • 40 bits • 104 bits
Key Index	Choose a key index from 1 to 4. One unique WEP key index can be applied to each WLAN. As there are only four WEP key indexes, only four WLANs can be configured for static WEP Layer2 encryption.
Key Format	Choose the encryption key format as either ASCII or HEX.
Encryption Key	Enter an encryption key that is 13 characters long.
Static WEP + 802.1X	

Parameters	Description
Key Size	Possible key size options: • 40 bits • 104 bits
Key Index	Choose a key index from 1 to 4. One unique WEP key index can be applied to each WLAN. As there are only four WEP key indexes, only four WLANs can be configured for static WEP Layer2 encryption.
Key Format	Choose the encryption key format as either ASCII or HEX.
Encryption Key	Enter an encryption key that is 13 characters long.
WEP Key Size	Choose from the WEP key sizes: • None • 40 bits • 104 bits

5. Click **Save & Apply to Device**.

The WLAN is updated with the selected multi-PSK security settings.

Configure a multi-preshared key (CLI)

Configure Multi-Preshared Key using commands.

Configure a WLAN to use a multi-PSK through the controller's CLI.

Before you begin

Know the required security settings (WEP, WPA, PSK, and so on.) for your WLAN.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN and SSID.

   Example

   ```
   Device(config)# wlan mywlan 1 SSID_name
   ```

3. Disable security AKM for dot1x.

Example

```
Device(config-wlan)# no security wpa akm
                       dot1x
```

4. Configure PSK.

   Example

   ```
   Device(config-wlan)# security wpa akm psk
   ```

5. Configure multi-PSK.

   Example

   ```
   Device(config-wlan)# security wpa wpa2 mpsk
   ```

6. Configure PSK priority and all its related passwords using the **priority** *priority_value* **set-key** {**ascii** [**0** | **8**] *pre-shared-key* | **hex** [**0** | **8**] *pre-shared-key*} command.

   Example

   ```
   Device(config-mpsk)# priority 0 set-key ascii 0 deadbeef
   ```

   The *priority_value* ranges from 0 to 4.

   > **Note**
   >
   > You need to configure **priority 0** key for multi-PSK.

7. Enable WLAN.

   Example

   ```
   Device(config-mpsk)# no shutdown
   ```

8. Exit WLAN configuration mode and returns to configuration mode.

   Example

   ```
   Device(config-wlan)# exit
   ```

9. Return to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Example

```
Device(config)# end
```

The WLAN is updated with the selected multi-PSK security settings.

Verify multi-PSK configurations

Verify Multi-PSK Configurations.

To verify the configuration of a WLAN and a client, use the following command:

```
Device# show wlan id 8
WLAN Profile Name     : wlan_8
================================================
Identifier                                    : 8
Network Name (SSID)                           : ssid_8
Status                                        : Enabled
Broadcast SSID                                : Enabled
Universal AP Admin                            : Disabled
Max Associated Clients per WLAN               : 0
Max Associated Clients per AP per WLAN        : 0
Max Associated Clients per AP Radio per WLAN  : 200
Number of Active Clients                      : 0
CHD per WLAN                                  : Enabled
Multicast Interface                           : Unconfigured
WMM                                           : Allowed
WifiDirect                                    : Invalid
Channel Scan Defer Priority:
  Priority (default)                          : 5
  Priority (default)                          : 6
Scan Defer Time (msecs)                       : 100
Media Stream Multicast-direct                 : Disabled
CCX - AironetIe Support                       : Enabled
CCX - Diagnostics Channel Capability          : Disabled
Peer-to-Peer Blocking Action                  : Disabled
Radio Policy                                  : All
DTIM period for 802.11a radio                 : 1
DTIM period for 802.11b radio                 : 1
Local EAP Authentication                      : Disabled
Mac Filter Authorization list name            : Disabled
Mac Filter Override Authorization list name   : Disabled
Accounting list name                          :
802.1x authentication list name               : Disabled
802.1x authorization list name                : Disabled
Security
    802.11 Authentication                     : Open System
    Static WEP Keys                           : Disabled
    802.1X                                    : Disabled
    Wi-Fi Protected Access (WPA/WPA2/WPA3)    : Enabled
        WPA (SSN IE)                          : Disabled
        WPA2 (RSN IE)                         : Enabled
            MPSK                              : Enabled
            AES Cipher                        : Enabled
            CCMP256 Cipher                    : Disabled
            GCMP128 Cipher                    : Disabled
            GCMP256 Cipher                    : Disabled
        WPA3 (WPA3 IE)                        : Disabled
        Auth Key Management
```

```
            802.1x                                  : Disabled
            PSK                                     : Enabled
            CCKM                                    : Disabled
            FT dot1x                                : Disabled
            FT PSK                                  : Disabled
            FT SAE                                  : Disabled
            PMF dot1x                               : Disabled
            PMF PSK                                 : Disabled
            SAE                                     : Disabled
            OWE                                     : Disabled
            SUITEB-1X                               : Disabled
            SUITEB192-1X                            : Disabled
    CCKM TSF Tolerance                              : 1000
    FT Support                                      : Adaptive
        FT Reassociation Timeout                    : 20
        FT Over-The-DS mode                         : Enabled
    PMF Support                                     : Disabled
        PMF Association Comeback Timeout            : 1
        PMF SA Query Time                           : 200
    Web Based Authentication                        : Disabled
    Conditional Web Redirect                        : Disabled
    Splash-Page Web Redirect                        : Disabled
    Webauth On-mac-filter Failure                   : Disabled
    Webauth Authentication List Name                : Disabled
    Webauth Authorization List Name                 : Disabled
    Webauth Parameter Map                           : Disabled
    Tkip MIC Countermeasure Hold-down Timer         : 60
Non Cisco WGB                                       : Disabled
Band Select                                         : Enabled
Load Balancing                                      : Disabled
Multicast Buffer                                    : Disabled
Multicast Buffer Size                               : 0
IP Source Guard                                     : Disabled
Assisted-Roaming
    Neighbor List                                   : Disabled
    Prediction List                                 : Disabled
    Dual Band Support                               : Disabled
IEEE 802.11v parameters
    Directed Multicast Service                      : Disabled
    BSS Max Idle                                    : Disabled
        Protected Mode                              : Disabled
    Traffic Filtering Service                       : Disabled
    BSS Transition                                  : Enabled
        Disassociation Imminent                     : Disabled
            Optimised Roaming Timer                 : 40
            Timer                                   : 200
    WNM Sleep Mode                                  : Disabled
802.11ac MU-MIMO                                    : Disabled
802.11ax paramters
    OFDMA Downlink                                  : unknown
    OFDMA Uplink                                    : unknown
    MU-MIMO Downlink                                : unknown
    MU-MIMO Uplink                                  : unknown
    BSS Color                                       : unknown
    Partial BSS Color                               : unknown
    BSS Color Code                                  :
```

To view the WLAN details, use the following command:

```
Device# show run wlan
wlan wlan_8 8 ssid_8
 security wpa psk set-key ascii 0 deadbeef
```

```
 no security wpa akm dot1x
 security wpa akm psk
 security wpa wpa2 mpsk
  priority 0 set-key ascii 0 deadbeef
  priority 1 set-key ascii 0 deaddead
  priority 2 set-key ascii 0 d123d123
  priority 3 set-key hex 0
02345678901234567890123456789012345678901234567890123456789012345678901234
  priority 4 set-key hex 0
12345678901234567890123456789012345678901234567890123456789012345678901234
 no shutdown
```

123 Wireless AAA Authentication Survivability Cache

Topics:

- Wireless AAA authentication survivability cache

Wireless AAA authentication survivability cache

Describes the wireless AAA authentication survivability cache feature.

The Wireless AAA authentication survivability cache feature enhances the reliability of wireless client authentication by storing successful authentication results locally on the controller. This cache includes details such as the client's MAC address, user name, hashed password, and Attribute-Value Pairs (AVPs) received from the RADIUS server.

Wireless AAA authentication survivability cache is a mechanism implemented in the controller that:

- provides a fallback methodology in wireless systems for AAA authentication servers,
- implements authentication and policy caching mechanism, and
- leverages the caching capability of the AAA module of System Management Daemon (SMD) in WNCd.

This feature is supported in both local mode and FlexConnect central authentication mode.

Wireless AAA authentication survivability cache as failover mechanism

If the AAA server becomes unreachable, for example, due to a network outage or server failure, the controller uses the cached credentials to authenticate clients. This ensures that the previously authenticated clients are allowed network access, even if the primary authentication source is unavailable.

Use case for wireless AAA authentication survivability cache

Some use cases for the wireless AAA authentication survivability cache feature are:

- Cache can be used when the controller loses connectivity with the AAA server, which may disconnect authenticated clients.
- Cache can be used as the primary source of authentication, with fallback to AAA, to avoid slow authentication time due to WAN latency.

Benefits of using AAA cache for failover

The benefits of using AAA cache for failover are:

- Seamless client access: If the AAA server fails, previously authenticated clients can still access the network without delays or disruptions.
- Reduced dependency on AAA server availability: This mechanism reduces the dependency on the availability of the AAA server for re-authentication, providing business continuity even if there is an issue with the server.
- Improved user experience: Clients do not experience interruptions when reconnecting, as the controller uses the cached authentication information to grant them access immediately.
- Load reduction on AAA server: During network failures or outages, the controller does not need to query the AAA server repeatedly for authentication. This helps reduce the load on the AAA server.
- Failover for high availability: If the primary AAA server temporarily fails, such as during maintenance, the AAA cache maintains client access automatically.

Feature history for wireless AAA authentication survivability cache

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 115: Feature history for wireless AAA authentication survivability cache

Release	Feature Information
Cisco IOS XE 17.18.1	The Wireless AAA authentication survivability cache feature enhances the reliability of wireless client authentication by storing successful authentication results locally on the controller. Cache can be used: • When the controller loses connectivity with the AAA server, which may disconnect authenticated clients. • As the primary source of authentication, with fallback to AAA, to avoid slow authentication time due to WAN latency. This feature is supported in both local mode and FlexConnect central authentication mode.

Configure wireless AAA authentication survivability cache

This section describes the steps to configure wireless AAA authentication survivability cache using CLI commands.

Before you begin

Ensure that the following are met:

- SSID already configured
- A RADIUS server such as Cisco ISE with authentication and authorization policy and profiles are created

Configure a cache profile and apply the cache profile to the AAA server group used for authentication and authorization.

Procedure

1. Enter configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a profile for the local RADIUS server.

   Example

   ```
   Device# aaa server radius dynamic-author
   ```

   This command creates the profile for local RADIUS server and enters into the local server dynamic author RADIUS configuration.

3. Configure a RADIUS client.

Example

```
Device(config-locsvr-da-radius)# client {A.B.C.D | X:X:X:X::X} server-key {{0
 | 6 | 7}server-key | server-key2}
```

This command specifies the IPv4 or the IPv6 address of the RADIUS client and the RADIUS client server key.

4. Specify the server authorization type.

Example

```
Device(config-locsvr-da-radius)# auth-type {all | any | sessionkey}
```

5. Exit the local server dynamic author configuration mode.

Example

```
Device(config-locsvr-da-radius)# exit
```

Configure RADIUS dynamic-author for CoA

Follow these steps to configure RADIUS dynamic-author for CoA:

Procedure

1. Enter configuration mode.

Example

```
Device# configure terminal
```

2. Configure a profile for the local RADIUS server.

Example

```
Device# aaa server radius dynamic-author
```

This command creates the profile for local RADIUS server and enters into the local server dynamic author RADIUS configuration.

3. Configure a RADIUS client.

Example

```
Device(config-locsvr-da-radius)# client {A.B.C.D | X:X:X:X::X} server-key {{0
 | 6 | 7}server-key | server-key2}
```

This command specifies the IPv4 or the IPv6 address of the RADIUS client and the RADIUS client server key.

4. Specify the server authorization type.

Example

```
Device(config-locsvr-da-radius)# auth-type {all | any | sessionkey}
```

5. Exit the local server dynamic author configuration mode.

   Example

   ```
   Device(config-locsvr-da-radius)# exit
   ```

The RADIUS client is configured.

Example

```
Device# configure terminal
Device# aaa server radius dynamic-author
Device(config-locsvr-da-radius)# client 10.10.10.1 server-key 0 server-key |
server-key2}
Device(config-locsvr-da-radius)# auth-type any
Device(config-locsvr-da-radius)# exit
```

What to do next

Configure the RADIUS server.

Configure RADIUS server

Procedure

1. Enter configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the RADIUS server.

   Example

   ```
   Device(config)# radius server radius-server-name
   ```

   This command configures the RADIUS server and enters the RADIUS server configuration mode.

3. Configure the RADIUS server address.

   Example

   ```
   Device(config-radius-server)# address ipv4 hostname or A.B.C.D [auth-port |
   alias | acct-port] <0-65534> [acct-port] <0-65534>
   ```

   This command configures the RADIUS server address and the UDP ports for the RADIUS authentication server and accounting server.

   The default port number for the RADIUS accounting server is 1813 and the default port number for the RADIUS authentication server is 1814.

4. Configure the per server encryption key.

Example

```
Device(config-radius-server)# key key
```

The RADIUS server is configured.

Example

```
Device# configure terminal
Device(config)# radius server radius-server-name
Device(config-radius-server)# address ipv4 hostname auth-port 1645 acct-port
1646
Device(config-radius-server)# key key
```

What to do next

Configure the RADIUS server group and cache expiry.

Configure RADIUS server group and cache expiry

Before you begin

> **Note**
>
> Only one AAA server group can be configured for AAA cache feature.

Procedure

1. Enter configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the RADIUS server group definition.

   Example

   ```
   Device(config)# aaa group server {ldap | radius | tacacs+} server-group-name
   ```

   This command configures the server group name with 32 as the maximum string length. Here, configure the RADIUS server group name.

3. Configure the RADIUS server name.

   Example

   ```
   Device(config-sg-radius)# server {hostmame or A.B.C.D | name} server-name
   ```

4. Configure the cache authorization profile.

Example

```
Device(config-sg-radius)# cache authorization profile cache-auth
```

5. Configure the cache authentication profile.

Example

```
Device(config-sg-radius)# cache authentication profile cache-auth
```

6. Configure the cache expiry.

Example

```
Device(config-sg-radius)# cache expiry 0-2147483647
```

This command configures the cache expiry time in hours. The default is 24 hours. The value 0 stands for never expire.

> **Note**
>
> If the cache entry expires and the AAA server is still unreachable, the client is not authenticated until the server is reachable again.

7. Exit the RADIUS server configuration mode.

Example

```
Device(config-sg-radius)# exit
```

The RADIUS server group and cache profile is configured.

Example

```
Device# configure terminal
Device(config)# aaa group server radius radius-group
Device(config-sg-radius)# server name server-name
Device(config-radius-server)# cache authorization profile cache-auth
Device(config-sg-radius)# cache authentication profile cache-auth
Device(config-sg-radius)# cache expiry 24
Device(config-sg-radius)# exit
```

What to do next

Configure AAA for network access using a cache profile.

Configure AAA cache profile

Procedure

---

1. Enter configuration mode.

Example

```
Device# configure terminal
```

2. Configure the AAA cache profile.

Example

```
Device(config)# aaa cache profile aaa-cache
```

3. Cache all entries.

Example

```
Device(config-profile-map)# all
```

The cache profile is configured.

Example

```
Device# configure terminal
Device(config)# aaa cache profile aaa-cache
Device(config-profile-map)# all
```

What to do next

Configure EAP profile.

Configure EAP profile

Procedure

---

1. Enter configuration mode.

Example

```
Device# configure terminal
```

2. Configure EAP profile.

Example

```
Device(config)# eap profile eap-profile
```

3. Configure EAP-TLS allowed method.

Example

```
Device(config-eap-profile)# method tls
```

4. Configure the default PKI trustpoint.

Example

```
Device(config-eap-profile)# pki-trustpoint pki_trustpoint_name
```

Here, you must specify the name of the PKI trustpoint.

The EAP profile is configured.

Example

```
Device# configure terminal
Device(config)# eap profile eap-profile
Device(config-eap-profile)# method tls
Device(config-eap-profile)# pki-trustpoint default-pki-trustpoint
```

What to do next

Configure WLAN.

Configure WLAN

Procedure

---

1. Enter configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a WLAN profile.

   Example

   ```
   Device(config)# wlan wlan-profile 1-4096 ssid-name
   ```

   This command configures a WLAN profile and enters the WLAN configuration mode.

3. Enable local auth EAP profile.

   Example

   ```
   Device(config-wlan)# local-auth eap-profile
   ```

4. Enable security authentication list for dot1x security.

   Example

   ```
   Device(config-wlan)# security dot1x authentication-list default
   ```

5. Enable security authorization list for dot1x security.

Example

```
Device(config-wlan)# security dot1x authorization-list default
```

6. Enable WLAN.

Example

```
Device(config-wlan)# no shutdown
```

A WLAN profile is configured.

Example

```
Device# configure terminal
Device(config)# wlan wlan-profile 2 ssid-name
Device(config-wlan)# security dot1x authentication-list default
Device(config-wlan)# security dot1x authorization-list default
Device(config-wlan)# no shutdown
```

AAA dead-server detection

This topic explains AAA dead-server detection.

AAA dead-server detection allows you to configure the criteria to be used to mark a RADIUS server as dead. If you have more than one RADIUS server, the following concepts come into picture:

- Deadtime: Defines the time in minutes a server marked as DEAD is held in that state. Once the deadtime expires, the controller marks the server as UP (ALIVE) and notifies the registered clients about the state change. If the server is still unreachable after the state is marked as UP and if the DEAD criteria is met, then server is marked as DEAD again for the deadtime interval.

  For example, `Device(config)# radius-server deadtime 5`.

  > **Note**
  >
  > You can configure deadtime for each server group or on a global level.

- Dead-criteria–To declare a server as DEAD, you need to configure dead-criteria and configure the conditions that determine when a RADIUS server is considered unavailable or dead.

  For example, `Device(config)# radius-server dead-criteria time 5 tries 5`.

Using AAA dead-server detection will result in less deadtime and quicker packet processing.

For more information, see the chapter AAA Dead-Server Detection.

RADIUS configuration for MSCHAPv2

Configure AV-pair attribute on RADIUS

Associate the following Cisco AV-pairs with the authorization profile associated with the client, on the ISE server:

- cisco-av-pair = AS-Username=Cisco

- cisco-av-pair = AS-Credential-Hash=F2E787D376CBF6D6DD3600132E9C215D

> **Note**
>
> Every user must configure the AV-pair attribute on RADIUS.

The Password or AS-Credential-Hash should be in the NT-hash format (https://codebeautify.org/ntlm-hash-generator).

Configure EAP-TLS on ISE

Describes the EAP-TLS configuration on ISE.

1. Generate a CSR and the import certificate into the controller (PCKS 12 or .pkg would be convenient to import as a single file).
2. Configure a PKI trustpoint in the EAP profile as described in .

Verify wireless AAA authentication survivability cache

Verify the wireless AAA authentication survivability using the show commands.

Verify configured cache entries

To verify the configured cache entries, run the **show aaa cache group** command.

> **Note**
>
> This command is extended to show cache entries for the Wireless Network Controller Daemon (WNCD) process.

```
Device# show aaa cache group aaa-cache-grp all
-----------------------------------------------------------
IOSD AAA Auth Cache entries:
Entries in Profile dB aaa-cache-grp for exact match:
No entries found in Profile dB
-----------------------------------------------------------
SMD AAA Auth Cache entries:
Total number of Cache entries is 0
WNCD AAA Auth Cache entries:
MAC ADDR: 0000.AAAA.BBBB
Profile Name: NEWCACHE
User Name: user1
Timeout: 3600
Created Timestamp: 03/12/25 17:11:23 UTC
Server IP Address: 1.1.1.1
MAC ADDR: AAAA.BBBB.CCCC
Profile Name: NEWCACHE
User Name: user2
Timeout: 3600
Created Timestamp: 03/12/25 17:04:59 UTC
Server IP Address: 1.1.1.1
```

```
Total number of Cache entries is 2
------------------------------------------------------------
```

124 Wi-Fi Protected Access 3

Topics:

- Overview of WPA3
- WPA3 Configuration Tasks

Overview of WPA3

Outlines WPA3 fundamentals, describing Simultaneous Authentication of Equals, Opportunistic Wireless Encryption, Hash-to-Element, YANG (RFC model), Transition Disable, and WPA3 SAE iPSK, and explains the core technologies that strengthen Wi-Fi security and user protections.

Simultaneous Authentication of Equals

A simultaneous authentication of equals (SAE) is a protocol used in WPA3 that

- provides stronger password protection from guessing attacks by third parties
- employs discrete logarithm cryptography to perform an efficient exchange that enables mutual authentication using a password, and
- resists offline dictionary attacks.

An offline dictionary attack is where an adversary attempts to determine a network password by trying possible passwords without further network interaction

Feature History

Feature Name	Release	Description
WPA3 – SAE H2E with Identity PSK	Cisco IOS XE 17.9.2	Added support for Identity PSK (iPSK) passphrase for SAE H2E authentication in local mode. iPSK replaces WLAN passphrase during SAE H2E authentication when configured.
Wi-Fi Protected Access 3 Hash-to-Element (H2E) Support for SAE Authentication	Cisco IOS XE 17.7.1	Introduced Hash-to-Element (H2E) support for SAE authentication. WLAN command options added: **h2e**, **hnp**, **both-h2e-hnp** (default).
Wi-Fi Protected Access3	Cisco IOS XE 16.12.1	WPA3 is the latest version of Wi-Fi ProtectedAccess (WPA), which is a suite of protocols and technologies that provide authentication and encryption for Wi-Fi networks.

WPA3 and SAE in Wi-Fi Security

WPA3 is the latest version of Wi-Fi Protected Access (WPA), which is a suite of protocols and technologies that provide authentication and encryption for Wi-Fi networks.

WPA3 leverages SAE to provide stronger protections for users against password guessing attempts by third parties.

- WPA3-Personal protects users through robust password-based authentication, making brute-force dictionary attacks more difficult.
- WPA3-Enterprise provides higher-grade security protocols for sensitive data networks.

When the client connects to the AP, they perform an SAE exchange. If the exchange is successful, both parties create a cryptographically strong key, which is used to derive the session key. The client and AP complete commit and confirm phases. After commitment, the devices transition to confirm states whenever a new session key is generated. This method uses forward secrecy so that if an intruder cracks one key, the other session keys remain secure.

> **Note**
>
> Home SSIDs configured using the OEAP GUI do not support WPA3 security in Cisco IOS-XE 17.6 and 17.7 releases.

Unsupported APs

Cisco APs do not support SAE. If you attempt to connect an AP client to an SAE SSID using these APs, the client will not be able to join after receiving M3 from the AP.

Cisco APs that do not support SAE include:

- Cisco Aironet 1815 Series APs (AP1815W, AP1815T, AP1815I, AP1815M)
- Cisco Aironet 1815T OfficeExtend APs
- Cisco Aironet 1800 Series APs (AP1800I, AP1800S)
- Cisco Aironet 1810 Series APs
- Cisco Aironet 1542 Series Outdoor APs (AP1542D, AP1542I)
- Cisco Aironet 1830 and 1850 Series APs
- Cisco Aironet 1840 Series APs (AP1840I)

> **Note**
>
> For more information on supported APs for SAE, refer to the Feature Matrix for Cisco Wireless Access Points.

Opportunistic Wireless Encryption

Opportunistic Wireless Encryption (OWE) is a wireless security protocol that

- serves as an extension to IEEE 802.11 standards
- provides encryption of the wireless medium without requiring user credentials, and
- enables secure wireless communication between APs and clients, even in networks traditionally considered open.

Expanded explanation

OWE authenticates devices using Diffie-Hellman algorithm-based cryptography to set up wireless encryption. During network access, the client and AP perform a Diffie-Hellman key exchange. This process establishes a unique pairwise secret. The secret is then used with the standard 4-way handshake to secure the session. This method reduces the risks of open wireless networks by encrypting data traffic and increases security for deployments that otherwise use unsecured or pre-shared key(PSK) configurations.

An airport Wi-Fi network that previously offered open access (no password required) can implement OWE to encrypt wireless communications, protecting user data from eavesdropping while still avoiding the need for users to enter a password.

A traditional open wireless network (without OWE) allows any nearby device to connect and intercept wireless transmissions, leaving users vulnerable to data theft.

Hash-to-Element (H2E)

Hash-to-Element (H2E) is a new SAE Password Element method.

An SAE password element method is a cryptographic procedure that

- Derives the secret password element (PWE) in the Simultaneous Authentication of Equals (SAE) protocol
- Transforms a password into a secure intermediary element for authentication, and
- Provides protection against security threats such as group downgrade attacks.

Hash-to-Element is a new SAE Password Element (PWE) calculation method for wireless security protocols.

- **Hash-to-Element (H2E)**: Hash-to-Element is a SAE password element method that derives the PWE directly from a password using a hashing process. When a station (STA) supporting H2E initiates SAE with an AP, it checks if the AP supports H2E. If supported, the AP uses the H2E method to generate the PWE, as indicated by a new Status Code in the SAE Commit message.

PWE derivation steps in H2E

When using the H2E method, PWE derivation consists of two steps:

1. Creation of secret intermediary element (PT): The secret intermediary element (PT) is derived offline from the password when it is initially configured for each supported group.
2. Derivation of the PWE: The PWE is generated in real-time from the stored PT during the SAE exchange, using the negotiated group and the Media Access Control (MAC) addresses of both peers.

Additional reference information

- If a device uses the older Hunting-and-Pecking method, the overall SAE exchange remains unchanged.
- The H2E method protects against man-in-the-middle attacks, including Group Downgrade attacks. During authentication, peers exchange lists of rejected groups as part of PMK derivation. If there is a mismatch, the system terminates authentication to prevent a downgrade attack.
- The 6-GHz band supports only H2E SAE PWE method.
- In a typical SAE exchange on a 6 GHz network, only the H2E method is used for password element generation.
- A station using H2E checks AP support, and if supported, uses H2E for secure and efficient authentication.

YANG (RPC model)

Steps to create an RPC for SAE Password Element mode.

To create an RPC for SAE Password Element (PWE) mode, use this RPC model:

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
message-id="urn:uuid:0a77124f-c563-469d-bd21-cc625a9691cc">
<nc:edit-config>
<nc:target>
<nc:running/>
</nc:target>
<nc:config>
<wlan-cfg-data xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-wlan-cfg">
<wlan-cfg-entries>
<wlan-cfg-entry>
```

```
<profile-name>test</profile-name>
<wlan-id>2</wlan-id>
<sae-pwe-mode>both-h2e-hnp</sae-pwe-mode>
</wlan-cfg-entry>
</wlan-cfg-entries>
</wlan-cfg-data>
</nc:config>
</nc:edit-config>
</nc:rpc>
```

To delete a 6-GHz radio policy and modify the SAE Password Element (PWE) mode, use this RPC model:

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
message-id="urn:uuid:2b8c4be6-492e-4488-b2cf-1f2a1e39fa8c"><nc:edit-config>
<nc:target>
<nc:running/>
</nc:target>
<nc:config>
<wlan-cfg-data xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-wlan-cfg">
<wlan-cfg-entries>
<wlan-cfg-entry>
<profile-name>test</profile-name>
<wlan-id>2</wlan-id>
<wlan-radio-policies>
<wlan-radio-policy nc:operation="delete">
<band>dot11-6-ghz-band</band>
</wlan-radio-policy>
</wlan-radio-policies>
</wlan-cfg-entry>
</wlan-cfg-entries>
</wlan-cfg-data>
</nc:config>
</nc:edit-config>
</nc:rpc>

##
Received message from host
<?xml version="1.0" ?>
<rpc-reply message-id="urn:uuid:2b8c4be6-492e-4488-b2cf-1f2a1e39fa8c"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
NETCONF rpc COMPLETE
NETCONF SEND rpc
Requesting 'Dispatch'
Sending:

#1268
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
message-id="urn:uuid:e19a3309-2509-446f-9dbe-c46a6de433db"><nc:edit-config>
<nc:target>
<nc:running/>
</nc:target>
<nc:config>
<wlan-cfg-data xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-wireless-wlan-cfg">
<wlan-cfg-entries>
<wlan-cfg-entry>
<profile-name>test</profile-name>
<wlan-id>2</wlan-id>
<wlan-radio-policies>
```

```
<wlan-radio-policy nc:operation="merge">
<band>dot11-5-ghz-band</band>
</wlan-radio-policy>
</wlan-radio-policies>
<sae-pwe-mode>hunting-and-pecking-only</sae-pwe-mode>
</wlan-cfg-entry>
</wlan-cfg-entries>
</wlan-cfg-data>
</nc:config>
</nc:edit-config>
</nc:rpc>

##
Received message from host
<?xml version="1.0" ?>
<rpc-reply message-id="urn:uuid:e19a3309-2509-446f-9dbe-c46a6de433db"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
<ok/>
</rpc-reply>
NETCONF rpc COMPLETE
```

Note

The **delete** operation performs one action at a time due to the current infra limitation. That is, in YANG module, the **delete** operation on multiple nodes are not supported.

Transition Disable

Disable few transition modes for subsequent connections to the APs network.

A Transition Disable indication is a wireless security feature that

- allows an AP to signal a client device (STA) to disable certain transition modes for future connections
- helps prevent downgrade attacks by restricting less secure connection options, and
- is used to strengthen network security without affecting legacy device connectivity.

Disabling transition modes to enforce WPA3 Security.

A client device (STA) may enable transition modes—such as WPA3-Personal transition mode—by default in its network profile, sometimes allowing fallback to less secure methods like PSK.

The Transition Disable indication lets the AP instruct supported STAs to disable these fallback modes, ensuring only secure connection methods are used when possible.

The Transition Disable indication thus provides protection against downgrade attacks.

> **Note**
>
> An AP that uses Transition Disable indication does not necessarily disable the corresponding transition modes on its own BSS.

The APs in WPA3-Personal network might use the Transition Disable indication to ensure that all STAs supporting WPA3-Personal are protected against the downgrade attack. However, the WPA3-Personal transition mode is enabled on the BSS for the legacy STAs to connect.

WPA3 SAE iPSK

A RADIUS server and Identity PSK that preshares keys for individuals or a group of users in the same SSID.

A WPA3 SAE iPSK is a Wi-Fi authentication method that:

- uses RADIUS server integration to generate unique pre-shared keys for individual users or groups,
- allows more secure and granular authentication in networks where devices do not support 802.1X, and
- limits breach impact so only compromised keys require updating, not all devices on the SSID.
- **SAE (Simultaneous Authentication of Equals)**: A protocol for mutual secure authentication between client and access point.
- **H2E (Hash-to-Element)**: An SAE mode using a password token derived from the authentication passphrase.

iPSK passphrase handling

In networks using WPA3 SAE iPSKs, the WLAN appears like a traditional PSK network from the client's perspective.

From Cisco IOS-XE 17.9.2, the iPSK passphrase is supported for SAE H2E authentication in Local mode. The iPSK passphrase is configured in the client authorization policy in the RADIUS server. The passphrase pushes the policy to the controller during client MAB authentication.

> **Note**
>
> The iPSK passphrase in the RADIUS policy replaces the WLAN profile passphrase when generating the password token.

If only one group's PSK is compromised, that group must update their key; the rest of the WLAN remains secure.

WPA3 Configuration Tasks

Guides configuration of WPA3 features, including SAE in mixed mode, WPA3 Enterprise and OWE via GUI and CLI, WPA3 OWE transition mode, SAE iPSK, SAE H2E, and Transition Disable, and provides verification steps for ensuring proper WPA3 operation and advanced security settings.

Configure SAE (WPA3+WPA2 mixed mode)

Enable secure Wi-Fi connectivity using both WPA3 and WPA2 protocols in a mixed mode for SAE on the device.

Performing this task configures a WLAN to support both legacy WPA2 and newer WPA3 authentication using SAE, enhancing security and compatibility for a range of client devices.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id SSID-name
   ```

   Example

   ```
   Device(config)# wlan WPA3 1 WPA3
   ```

3. Disable security AKM for 802.1X.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

4. Disable fast transition over the data source on the WLAN.

   Example

   ```
   Device(config-wlan)# no security ft over-the-ds
   ```

5. Disable 802.11r fast transition on the WLAN.

   Example

   ```
   Device(config-wlan)# no security ft
   ```

6. Configure WPA2 cipher.

   Example

   ```
   Device(config-wlan)# security wpa wpa2 ciphers aes
   ```

> **Note**
>
> You can check whether cipher is configured using **no security wpa wpa2 ciphers aes**command. If cipher is not reset, configure the cipher.

7. Specify a preshared key.

   Example

   ```
   Device(config-wlan)# security wpa psk set-key ascii value preshared-key
   ```

   Example

   ```
   Device(config-wlan)# security wpa psk set-key ascii 0 Cisco123
   ```

8. Enable WPA3 support.

   Example

   ```
   Device(config-wlan)# security wpa wpa3
   ```

   > **Note**
   >
   > If both WPA2 and WPA3 are supported (SAE and PSK together), it is optional to configure PMF. However, you cannot disable PMF. For WPA3, PMF is mandatory.

9. Enable AKM SAE support.

   Example

   ```
   Device(config-wlan)# security wpa akm sae
   ```

10. Enable AKM PSK support.

    Example

    ```
    Device(config-wlan)# security wpa akm psk
    ```

11. Enable the WLAN.

    Example

    ```
    Device(config-wlan)# no shutdown
    ```

12. Return to the privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

The WLAN is configured to support both WPA3 (with SAE) and WPA2 authentication modes for client connectivity.

Configure WPA3 enterprise (GUI)

Enable WPA3 enterprise authentication on your WLAN to secure wireless communications.

Use this task when you need to enable WPA3 enterprise security for a WLAN profile on your wireless controller using the GUI.

Before you begin

Ensure necessary RADIUS or AAA servers and authentication lists are configured.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**, the **SSID** and the **WLAN ID**.
4. Choose **Security** > **Layer2** tab. Choose **WPA2+WPA3** in **Layer 2 Security Mode** drop-down list.
5. Uncheck the **WPA2 Policy** and **802.1x** check boxes. Check the **WPA3 Policy** and **802.1x-SHA256** check boxes.
6. Choose **Security** > **AAA**, and then choose the Authentication List from the **Authentication List** drop-down list.
7. Click **Apply to Device**.

Your newly created WLAN uses WPA3 enterprise security for client authentication.

Configure WPA3 enterprise

Set up a secure wireless LAN (WLAN) that uses WPA3 Enterprise authentication to enhance network security for enterprise environments.

WPA3 Enterprise provides stronger data protection and improved security features compared to earlier WLAN authentication methods. You can enable WPA3 Enterprise on supported APs and controllers with this configuration.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id SSID-name
   ```

3. Disable security AKM for 802.1X.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

4. Disables WPA2 security.

   Example

   ```
   Device(config-wlan)# no security wpa wpa2
   ```

5. Configures 802.1x support.

   Example

   ```
   Device(config-wlan)# security wpa akm dot1x-sha256
   ```

6. Enables WPA3 support.

   Example

   ```
   Device(config-wlan)# security wpa wpa3
   ```

7. Configure security authentication list for dot1x security.

   Example

   ```
   Device(config-wlan)# security dot1x authentication-list list-name
   ```

8. **no shutdown**

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

   Enables the WLAN.

9. **end**

   Example

   ```
   Device(config-wlan)# end
   ```

   Return to the privileged EXEC mode.

> **Note**
>
> C9115 and C9120 access points do not support WLANs configured with WPA3 Enterprise (SUITEB192-1X).

The WPA3 enterprise WLAN is configured and enabled. Devices can now securely connect using WPA3 Enterprise authentication.

Configure WPA3 OWE

Configure the WPA3 OWE using commands.

Enable WPA3 Opportunistic Wireless Encryption (OWE) on a WLAN to provide enhanced security for wireless clients.

Use this task when you need to configure a WLAN with WPA3 OWE mode using CLI.

Before you begin

- Ensure Protected Management Frames (PMF) are configured internally.
- WPA2 ciphers are valid for the cipher configuration.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id SSID-name
   ```

3. Disable fast transition over the data source on the WLAN.

   Example

   ```
   Device(config-wlan)# no security ft over-the-ds
   ```

4. Disable 802.11r fast transition on the WLAN.

   Example

   ```
   Device(config-wlan)# no security ft
   ```

5. Disable security AKM for dot1x.

Example

```
Device(config-wlan)# no security wpa akm dot1x
```

6. Disable WPA2 security.

   Example

   ```
   Device(config-wlan)# no security wpa wpa2
   ```

   This action also disables PMF.

7. Enable WPA2 ciphers for AES.

   Example

   ```
   Device(config-wlan)# security wpa wpa2 ciphers aes
   ```

   > **Note**
   >
   > WPA2 and WPA3 use common ciphers.

8. Enable WPA3 support.

   Example

   ```
   Device(config-wlan)# security wpa wpa3
   ```

9. Enable WPA3 OWE support.

   Example

   ```
   Device(config-wlan)# security wpa akm owe
   ```

10. Enable the WLAN to allow client devices to connect.

    Example

    ```
    Device(config-wlan)# no shutdown
    ```

11. Return to the privileged EXEC mode.

    Example

    ```
    Device(config-wlan)# end
    ```

You have enabled WPA3 OWE on your WLAN, enhancing security for your devices when they connect to the SSID.

Configure WPA3 OWE transition mode (GUI)

Configure WPA3 OWE Transition Mode using GUI.

Configure WPA3 OWE Transition Mode using the GUI to enable secure wireless connectivity with Opportunistic Wireless Encryption (OWE) in transition mode.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**, the **SSID** and the **WLAN ID**.
4. Choose **Security** > **Layer2** tab. Choose **WPA2+WPA3** in **Layer 2 Security Mode** drop-down list.
5. Uncheck the **WPA2 Policy**, **802.1x**, **Over the DS**, **FT + 802.1x** and **FT + PSK**check boxes.Check the **WPA3 Policy**, **AES** and **OWE** check boxes.
6. Enter the **Transition Mode WLAN ID**.
7. Click **Apply to Device**.

The WLAN is configured with WPA3 OWE Transition Mode, enabling secure connections for compatible devices.

Configure WPA3 OWE transition mode

Configure a WLAN to support WPA3 OWE transition mode for secure wireless connectivity.

WPA3 Opportunistic Wireless Encryption (OWE) transition mode allows clients to connect using either open or secured (WPA3 OWE) methods during transition deployments.

Policy validation does not take place between the open WLAN and the OWE WLAN. You must configure both WLANs correctly.

Before you begin

Identify the WLAN ID and names for the open and OWE WLANs you will configure.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id SSID-name
   ```

3. Disable security AKM for 802.1X.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

4. Disable fast transition over the data source on the WLAN.

   Example

   ```
   Device(config-wlan)# no security ft over-the-ds
   ```

5. Disable 802.11r fast transition on the WLAN.

   Example

   ```
   Device(config-wlan)# no security ft
   ```

6. Disable WPA2 security. PMF is disabled now.

   Example

   ```
   Device(config-wlan)# no security wpa wpa2
   ```

7. Enable WPA2 ciphers for AES.

   Example

   ```
   Device(config-wlan)# security wpa wpa2 ciphers aes
   ```

8. Enable WPA3 support.

   Example

   ```
   Device(config-wlan)# security wpa wpa3
   ```

9. Enable WPA3 OWE support.

   Example

   ```
   Device(config-wlan)# security wpa akm owe
   ```

10. Configure the open or OWE transition mode WLAN ID.

    Example

    ```
    Device(config-wlan)# security wpa transition-mode-wlan-id wlan-id
    ```

> **Note**
>
> Validation is not performed on the transition mode WLAN. You must configure it correctly by assigning the open WLAN identifier to the OWE WLAN, and the OWE WLAN identifier to the open WLAN configuration.
>
> Assign the OWE WLAN ID as the transition mode WLAN ID in the open WLAN configuration. Similarly, assign the open WLAN ID as the transition mode WLAN ID in the OWE WLAN configuration.

11. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

12. Return to the privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

The WLAN is configured in WPA3 OWE transition mode and is ready for client connections using the specified security settings.

Configure WPA3 SAE (GUI)

Set up a wireless LAN (WLAN) that uses WPA3 Simultaneous Authentication of Equals (SAE) security using GUI.

Perform this task to enable enhanced security for your WLAN with WPA3 SAE. Use this configuration to support WPA3 for client authentication and ensure optimal wireless security.

Before you begin

Ensure Protected Management Frames (PMF) are configured internally.

- WPA2 ciphers can be used as associated ciphers.
- Fast Transition Adaptive is not supported for WPA3 SAE.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**, the **SSID** and the **WLAN ID**.
4. Choose **Security** > **Layer2** tab. Choose **WPA2+WPA3** in **Layer 2 Security Mode** drop-down list.
5. Uncheck the **WPAPolicy**, **802.1x**, **Over the DS**, **FT + 802.1x**, and **FT + PSK** checkboxes. Check the **WPA3 Policy**, **AES**, and **PSK** checkboxes. Enter the **Pre-Shared Key**, and choose the PSK Format from the **PSK Format** drop-down list and the PSK Type from the **PSK Type** drop-down list.
6. Click **Apply to Device**.

Your WLAN profile is now configured with WPA3 SAE security, and clients can connect using WPA3 authentication.

Configure WPA3 SAE

Enable WPA3 SAE authentication on a WLAN for enhanced Wi-Fi security.

Use this task to configure WPA3 SAE (Simultaneous Authentication of Equals) on a Cisco device. WPA3 SAE offers stronger security for wireless networks and is required for environments that need improved protection against offline dictionary attacks.

Before you begin

Configure PMF (Protected Management Frames) internally.

- You can use WPA2 ciphers with this configuration.
- Fast Transition Adaptive is not supported for WPA3 SAE.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id SSID-name
   ```

   Example

   ```
   Device(config)# wlan WPA3 1 WPA3
   ```

3. Disable security AKM for dot1x.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

4. Disable fast transition over the data source on the WLAN.

   Example

   ```
   Device(config-wlan)# no security ft over-the-ds
   ```

5. Disable 802.11r fast transition on the WLAN.

   Example

   ```
   Device(config-wlan)# no security ft
   ```

6. Disable WPA2 security. PMF is disabled now.

Example

```
Device(config-wlan)# no security wpa wpa2
```

7. Configure WPA2 cipher.

Example

```
Device(config-wlan)# security wpa wpa2 ciphers aes
```

> **Note**
>
> You can check whether cipher is configured using **no security wpa wpa2 ciphers aes**command. If cipher is not reset, configure the cipher.

8. Specify a preshared key

Example

```
Device(config-wlan)# security wpa psk set-key ascii value preshared-key
```

Example

```
Device(config-wlan)# security wpa psk set-key ascii 0 Cisco123
```

9. Enable WPA3 support.

Example

```
Device(config-wlan)# security wpa wpa3
```

> **Note**
>
> If both WPA2 and WPA3 are supported (SAE and PSK together), it is optional to configure PMF. However, you cannot disable PMF. For WPA3, PMF is mandatory.

10. Enable AKM SAE support.

Example

```
Device(config-wlan)# security wpa akm sae
```

11. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

12. Return to the privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

The WLAN is configured using WPA3 SAE authentication, providing enhanced wireless security.

Configure a WPA3 SAE iPSK WLAN profile (CLI)

Configure a WLAN profile with WPA3 SAE individual pre-shared key (iPSK) security on a Cisco device by using CLI.

Use this procedure to set up a WLAN profile with enhanced authentication and security standards (WPA3 SAE iPSK). Only authorized devices can connect using individual preshared keys

Before you begin

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the WLAN.

Example

```
Device(config)# wlanwlan-name wland-id SSID-name
```

- **wlan-name** is the name of the configured WLAN.
- **wlan-id** is the WLAN identifier. The range is one to 512.
- **SSID-name** is the SSID name which can have up to 32 alphanumeric characters.

If you have already created and configured the WLAN, use the **wlan** *wlan-name* command.

3. Set MAC filtering support in WLAN.

Example

```
Device(config-wlan)# mac-filtering mac-filter-name
```

4. Disable adaptive 802.11r.

Example

```
Device(config-wlan)# no security ft adaptive
```

5. Disable WPA2 security.

   Example

   ```
   Device(config-wlan)# no security wpa wpa2
   ```

6. Configures the preshared key in WLAN using the **security wpa psk set-key** [*ascii/hex*] **0** [*key*] command.

   Example

   ```
   Device(config-wlan)# security wpa psk set-key ascii 0 key
   ```

   Note

   WPA preshared keys must contain 8 to 63 ASCII text characters or 64 hexadecimal characters.

7. Disable security AKM for 802.1X.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

8. Enable AKM SAE support.

   Example

   ```
   Device(config-wlan)# security wpa akm sae
   ```

9. Enable AKM SAE PWE support (hash-to-element).

   Example

   ```
   Device(config-wlan)# security wpa akm sae pwe h2e
   ```

   Note

   This step applies to the Hunting and Pecking (HnP) password element method as well.

10. Enable WPA3 support.

Example

```
Device(config-wlan)# security wpa wpa3
```

11. Require clients to negotiate Protected Management Frames (PMF) protection in the WLAN.

Example

```
Device(config-wlan)# security pmf mandatory
```

12. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

Clients can now connect to this WLAN using the specified individual pre-shared keys and WPA3 security protocols after the WPA3 SAE iPSK WLAN profile is configured and enabled.

Configure a policy profile with AAA override and VLAN assignment (CLI)

Create and activate a wireless policy profile using the CLI.

Use this procedure to define a policy profile that controls AAA override, VLAN assignment, and other interface features on a Cisco device.

Before you begin

Verify that necessary VLANs exist on the device.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure policy profile.

Example

```
Device(config)# wireless profile policy policy-profile-name
```

3. Configure AAA override to apply to the policies coming from the AAA or Cisco Identity Services Engine (ISE) server.

Example

```
Device(config-wireless-policy)# aaa-override
```

4. Configure VLAN.

Example

```
Device(config-wireless-policy)# vlan 166
```

5. Enable policy profile.

Example

```
Device(config-wireless-policy)# no shutdown
```

The policy profile is configured and enabled for wireless operations on the device.

Configure a passphrase in a client authorization policy in the RADIUS server(GUI)

Set up a passphrase in a client authorization policy on the RADIUS server to support secure client authentication.

Configure a passphrase in Cisco Identity Services Engine (ISE). To do this, create a client authorization profile and include the required advanced attributes.

Before you begin

Procedure

---

1. Log in to the Cisco Identity Services Engine (ISE).
2. Click **Policy** and then click **Policy Elements**.
3. Click **Results**.
4. Expand **Authorization** and click **Authorization Profiles**.
5. Click **Add** to create a new authorization profile for the URL filter.
6. In the **Name** field, enter a name for the profile, for example, *po-sae-ipsk*.
7. From the **Access Type** drop-down list, choose **ACCESS_ACCEPT**.
8. From the **Termination-Action** drop-down list, choose **RADIUS-Request**.
9. In the **Advanced Attributes Setting** section, from the drop-down list, choose **Cisco:cisco-av-pair**.
10. Enter each value separately. After adding a value, click (+) icon after each of them:
    - `cisco-av-pair = psk-mode=ascii`
    - `cisco-av-pair = psk=123123123`
11. Verify the contents in the **Attributes Details** section and click **Save**.

You created a new client authorization profile in Cisco ISE with a custom passphrase. This enables secure client authentication through RADIUS.

Configure WPA3 SAE H2E (GUI)

Use WPA3 SAE H2E to secure your WLAN through a streamlined GUI-based configuration.

Use this procedure to strengthen Wi-Fi security by configuring WPA3 SAE H2E, which enhances protection against offline dictionary attacks.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**, the **SSID**, and the **WLAN ID**.
4. Choose **Security** > **Layer2** tab. From the **Layer 2 Security Mode** drop-down list, choose **WPA2+WPA3** or **WPA3**.
5. Uncheck the **WPAPolicy**, **802.1x**, **Over the DS**, **FT + 802.1x**, and **FT + PSK** check boxes. Check the **WPA3 Policy**, **AES** and **PSK** check boxes. Enter the **Pre-Shared Key**, and from the **PSK Format** drop-down list, choose the PSK Format, and from the **PSK Type** drop-down list, choose the PSK Type.
6. Check the **SAE** check box.

   Note

   SAE is enabled only if the Fast Transition is disabled.

7. From the **SAE Password Element** drop-down list, choose **Hash to Element Only** to configure the WPA3 SAE H2E.
8. Click **Apply to Device**.

Your WLAN is now secured with WPA3 SAE H2E, based on your configuration.

Configure WPA3 SAE H2E (CLI)

Enable WPA3 SAE Hash-to-Element (H2E) for secure WLAN connectivity.

Use this task to configure WPA3 SAE H2E for a WLAN on a device running Cisco IOS-XE.

Before you begin

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id SSID-name
   ```

   Example

   ```
   Device(config)# wlan WPA3 1 WPA3
   ```

3. Disable security AKM for 802/1X.

Example

```
Device(config-wlan)# no security wpa akm dot1x
```

4. Disable fast transition over the data source on the WLAN.

Example

```
Device(config-wlan)# no security ft over-the-ds
```

5. Disable 802.11r fast transition on the WLAN.

Example

```
Device(config-wlan)# no security ft
```

6. Disable WPA2 security. PMF is disabled now.

Example

```
Device(config-wlan)# no security wpa wpa2
```

7. Configure WPA2 cipher.

Example

```
Device(config-wlan)# security wpa wpa2 ciphers aes
```

> **Note**
>
> You can check whether cipher is configured using **no security wpa wpa2 ciphers aes** command. If cipher is not reset, configure the cipher.

8. Specify a preshared key

Example

```
Device(config-wlan)# security wpa psk set-key ascii value preshared-key
```

Example

```
Device(config-wlan)# security wpa psk set-key ascii 0 Cisco123
```

9. Enable WPA3 support.

Example

```
Device(config-wlan)# security wpa wpa3
```

10. Enable AKM SAE support.

Example

```
Device(config-wlan)# security wpa akm sae
```

11. Enable AKM SAE PWE support using the **security wpa akm sae pwe {h2e | hnp | both-h2e-hnp}** command.

Example

```
Device(config-wlan)# security wpa akm sae pwe
```

PWE supports these options:

- h2e: Hash-to-Element only; disables HnP.
- hnp: Hunting and Pecking only; disables H2E.
- Both-h2e-hnp: Both Hash-to-Element and Hunting and Pecking support (Is the default option).

12. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

13. Return to the privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

WPA3 SAE H2E is enabled for the specified WLAN. Wireless clients can now connect securely using WPA3 SAE H2E.

Configure WPA3 WLAN for Transition Disable

Configure WPA3 WLAN for Transition Disable using commands.

Enable the transition-disable feature for a WPA3 Wi-Fi network. This improves wireless security and prevents fallback to less secure protocols.

Use this task to ensure clients connect only with WPA3 security. The configuration prevents fallback to WPA2 in transition mode.

Before you begin

Make sure the **security wpa wpa3** command is enabled on your device, as transition disable is available only when WPA3 is enabled.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enter the WLAN configuration sub-mode.

Example

```
Device(config)# wlan wlan-name wlan-id SSID-name
```

3. Enable Transition Disable support.

Example

```
Device(config-wlan)# transition-disable
```

4. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

After completing these steps, clients can connect to your WPA3 WLAN only using WPA3 security.

Configure anti-clogging and SAE retransmission (GUI)

Set up Anti-Clogging and SAE retransmission parameters for a WLAN profile using the graphical user interface.

Perform this task to enable stronger wireless security features. This task also optimizes retransmission settings for your SSID.

Before you begin

Gather necessary values for Profile Name, SSID, WLAN ID, anti-clogging threshold, maximum retries, and retransmit timeout.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**, the **SSID** and the **WLAN ID**.
4. Enable or disable **Status** and **Broadcast SSID** toggle buttons.
5. From the **Radio Policy** drop-down list, choose a policy.
6. Choose **Security** > **Layer2** tab. Check the **SAE** check box.
7. Enter the **Anti Clogging Threshold**, **Max Retries** and **Retransmit Timeout**.
8. Click **Apply to Device**.

Anti-clogging and SAE retransmission settings are applied to the selected WLAN profile. Devices connecting to this SSID will now use the updated security and retransmission settings.

Configure anti-clogging and SAE retransmission (CLI)

Configure anti-clogging thresholds and SAE retransmission settings on a WLAN using CLI commands.

Anti-clogging and Secure Authentication of Equals (SAE) retransmission settings help prevent authentication floods and improve handshake reliability for your WLAN. Perform these steps after you configure the basic SAE WLAN settings.

Before you begin

Complete the SAE WLAN configuration

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id SSID-name
   ```

3. Disable the WLAN.

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

4. Enable simultaneous authentication of equals as a security protocol.

   Example

   ```
   Device(config-wlan)# security wpa akm sae
   ```

5. Configure threshold on the number of open sessions to trigger the anti-clogging procedure for new sessions.

   Example

   ```
   Device(config-wlan)# security wpa akm sae anti-clogging-threshold 2000
   ```

6. Configure the maximum number of retransmissions.

   Example

   ```
   Device(config-wlan)# security wpa akm sae max-retries retry-limit
   ```

7. Configure SAE message retransmission timeout value.

Example

```
Device(config-wlan)# security wpa akm sae retransmit-timeout
retransmit-timeout-limit
```

8. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

9. Return to the privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

Anti-clogging and SAE retransmission settings are applied to the WLAN.

Verify WPA3 SAE and OWE

Verify WPA3 SAE and OWE.

To view the system level statistics for the client that has undergone successful SAE authentication, SAE authentication failures, SAE ongoing sessions, SAE commit and confirm message exchanges, use this show command:

```
Device# show wireless stats client detail

        Total Number of Clients : 0

        client global statistics:

------------------------------------------------------------------------------

        Total association requests received             : 0
        Total association attempts                      : 0
        Total FT/LocalAuth requests                     : 0
        Total association failures                      : 0
        Total association response accepts              : 0
        Total association response rejects              : 0
        Total association response errors               : 0
        Total association failures due to blacklist     : 0
        Total association drops due to multicast mac    : 0
        Total association drops due to throttling       : 0
        Total association drops due to unknown bssid    : 0
        Total association drops due to parse failure    : 0
        Total association drops due to other reasons    : 0
        Total association requests wired clients        : 0
        Total association drops wired clients           : 0
        Total association success wired clients         : 0
        Total peer association requests wired clients   : 0
        Total peer association drops wired clients      : 0
        Total peer association success wired clients    : 0
        Total 11r ft authentication requests received   : 0
        Total 11r ft authentication response success    : 0
        Total 11r ft authentication response failure    : 0
        Total 11r ft action requests received           : 0
```

```
        Total 11r ft action response success             : 0
        Total 11r ft action response failure             : 0
        Total AID allocation failures                    : 0
        Total AID free failures                          : 0
        Total roam attempts                              : 0
        Total CCKM roam attempts                       : 0
        Total 11r roam attempts                        : 0
        Total 11i fast roam attempts                   : 0
        Total 11i slow roam attempts                   : 0
        Total other roam type attempts                 : 0
        Total roam failures in dot11                     : 0

        Total WPA3 SAE attempts                          : 0
        Total WPA3 SAE successful authentications        : 0
        Total WPA3 SAE authentication failures           : 0
        Total incomplete protocol failures             : 0
        Total WPA3 SAE commit messages received          : 0
        Total WPA3 SAE commit messages rejected          : 0
        Total unsupported group rejections             : 0
        Total WPA3 SAE commit messages sent              : 0
        Total WPA3 SAE confirm messages received         : 0
        Total WPA3 SAE confirm messages rejected         : 0
        Total WPA3 SAE confirm messgae field mismatch  : 0
        Total WPA3 SAE confirm message invalid length  : 0
        Total WPA3 SAE confirm messages sent             : 0
        Total WPA3 SAE Open Sessions                     : 0
        Total SAE Message drops due to throttling        : 0

        Total Flexconnect local-auth roam attempts       : 0
        Total AP 11i fast roam attempts                : 0
        Total 11i slow roam attempts                   : 0

        Total client state starts                        : 0
        Total client state associated                    : 0
        Total client state l2auth success                : 0
        Total client state l2auth failures               : 0
        Total blacklisted clients on dot1xauth failure   : 0
        Total client state mab attempts                  : 0
        Total client state mab failed                    : 0
        Total client state ip learn attempts             : 0
        Total client state ip learn failed               : 0
        Total client state l3 auth attempts              : 0
        Total client state l3 auth failed                : 0
        Total client state session push attempts         : 0
        Total client state session push failed           : 0
        Total client state run                           : 0
        Total client deleted                             : 0
```

To view the WLAN summary details, use this command.

```
Device# show wlan summary

        Number of WLANs: 3

        ID   Profile Name                    SSID
  Status Security


--------------------------------------------------------------------------------------------------
```

```
      1    wlan-demo                          ssid-demo
 DOWN   [WPA3][SAE][AES]

      3    CR1_SSID_mab-ext-radius            CR1_SSID_mab-ext-radius
 DOWN   [WPA2][802.1x][AES]

      109  guest-wlan1                        docssid
 DOWN   [WPA2][802.1x][AES],[Web Auth]
```

To view the WLAN properties (WPA2 and WPA3 mode) based on the WLAN ID, use this command.

```
Device# show wlan id 1

        WLAN Profile Name     : wlan-demo
        ================================================
        Identifier                                    : 1

        !
        !
        !
        Security
        802.11 Authentication                        : Open System
        Static WEP Keys                              : Disabled
        Wi-Fi Protected Access (WPA/WPA2/WPA3)     : Enabled
        WPA (SSN IE)                              : Disabled
        WPA2 (RSN IE)                             : Disabled
        WPA3 (WPA3 IE)                            : Enabled
        AES Cipher                              : Enabled
        CCMP256 Cipher                          : Disabled
        GCMP128 Cipher                          : Disabled
        GCMP256 Cipher                          : Disabled
        Auth Key Management
        802.1x                                  : Disabled
        PSK                                     : Disabled
        CCKM                                    : Disabled
        FT dot1x                                : Disabled
        FT PSK                                  : Disabled
        Dot1x-SHA256                            : Disabled
        PSK-SHA256                              : Disabled
        SAE                                     : Enabled
        OWE                                     : Disabled
        SUITEB-1X
```

Verify WPA3 SAE H2E support in WLAN

Verify WPA3 SAE H2E Support in WLAN.

To view the WLAN properties (PWE method) based on the WLAN ID, use this command:

```
Device# show wlan id 1
WLAN Profile Name     : wpa3
================================================
Identifier                                    : 1
Description                                   :
Network Name (SSID)                           : wpa3
Status                                        : Enabled
Broadcast SSID                                : Enabled
```

```
Advertise-Apname                                : Disabled
Universal AP Admin                              : Disabled
Max Associated Clients per WLAN                 : 0
Max Associated Clients per AP per WLAN          : 0
Max Associated Clients per AP Radio per WLAN    : 200
OKC                                             : Enabled
Number of Active Clients                        : 0
CHD per WLAN                                    : Enabled
WMM                                             : Allowed
WiFi Direct Policy                              : Disabled
Channel Scan Defer Priority:
  Priority (default)                            : 5
  Priority (default)                            : 6
Scan Defer Time (msecs)                         : 100
Media Stream Multicast-direct                   : Disabled
CCX - AironetIe Support                         : Disabled
Peer-to-Peer Blocking Action                    : Disabled
DTIM period for 802.11a radio                   : 1
DTIM period for 802.11b radio                   : 1
Local EAP Authentication                        : Disabled
Mac Filter Authorization list name              : Disabled
Mac Filter Override Authorization list name     : Disabled
Accounting list name                            :
802.1x authentication list name                 : Disabled
802.1x authorization list name                  : Disabled
Security
    802.11 Authentication                       : Open System
    Static WEP Keys                             : Disabled
    Wi-Fi Protected Access (WPA/WPA2/WPA3)      : Enabled
        WPA (SSN IE)                            : Disabled
        WPA2 (RSN IE)                           : Disabled
        WPA3 (WPA3 IE)                          : Enabled
            AES Cipher                          : Enabled
            CCMP256 Cipher                      : Disabled
            GCMP128 Cipher                      : Disabled
            GCMP256 Cipher                      : Disabled
        Auth Key Management
            802.1x                              : Disabled
            PSK                                 : Disabled
            CCKM                                : Disabled
            FT dot1x                            : Disabled
            FT PSK                              : Disabled
            Dot1x-SHA256                        : Disabled
            PSK-SHA256                          : Disabled
            SAE                                 : Enabled
            OWE                                 : Disabled
            SUITEB-1X                           : Disabled
            SUITEB192-1X                        : Disabled
    SAE PWE Method                              : Hash to Element(H2E)
    Transition Disable                          : Disabled
    CCKM TSF Tolerance (msecs)                  : 1000
    OWE Transition Mode                         : Disabled
    OSEN                                        : Disabled
    FT Support                                  : Disabled
        FT Reassociation Timeout (secs)         : 20
        FT Over-The-DS mode                     : Disabled
    PMF Support                                 : Required
        PMF Association Comeback Timeout (secs): 1
        PMF SA Query Time (msecs)               : 200
    Web Based Authentication                    : Disabled
    Conditional Web Redirect                    : Disabled
    Splash-Page Web Redirect                    : Disabled
```

```
    Webauth On-mac-filter Failure              : Disabled
    Webauth Authentication List Name           : Disabled
    Webauth Authorization List Name            : Disabled
    Webauth Parameter Map                      : Disabled
Band Select                                    : Disabled
Load Balancing                                 : Disabled
Multicast Buffer                               : Disabled
Multicast Buffers (frames)                     : 0
IP Source Guard                                : Disabled
Assisted-Roaming
    Neighbor List                              : Enabled
    Prediction List                            : Disabled
    Dual Band Support                          : Disabled
IEEE 802.11v parameters
    Directed Multicast Service                 : Enabled
    BSS Max Idle                               : Enabled
        Protected Mode                         : Disabled
    Traffic Filtering Service                  : Disabled
    BSS Transition                             : Enabled
        Disassociation Imminent                : Disabled
            Optimised Roaming Timer (TBTTS)    : 40
            Timer (TBTTS)                      : 200
        Dual Neighbor List                     : Disabled
    WNM Sleep Mode                             : Disabled
802.11ac MU-MIMO                               : Enabled
802.11ax parameters
    802.11ax Operation Status                  : Enabled
    OFDMA Downlink                             : Enabled
    OFDMA Uplink                               : Enabled
    MU-MIMO Downlink                           : Enabled
    MU-MIMO Uplink                             : Enabled
    BSS Target Wake Up Time                    : Enabled
    BSS Target Wake Up Time Broadcast Support  : Enabled
802.11 protocols in 2.4ghz band
    Protocol                                   : dot11bg
Advanced Scheduling Requests Handling          : Enabled
mDNS Gateway Status                            : Bridge
WIFI Alliance Agile Multiband                  : Disabled
Device Analytics
    Advertise Support                          : Enabled
    Advertise Support for PC analytics         : Enabled
    Share Data with Client                     : Disabled
Client Scan Report (11k Beacon Radio Measurement)
    Request on Association                     : Disabled
    Request on Roam                            : Disabled
WiFi to Cellular Steering                      : Disabled
Advanced Scheduling Requests Handling          : Enabled
Locally Administered Address Configuration
    Deny LAA clients                           : Disabled
```

To verify the client association who have used the PWE method as H2E or HnP, use this command:

```
Device# show wireless client mac-address e884.a52c.47a5 detail
Client MAC Address : e884.a52c.47a5
Client MAC Type : Universally Administered Address
Client DUID: NA
Client IPv4 Address : 11.11.0.65
Client IPv6 Addresses : fe80::c80f:bb8c:86f6:f71f
Client Username: N/A
AP MAC Address : d4ad.bda2.e9e0
AP Name: APA453.0E7B.E73C
AP slot : 1
```

```
Client State : Associated
Policy Profile : default-policy-profile
Flex Profile : N/A
Wireless LAN Id: 1
WLAN Profile Name: wpa3
Wireless LAN Network Name (SSID): wpa3
BSSID : d4ad.bda2.e9ef
Connected For : 72 seconds
Protocol : 802.11ax - 5 GHz
Channel : 36
Client IIF-ID : 0xa0000001
Association Id : 2
Authentication Algorithm : Simultaneous Authentication of Equals (SAE)
Idle state timeout : N/A
Session Timeout : 1800 sec (Remaining time: 1728 sec)
Session Warning Time : Timer not running
Input Policy Name  : None
Input Policy State : None
Input Policy Source : None
Output Policy Name  : None
Output Policy State : None
Output Policy Source : None
WMM Support : Enabled
U-APSD Support : Disabled
Fastlane Support : Disabled
Client Active State : Active
Power Save : OFF
Current Rate : m6 ss2
Supported Rates : 6.0,9.0,12.0,18.0,24.0,36.0,48.0,54.0
AAA QoS Rate Limit Parameters:
  QoS Average Data Rate Upstream             : 0 (kbps)
  QoS Realtime Average Data Rate Upstream    : 0 (kbps)
  QoS Burst Data Rate Upstream               : 0 (kbps)
  QoS Realtime Burst Data Rate Upstream      : 0 (kbps)
  QoS Average Data Rate Downstream           : 0 (kbps)
  QoS Realtime Average Data Rate Downstream  : 0 (kbps)
  QoS Burst Data Rate Downstream             : 0 (kbps)
  QoS Realtime Burst Data Rate Downstream    : 0 (kbps)
Mobility:
  Move Count                  : 0
  Mobility Role               : Local
  Mobility Roam Type          : None
  Mobility Complete Timestamp : 08/24/2021 04:39:47 Pacific
Client Join Time:
  Join Time Of Client : 08/24/2021 04:39:47 Pacific
Client State Servers : None
Client ACLs : None
Policy Manager State: Run
Last Policy Manager State : IP Learn Complete
Client Entry Create Time : 72 seconds
Policy Type : WPA3
Encryption Cipher : CCMP (AES)
Authentication Key Management : SAE
AAA override passphrase : No
SAE PWE Method : Hash to Element(H2E)
Transition Disable Bitmap : None
User Defined (Private) Network : Disabled
User Defined (Private) Network Drop Unicast : Disabled
Encrypted Traffic Analytics : No
Protected Management Frame - 802.11w : Yes
EAP Type : Not Applicable
VLAN Override after Webauth : No
```

```
VLAN : VLAN0011
Multicast VLAN : 0
WiFi Direct Capabilities:
  WiFi Direct Capable           : No
Central NAT : DISABLED
Session Manager:
  Point of Attachment : capwap_90000006
  IIF ID              : 0x90000006
  Authorized          : TRUE
  Session timeout     : 1800
  Common Session ID: 000000000000000C76750C17
  Acct Session ID  : 0x00000000
  Auth Method Status List
   Method : SAE
  Local Policies:
   Service Template : wlan_svc_default-policy-profile_local (priority 254)
    VLAN              : VLAN0011
    Absolute-Timer    : 1800
  Server Policies:
  Resultant Policies:
                VLAN Name         : VLAN0011
    VLAN              : 11
    Absolute-Timer    : 1800
DNS Snooped IPv4 Addresses : None
DNS Snooped IPv6 Addresses : None
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 0
Fast BSS Transition Details :
  Reassociation Timeout : 0
11v BSS Transition : Implemented
11v DMS Capable : No
QoS Map Capable : Yes
FlexConnect Data Switching : N/A
FlexConnect Dhcp Status : N/A
FlexConnect Authentication : N/A
Client Statistics:
  Number of Bytes Received from Client : 21757
  Number of Bytes Sent to Client : 4963
  Number of Packets Received from Client : 196
  Number of Packets Sent to Client : 37
  Number of Policy Errors : 0
  Radio Signal Strength Indicator : -72 dBm
  Signal to Noise Ratio : 20 dB
Fabric status : Disabled
Radio Measurement Enabled Capabilities
  Capabilities: Neighbor Report, Passive Beacon Measurement, Active Beacon
Measurement, Table Beacon Measurement
Client Scan Report Time : Timer not running
Client Scan Reports
Assisted Roaming Neighbor List
```

To view the number of SAE authentications using the H2E and HnP, use this command:

```
Device# show wireless stats client detail
Total Number of Clients : 0
```

```
Protocol Statistics

--------------------------------------------------------------------------------
Protcol            Client Count
 802.11b              : 0
 802.11g              : 0
 802.11a              : 0
 802.11n-2.4GHz       : 0
 802.11n-5 GHz        : 0
 802.11ac             : 0
 802.11ax-5 GHz       : 0
 802.11ax-2.4 GHz     : 0
  802.11ax-6 GHz      : 0



Current client state statistics:
--------------------------------------------------------------------------------

  Authenticating         : 0
  Mobility               : 0
  IP Learn               : 0
  Webauth Pending        : 0
  Run                    : 0
  Delete-in-Progress     : 0


Client Summary
-----------------------------
Current Clients : 0
Excluded Clients: 0
Disabled Clients: 0
Foreign Clients : 0
Anchor Clients  : 0
Local Clients   : 0
Idle Clients    : 0
Locally Administered MAC Clients: 0

client global statistics:
--------------------------------------------------------------------------------
Total association requests received              : 0
Total association attempts                       : 0
Total FT/LocalAuth requests                      : 0
Total association failures                       : 0
Total association response accepts               : 0
Total association response rejects               : 0
Total association response errors                : 0
Total association failures due to exclusion list      : 0
Total association drops due to multicast mac     : 0
Total association drops due to random mac        : 0
Total association drops due to throttling        : 0
Total association drops due to unknown bssid     : 0
Total association drops due to parse failure     : 0
Total association drops due to other reasons     : 0
Total association requests wired clients         : 0
Total association drops wired clients            : 0
Total association success wired clients          : 0
Total peer association requests wired clients    : 0
Total peer association drops wired clients       : 0
Total peer association success wired clients     : 0
Total association success wifi direct clients    : 0
Total association rejects wifi direct clients    : 0
```

```
Total association response errors                : 0
Total 11r ft authentication requests received    : 0
Total 11r ft authentication response success     : 0
Total 11r ft authentication response failure     : 0
Total 11r ft action requests received            : 0
Total 11r ft action response success             : 0
Total 11r ft action response failure             : 0
Total 11r PMKR0-Name mismatch                    : 0
Total 11r PMKR1-Name mismatch                    : 0
Total 11r MDID mismatch                          : 0
Total AID allocation failures                    : 0
Total AID free failures                          : 0
Total Roam Across Policy Profiles                : 0
Total roam attempts                              : 0
  Total CCKM roam attempts                       : 0
  Total 11r roam attempts                        : 0
  Total 11r slow roam attempts                   : 0
  Total 11i fast roam attempts                   : 0
  Total 11i slow roam attempts                   : 0
  Total other roam type attempts                 : 0
Total roam failures in dot11                     : 0

Total WPA3 SAE attempts                          : 0
Total WPA3 SAE successful authentications        : 0
Total WPA3 SAE authentication failures           : 0
  Total incomplete protocol failures             : 0
Total WPA3 SAE commit messages received          : 0
Total WPA3 SAE commit messages rejected                                    : 0
  Total unsupported group rejections                                       : 0
  Total PWE method mismatch for SAE Hash to Element commit received        : 0
  Total PWE method mismatch for SAE Hunting And Pecking commit received : 0
Total WPA3 SAE commit messages sent              : 0
Total WPA3 SAE confirm messages received         : 0
Total WPA3 SAE confirm messages rejected         : 0
  Total WPA3 SAE message confirm field mismatch  : 0
  Total WPA3 SAE confirm message invalid length  : 0
Total WPA3 SAE confirm messages sent             : 0
Total WPA3 SAE Open Sessions                     : 0
Total SAE Message drops due to throttling        : 0
Total WPA3 SAE Hash to Element commit received   : 0
Total WPA3 SAE Hunting and Pecking commit received : 0

Total Flexconnect local-auth roam attempts       : 0
  Total AP 11i fast roam attempts                : 0
  Total AP 11i slow roam attempts                : 0
  Total 11r flex roam attempts                   : 0
```

Verify WPA3 transition disable in WLAN

Verify WPA3 Transition Disable in WLAN.

To view the WLAN properties (transition disable) based on the WLAN ID, use this command:

```
Device# show wlan id 7

WLAN Profile Name     : wl-sae
================================================
Identifier                                       : 7
Description                                      :
Network Name (SSID)                              : wl-sae
Status                                           : Enabled
```

```
Broadcast SSID                                  : Enabled
Advertise-Apname                                : Disabled
Universal AP Admin                              : Disabled
Max Associated Clients per WLAN                 : 0
Max Associated Clients per AP per WLAN          : 0
Max Associated Clients per AP Radio per WLAN    : 200
OKC                                             : Enabled
Number of Active Clients                        : 0
CHD per WLAN                                    : Enabled
WMM                                             : Allowed
WiFi Direct Policy                              : Disabled
Channel Scan Defer Priority:
  Priority (default)                            : 5
  Priority (default)                            : 6
Scan Defer Time (msecs)                         : 100
Media Stream Multicast-direct                   : Disabled
CCX - AironetIe Support                         : Disabled
Peer-to-Peer Blocking Action                    : Disabled
Configured Radio Bands                          : All
Operational State of Radio Bands
    2.4ghz                                      : UP
    5ghz                                        : UP
DTIM period for 802.11a radio                   :
DTIM period for 802.11b radio                   :
Local EAP Authentication                        : Disabled
Mac Filter Authorization list name              : Disabled
Mac Filter Override Authorization list name     : Disabled
Accounting list name                            :
802.1x authentication list name                 : Disabled
802.1x authorization list name                  : Disabled
Security
    802.11 Authentication                       : Open System
    Static WEP Keys                             : Disabled
    Wi-Fi Protected Access (WPA/WPA2/WPA3)      : Enabled
        WPA (SSN IE)                            : Disabled
        WPA2 (RSN IE)                           : Enabled
            MPSK                                : Disabled
            EasyPSK                             : Disabled
            AES Cipher                          : Enabled
            CCMP256 Cipher                      : Disabled
            GCMP128 Cipher                      : Disabled
            GCMP256 Cipher                      : Disabled
            Randomized GTK                      : Disabled
        WPA3 (WPA3 IE)                          : Enabled
            AES Cipher                          : Enabled
            CCMP256 Cipher                      : Disabled
            GCMP128 Cipher                      : Disabled
            GCMP256 Cipher                      : Disabled
        Auth Key Management
            802.1x                              : Disabled
            PSK                                 : Enabled
            CCKM                                : Disabled
            FT dot1x                            : Disabled
            FT PSK                              : Disabled
            Dot1x-SHA256                        : Disabled
            PSK-SHA256                          : Disabled
            SAE                                 : Enabled
            OWE                                 : Disabled
            SUITEB-1X                           : Disabled
            SUITEB192-1X                        : Disabled
    Transition Disable                          : Enabled
```

```
    CCKM TSF Tolerance (msecs)                  : 1000
```

To verify the client association who have used the transition disable, use this command:

```
Device# show wireless client mac-address 2c33.7a5b.8fc5 detail
Client MAC Address : 2c33.7a5b.8fc5
Client MAC Type : Universally Administered Address
Client DUID: NA
Client IPv4 Address : 10.166.1.101
Client Username: N/A
AP MAC Address : 7c21.0d48.ed00
AP Name: APF4BD.9EBD.A66C
AP slot : 0
Client State : Associated
Policy Profile : po-sae
Flex Profile : N/A
Wireless LAN Id: 7
WLAN Profile Name: wl-sae
Wireless LAN Network Name (SSID): wl-sae
BSSID : 7c21.0d48.ed02
Connected For : 15 seconds
Protocol : 802.11n - 2.4 GHz
Channel : 11
Client IIF-ID : 0xa0000002
Association Id : 1
Authentication Algorithm : Simultaneous Authentication of Equals (SAE)
Idle state timeout : N/A
Session Timeout : 1800 sec (Remaining time: 1787 sec)
Session Warning Time : Timer not running
Input Policy Name : None
Input Policy State : None
Input Policy Source : None
Output Policy Name : None
Output Policy State : None
Output Policy Source : None
WMM Support : Enabled
U-APSD Support : Disabled
Fastlane Support : Disabled
Client Active State : In-Active
Power Save : OFF
Supported Rates : 1.0,2.0,5.5,6.0,9.0,11.0,12.0,18.0,24.0,36.0,48.0,54.0
AAA QoS Rate Limit Parameters:
QoS Average Data Rate Upstream : 0 (kbps)
QoS Realtime Average Data Rate Upstream : 0 (kbps)
QoS Burst Data Rate Upstream : 0 (kbps)
QoS Realtime Burst Data Rate Upstream : 0 (kbps)
QoS Average Data Rate Downstream : 0 (kbps)
QoS Realtime Average Data Rate Downstream : 0 (kbps)
QoS Burst Data Rate Downstream : 0 (kbps)
QoS Realtime Burst Data Rate Downstream : 0 (kbps)
Mobility:
Move Count : 0
Mobility Role : Local
Mobility Roam Type : None
Mobility Complete Timestamp : 05/16/2021 11:18:14 UTC
Client Join Time:
Join Time Of Client : 05/16/2021 11:18:14 UTC
Client State Servers : None
Client ACLs : None
Policy Manager State: Run
Last Policy Manager State : IP Learn Complete
```

```
Client Entry Create Time : 15 seconds
Policy Type : WPA3
Encryption Cipher : CCMP (AES)
Authentication Key Management : SAE
AAA override passphrase : No
Transition Disable Bitmap : 0x01
User Defined (Private) Network : Disabled
User Defined (Private) Network Drop Unicast : Disabled
Encrypted Traffic Analytics : No
Protected Management Frame - 802.11w : Yes
```

125 WPA3 Security Enhancements for Access Points

Topics:

- WPA3 security enhancements for APs

About WPA3 security enhancements for APs.

WPA3 security enhancements for APs

Introduces WPA3 security enhancements for access points and outlines key guidelines and limitations to consider when deploying WPA3 features in wireless environments.

WPA3 security enhancements for access points are wireless security features that

- introduce advanced encryption methods such as GCMP-256 and SuiteB-192-1X AKM
- strengthen authentication and data protection with features like SAE-EXT-KEY support and AP beacon protection, and
- allow multiple cipher suites to be configured per WLAN, providing greater flexibility and security for wireless networks.

Cipher suites are sets of encryption and integrity algorithms designed to protect radio communication on your wireless LANs. You must use a cipher suite when using Wi-Fi Protected Access (WPA), WPA2, WPA3, or Cisco Centralized Key Management (CCKM). Wired Equivalent Privacy, or WEP, is a form of wireless authentication used for associating to 802.11 wireless networks.

Feature Name	Release	Description
Wi-Fi 7 WPA3 Security Constraints	Cisco IOS XE 17.15.2	In this release, the Wi-Fi 7 standard dictates these security constraints, which are applicable for Wi-Fi 7-compliant APs. The security standards listed are beaconed as Wi-Fi 7 clients, which is a deviation from the actual security constraint. • Open authentication as Wi-Fi 7 is not permitted to associate. • WPA1 as Wi-Fi 7 is not permitted to associate. • WPA2 as Wi-Fi 7 is not permitted to associate. • WPA3 is permitted with certain restrictions. Restrictions to permit WPA3. • SAE (24/25) is permitted with GCMP-256. • SAE (8/9) is permitted. This is a deviation from the actual security constraint. • WPA2 PSK/802.1X with PMF is permitted. This is a deviation from the actual security constraint. • 802.1X-SHA256 with PMF is permitted. • Suite-B-192 with PMF is permitted.

Feature Name	Release	Description
Decoupled GCMP-256 and Suite-B-192-1X AKM Configuration	Cisco IOS XE 17.15.1	There is a strong dependency between the GCMP-256 cipher and the Suite-B-192-1X AKM. Until Cisco IOS XE 17.14.1, configuring the GCMP-256 cipher automatically enabled the Suite-B-192-1X AKM because it could not be enabled separately using commands. In the Cisco IOS XE 17.15.1 release, this dependency is removed through new configuration options, allowing the GCMP-256 cipher to be used with other supported AKMs. Suite-B-192-1X AKM is useful for enterprise networks, including federal government and health care deployments that require the highest level of security. Until Cisco IOS XE 17.14.1, Suite-B-192-1X AKM was tied to GCMP-256 and was implicitly enabled when GCMP-256 was configured at the WLAN level. From Cisco IOS XE 17.15.1 onward, a new AKM configuration allows Suite-B-192-1X AKM to be enabled separately, and GCMP-256 cipher configuration applies only to the cipher.

Feature Name	Release	Description
Wi-Fi Protected Access (WPA3) Security Enhancements for Access Points	Cisco IOS XE 17.15.1	From Cisco IOS XE 17.15.1 onwards, OWE association is supported on both CCMP-128 and GCMP-256 ciphers. If you configure both ciphers, a client will select its desired cipher suite while connecting in the association request. The security enhancements developed in Cisco IOS XE 17.15.1 for APs are: • GCMP-256 Cipher and Secure Agile Exchange (SAE) Autonomous Key Management (AKM) • SAE-EXT Key Support • AP Beacon Protection • Multiple Cipher Support Per WLAN • Opportunistic Wireless Encryption (OWE) Support with GCMP-256 Cipher The configuration commands introduced are: • **security wpa akm sae ext-key** • **security wpa akm ft sae ext-key** • **security wpa akm suiteb-192** • **security wpa akm suiteb** • **security wpa wpa2 ciphers** • **security wpa wpa3 beacon-protection**
Opportunistic Wireless Encryption (OWE) Support with GCMP-256 Cipher	Cisco IOS XE 17.14.1	Until Cisco IOS XE 17.14.1, OWE was supported with the CCMP-128 cipher.

Benefits of using GCMP-Based ciphers

- Provides secure communication and data transmission.
- Provides confidentiality and integrity protection.
- Provides parallel processing and fast encryption.

Wireless Encryption Methods for Data Protection

Encryption protects data by making it unreadable to unauthorized users.

These are the encryption protocols are used in wireless authentication.

- **Temporal Key Integrity Protocol (TKIP)**: TKIP is the encryption method used by WPA and supports legacy WLAN equipment. TKIP addresses the original flaws associated with the 802.11 WEP encryption method. It makes use of WEP but encrypts the Layer 2 payload using TKIP and carries out a message integrity check (MIC) in encrypted packets to ensure that messages have not been altered.
- **Advanced Encryption Standard (AES)**: AES uses Counter Cipher Mode with Block Chaining Message Authentication Code Protocol (CCMP), which allows destination hosts to recognize if the encrypted and non-encrypted bits have been altered.

  Attention

  AES is a preferred method because of its strong encryption.

  CCMP is the standard encryption protocol for use with Wi-Fi Protected Access 2 (WPA2) and is much more secure than the WEP protocol, and TKIP of WPA.
- **Galois or Counter Mode Protocol (GCMP):** GCMP is more secure and efficient than CCMP.

Note

Choose the appropriate note type from the attributes.

Supported platforms

- Cisco Catalyst 9800-CL Wireless Controller for Cloud
- Cisco Catalyst 9800-L Wireless Controller
- Cisco Catalyst 9800-40 Wireless Controller
- Cisco Catalyst 9800-80 Wireless Controller
- Cisco Catalyst 9300 Series Switches
- Cisco Embedded Wireless Controller on Catalyst Access Points

Supported APs

- Cisco Aironet 2800 Series Access Points
- Cisco Aironet 3800 Series Access Points
- Cisco Aironet 4800 Series Access Points
- Cisco Catalyst 9117 Series Access Points
- Cisco Catalyst 9124AX Series Access Points
- Cisco Catalyst 9130AX Series Access Points
- Cisco Catalyst 9136 Series Access Points
- Cisco Catalyst 9162 Series Access Points
- Cisco Catalyst 9164 Series Access Points

- Cisco Catalyst 9166 Series Access Points
- Cisco Aironet 1560 Series Outdoor Access Points

Restrictions

This section lists the guidelines and limitations for AP WPA3 security enhancements.

- WPA3 is not supported on Cisco Wave 1 APs.
- GCMP-256 is not supported on Cisco Catalyst 9105, 9110, 9115, 9120 APs and 802.11ac Wave2 QCA APs such as 1852.
- Beacon Protection is only supported on QCA-based APs, such as 9130 , 9136, 9162, 9164, and 9166.

Wi-Fi 7 WPA3 security constraints

In the Cisco IOS XE 17.15.2 release, the Wi-Fi 7 standard dictates these security constraints, which are applicable for Wi-Fi 7 compliant APs:

The security standards described here are advertised as Wi-Fi 7 clients. This is a deviation from the actual security constraint.

- Open authentication as Wi-Fi 7 is not permitted to associate.
- WPA1 as Wi-Fi 7 is not permitted to associate.
- WPA2 as Wi-Fi 7 is not permitted to associate.
- WPA3 is permitted with certain restrictions:
  - SAE 24 or 25 is permitted with GCMP-256.
  - SAE 8 or 9 is permitted. (This is a deviation from the actual security constraint.)
  - WPA2 PSK or 802.1x with PMF is permitted. (This is a deviation from the actual security constraint.)
  - 802.1x-SHA256 with PMF is permitted.
  - Suite-B-192 with PMF is permitted.

GCMP-256 Cipher and SuiteB-192-1X AKM

Explains support for GCMP-256 cipher and SuiteB-192-1X AKM, providing procedures for configuring SuiteB-192-1X AKM using both GUI and CLI interfaces.

A GCMP-256 cipher and SuiteB-192-1X AKM are Wi-Fi security mechanisms that

- provide strong encryption and authentication for wireless communications
- help meet high-security requirements for environments such as government and healthcare, and
- support flexible configuration in recent Cisco IOS XE releases.

Decoupled GCMP-256 and Suite-B-192-1X AKM configuration

There is a strong dependency between the GCMP-256 cipher with Suite-B-192-1X AKM. Therefore, until Cisco IOS XE 17.14.1, if you configure the GCMP-256 cipher, the Suite-B-192-1X AKM automatically gets enabled, as Suite-B-192-1X Authentication and Key Management (AKM) cannot be enabled separately using commands.

In the Cisco IOS XE 17.15.1 release, certain commands eliminate the dependency between Suite-B-192-1X AKM and the GCMP-256 cipher. CMP-256 cipher can be configured with other supported AKMs.

GCMP-256 cipher and Suite-B-192-1X AKM were previously strongly dependent features.

- Until Cisco IOS XE 17.14.1, enabling GCMP-256 cipher also enabled Suite-B-192-1X AKM, and Suite-B-192-1X AKM could not be enabled separately.
- From Cisco IOS XE 17.15.1 onwards, the dependency is removed and each can be configured independently.

Configure SuiteB-192-1X AKM (GUI)

Enable SuiteB-192-1X Authentication and Key Management (AKM) for a WLAN using the required cipher combinations.

Use this procedure to configure secure authentication and encryption for WLANs, ensuring compliance with SuiteB-192-1X standards.

Before you begin

Confirm the device supports SuiteB-192-1X AKM and required cipher options.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.

   The **Add WLAN** window is displayed.
3. In the **General** tab, enter the **Profile Name**, **SSID**, and the **WLAN ID**.
4. Choose **Security** > **Layer2**, select one of these options:
   - **WPA + WPA2**
   - **WPA2 + WPA3**
   - **WPA3**

   The **Auth Key Mgmt (AKM)** section displays the AKMs supported by the cipher you select in the **WPA2/WPA3 Encryption** section. Valid cipher and AKM combinations are displayed in the **Auth Key Mgmt (AKM)** section.

   For example, to enable SuiteB-192-1x AKM,
   - The valid security encryption and AKM combination for **WPA + WPA2** and **WPA2 + WPA3** is **CCMP256** and/or **GCMP256** cipher + **SuiteB-192-1X** AKM.

     Note

     CCMP256 cipher is not valid without the GCMP256 cipher for SuiteB-192-1X AKM.

   - The valid security encryption and AKM combination for **WPA3** is **GCMP256** cipher + **SUITEB-192-1X** or **OWE** or **SAE-EXT-KEY** or **FT + SAE-EXT-KEY** AKM.

     Note

     At least one AKM should be enabled. To enable SuiteB-192-1X, check the SUITEB 192-1X check box.

5. In the **WPA2 Encryption** section, check the **GCMP256** check box.

   Valid cipher and AKM combinations are displayed in the **Auth Key Mgmt (AKM)** section.

6. In the **Fast Transition** section, in the **Status** drop-down list, select **Disabled**.

> **Note**
>
> Disable **Fast Transition** when Suite-B cipher (GCMP256, CCMP256, GCMP128) is configured.

7. In the **Auth Key Mgmt (AKM)** section, check the **SUITEB192-1X** check box.
8. Click **Apply to Device**.

The WLAN is now configured with SuiteB-192-1X AKM and the required ciphers, enabling secure authentication and encryption.

Configure SuiteB-192-1X AKM (CLI)

Enable SuiteB-192-1X Authentication and Key Management (AKM) for a WLAN using the required cipher combinations.

Use this procedure to configure secure authentication and encryption for WLANs, ensuring compliance with SuiteB-192-1X standards.

Before you begin

Confirm the device supports SuiteB-192-1X AKM and required cipher options.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the WLAN profile and SSID. Enter the WLAN configuration mode.

Example

```
Device(config)# wlan wlan-profile-name wlan-id ssid-name
```

3. Disable adaptive 802.11r.

Example

```
Device(config-wlan)# no security ft adaptive
```

4. Disable security AKM for 802.1X.

Example

```
Device(config-wlan)# no security wpa akm dot1x
```

5. Configure the SuiteB-192-1X support.

Example

```
Device(config-wlan)# security wpa akm suiteb-192
```

6. Configure the GCMP256 support using the **security wpa wpa2 ciphers {aes | ccmp256 | gcmp128 | gcmp256}** command.

Example

```
Device(config-wlan)# security wpa wpa2 ciphers gcmp256
```

The WLAN is now configured with SuiteB-192-1X AKM and the required ciphers, enabling secure authentication and encryption.

SAE-EXT-KEY support

Describes SAE-EXT-KEY and FT-SAE-EXT-KEY AKM capabilities, offering step-by-step instructions for configuring them through GUI and CLI, and explains how to verify SAE-EXT-KEY AKM support.

The Cisco IOS XE 17.15.1 release introduces new SAE AKMs: SAE-EXT-KEY (24) and FT-SAE-EXT-KEY (25).

- Devices can connect using the new SAE AKMs (24 and 25) and negotiate encryption with GCMP-256, CCMP-128, or both ciphers.

> **Note**
>
> Ensure that the WPA3 policy is enabled for the new AKMs to be displayed.

Configure SAE-EXT-KEY AKMs (GUI)

Set up SAE-EXT-KEY Authentication and Key Management (AKM) on a WLAN to enhance wireless security with advanced password-based authentication protocols.

Use this task when configuring a Wi-Fi network profile requiring the SAE-EXT-KEY or Fast Transition (FT) + SAE-EXT-KEY AKMs for WPA2 or WPA3-enabled WLANs. These settings enable secure password management and provide robust wireless security.

Before you begin

- Ensure that the WLAN controller supports SAE-EXT-KEY and FT + SAE-EXT-KEY AKMs.
- Verify that WPA3 security policies are enabled.
- Gather values for anti-clogging threshold, maximum retries, retransmit timeout, and the required pre-shared key (PSK).

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.

   The **Add WLAN** window is displayed.

3. In the **General** tab, enter the **Profile Name**, **SSID**, and the **WLAN ID**.
4. Choose **Security** > **Layer2** and select one of these options:
   - **WPA2 + WPA3**
   - **WPA3**

   The **Auth Key Mgmt (AKM)** section will be populated with the possible AKMs that are supported by the cipher that is selected in the **WPA2/WPA3 Encryption** section. Valid AKMs are displayed in the **Auth Key Mgmt (AKM)** section.

   > **Note**
   >
   > Ensure that the WPA3 policy is enabled for the new AKMs to be displayed.

5. In the **WPA2/WPA3 Encryption** section, check the **GCMP256** check box, or the **AES(CCMP128)** check box, or a combination of both these check boxes.

   > **Note**
   >
   > The AES(CCMP128) cipher check box is selected by default.

   The AKMs are displayed in the **Auth Key Mgmt (AKM)** section.

6. In the **Auth Key Mgmt (AKM)** section, check either the **SAE-EXT-KEY** check box or the **FT + SAE-EXT-KEY** check box, or select both the AKMs.

   Complete these steps:

   a) Enter the **Anti Clogging Threshold** value. Valid range is 0 to 3000; default value is 1500.
   b) Enter the number of allowed **Max Retries**. Valid range is 1 to 10; default value is five.
   c) Enter the **Retransmit Timeout** value in seconds. Valid range is 1 to 10000; default value is 400.
   d) From the drop-down lists, select the **PSK Format** and the **PSK Type**.
   e) Enter the **Pre-Shared Key**.
   f) From the **SAE Password Element** drop-down list, select one of these methods to generate the SAE password element:
      - **Both H2E and HnP**: The password element is generated from both Hash-to Element (H2E) and Hunting and Pecking methods (HnP). This is the default option.
      - **Hash to Element only**: In this method, the secret password element used in the SAE protocol is generated from a password. H2E is based on a noniterative algorithm that is more computationally efficient and provides robust resistance to side-channel attacks. If selected, HnP is disabled.
      - **Hunting and Pecking only**: This method uses the iterative looping algorithm to generate the password element. As this method is prone to attacks, we recommend that you use the other two methods. If you select the Hunting and Pecking only option, H2E is disabled.

> **Note**
>
> SAE-EXT-KEY and FT + SAE-EXT-KEY requires the password element mode to be **Both H2E and HnP** or **Hash to Element only**.

If you select an option with WPA2, configure MPSK by completing these steps:

a. In the **MPSK Configuration** section, check the **Enable MPSK** check box.

b. In the **Auth Key Mgmt** section, choose the **PSK Format** (default is ASCII), PSK Type (default is unencrypted), and enter the **Pre-Shared Key**.

c. In the **MPSK Configuration** section, click **Add**.

Ensure that there are no warnings or error messages in the **Auth Key Mgmt** section, related to encryption and cipher combination.

d. Click **Apply**, and then click **Apply to Device**.

7. Click **Apply to Device**.

The WLAN is now configured with enhanced SAE-EXT-KEY authentication key management, providing robust, standards-based password security for WPA2 or WPA3 wireless networks.

Configure SAE-EXT-KEY AKMs (GUI)

Set up SAE-EXT-KEY Authentication and Key Management (AKM) on a WLAN to enhance wireless security with advanced password-based authentication protocols.

Use this task when configuring a Wi-Fi network profile requiring the SAE-EXT-KEY or Fast Transition (FT) + SAE-EXT-KEY AKMs for WPA2 or WPA3-enabled WLANs. These settings enable secure password management and provide robust wireless security.

Before you begin

- Ensure that the WLAN controller supports SAE-EXT-KEY and FT + SAE-EXT-KEY AKMs.
- Verify that WPA3 security policies are enabled.
- Gather values for anti-clogging threshold, maximum retries, retransmit timeout, and the required pre-shared key (PSK).

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the WLAN profile and SSID. Enter the WLAN configuration mode.

Example

```
Device(config)# wlan wlan-profile-name wlan-id ssid-name
```

3. Disable adaptive 802.11r.

   Example

   ```
   Device(config-wlan)# no security ft adaptive
   ```

4. Configure the pre-shared key (PSK) either in the ASCII format or the HEX format. with the **security wpa psk set-key {ascii | hex} {0 | 8}** *pre-shared-key* command.

   Example

   ```
   Device(config-wlan)# security wpa psk set-key ascii 0 123456789
   ```

5. Disable security Auth Key Management (AKM) for 802.1X.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

6. Configure the SAE-EXT-KEY AKM support.

   Example

   ```
   Device(config-wlan)# security wpa akm sae ext-key
   ```

7. Configure WPA3 support.

   Example

   ```
   Device(config-wlan)# security wpa wpa3
   ```

8. Configure WPA2 and GCMP-256 cipher support.

   Example

   ```
   Device(config-wlan)# security wpa wpa2 ciphers gcmp256
   ```

The WLAN is now configured with enhanced SAE-EXT-KEY authentication key management, providing robust, standards-based password security for WPA2 or WPA3 wireless networks.

Configure FT-SAE-EXT-KEY AKMs (CLI)

Enable FT-SAE-EXT-KEY Authentication and Key Management (AKM) for a WLAN profile on your device using CLI.

Use this task when you want to support fast transition (FT) and SAE AKM with extended keys for secure wireless authentication.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN profile and SSID. Enter the WLAN configuration mode.

   Example

   ```
   Device(config)# wlan wlan-profile-name wlan-id ssid-name
   ```

3. Configure fast transition

   Example

   ```
   Device(config-wlan)# security ft adaptive
   ```

4. Configure the pre-shared key (PSK) either in the ASCII format or the HEX format using the **security wpa psk set-key {ascii | hex} {0 | 8}** *pre-shared-key* command.

   Example

   ```
   Device(config-wlan)# security wpa psk set-key ascii 0 123456789
   ```

5. Disable security Auth Key Management (AKM) for 802.1X.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

6. Configure the FT-SAE-EXT-KEY AKM support.

   Example

   ```
   Device(config-wlan)# security wpa akm ft sae ext-key
   ```

7. Configure WPA3 support.

   Example

   ```
   Device(config-wlan)# security wpa wpa3
   ```

8. Configure WPA2 and GCMP-256 cipher support.

   Example

   ```
   Device(config-wlan)# security wpa wpa2 ciphers gcmp256
   ```

The WLAN profile is configured to use FT-SAE-EXT-KEY AKM for secure, fast wireless transitions supporting WPA3 and WPA2 or GCMP-256.

Verify the SAE-EXT-KEY AKM support

Describes the verification of the SAE-EXT-KEY AKM Support

Summary of SAE-EXT-KEY AKMs

To view the summary of the SAE-EXT-KEY AKMs, use this command:

```
Device# show wlan summary
Number of WLANs: 5

ID   Profile Name                      SSID                              Status
 2.4GHz/5GHz Security                            6GHz Security
-------------------------------------------------------------------------------------------
1    wpa3-sae_profile                  wpa3-sae                          UP
 [WPA3][SAE][AES]                                [WPA3][SAE][AES]
2    wpa3-sae-ext_profile              wpa3-sae-ext                      UP
 [WPA3][SAE-EXT-KEY][GCMP256]                    [WPA3][SAE-EXT-KEY][GCMP256]
3    wpa3-sae-ext-mab_profile          wpa3-sae-ext-mab                  UP
 [WPA3][MAB][SAE-EXT-KEY][GCMP256]
[WPA3][MAB][SAE-EXT-KEY][GCMP256]
4    wpa3-sae-ext-webauth_profile      wpa3-sae-ext-webauth_profile      UP
 [WPA3][SAE-EXT-KEY][Webauth][GCMP256]
[WPA3][SAE-EXT-KEY][Webauth][GCMP256]
5    wpa3-sae-ext-mab-webauth_profile wpa3-sae-ext-mab-webauth_profile UP
 [WPA3][MAB][SAE-EXT-KEY][Webauth][GCMP256]
[WPA3][MAB][SAE-EXT-KEY][Webauth][GCMP256]
6    wpa3-ft-sae_profile               wpa3-ft-sae                       UP
 [WPA3][FT + SAE][AES]                           [WPA3][FT + SAE][AES]
7    wpa3-ft-sae-ext_profile           wpa3-ft-sae-ext                   UP
 [WPA3][FT + SAE-EXT-KEY][GCMP256]               [WPA3][FT +
SAE-EXT-KEY][GCMP256]
8    wpa3-ft-sae-ext-mab_profile       wpa3-ft-sae-ext-mab               UP
 [WPA3][MAB][FT + SAE-EXT-KEY][GCMP256]          [WPA3][MAB][FT +
SAE-EXT-KEY][GCMP256]
9    wpa3-ft-sae-ext-webauth_profile  wpa3-ft-sae-ext-webauth           UP
 [WPA3][FT + SAE-EXT-KEY][Webauth][GCMP256]      [WPA3][FT +
SAE-EXT-KEY][Webauth][GCMP256]
10   wpa3-ft-sae-ext-mab-webauth_pro  wpa3-ft-sae-ext-mab-webauth      UP
 [WPA3][MAB][FT + SAE-EXT-KEY][Webauth][GCMP256] [WPA3][MAB][FT +
SAE-EXT-KEY][Webauth][GCMP256]
```

SAE-EXT-KEY and FT-SAE-EXT-KEY AKM in WLAN Profiles

To view the details of the SAE-EXT-KEY and FT-SAE-EXT-KEY AKMs, use these commands:

```
Device# show wlan name wpa3-sae-ext-key-profile
WLAN Profile Name     : wpa3-sae-ext-key-profile
================================================
Identifier                                : 2
Description                               :
Network Name (SSID)                       : wpa3-sae-ext-key
<...>
Security
    802.11 Authentication                 : Open System
    Static WEP Keys                       : Disabled
    Wi-Fi Protected Access (WPA/WPA2/WPA3)     : Enabled
        WPA (SSN IE)                      : Disabled
```

```
        WPA2 (RSN IE)                           : Disabled
        WPA3 (WPA3 IE)                          : Enabled
            AES Cipher                          : Disabled
            CCMP256 Cipher                      : Disabled
            GCMP128 Cipher                      : Disabled
            GCMP-256 Cipher                      : Enabled
        Auth Key Management
            802.1x                              : Disabled
            PSK                                 : Disabled
            CCKM                                : Disabled
            FT dot1x                            : Disabled
            FT PSK                              : Disabled
            FT SAE                              : Disabled
            FT SAE-EXT-KEY                      : Disabled
            Dot1x-SHA256                        : Disabled
            PSK-SHA256                          : Disabled
            SAE                                 : Disabled
            SAE-EXT-KEY                         : Enabled
            OWE                                 : Disabled
            SUITEB-1X                           : Disabled
            SUITEB192-1X                        : Disabled
    SAE PWE Method                              : Hash to Element, Hunting and
 Pecking(H2E-HNP)
.
.
.
```

```
Device# show wlan name wpa3-ft-sae-ext-key-profile
WLAN Profile Name     : wpa3-ft-sae-ext-key-profile
===============================================
Identifier                                      : 7
Description                                     :
Network Name (SSID)                             : wpa3-ft-sae-ext-key
<...>
Security
    802.11 Authentication                       : Open System
    Static WEP Keys                             : Disabled
    Wi-Fi Protected Access (WPA/WPA2/WPA3)      : Enabled
        WPA (SSN IE)                            : Disabled
        WPA2 (RSN IE)                           : Disabled
        WPA3 (WPA3 IE)                          : Enabled
            AES Cipher                          : Disabled
            CCMP256 Cipher                      : Disabled
            GCMP128 Cipher                      : Disabled
            GCMP-256 Cipher                      : Enabled
        Auth Key Management
            802.1x                              : Disabled
            PSK                                 : Disabled
            CCKM                                : Disabled
            FT dot1x                            : Disabled
            FT PSK                              : Disabled
            FT SAE                              : Disabled
            FT SAE-EXT-KEY                      : Enabled
            Dot1x-SHA256                        : Disabled
            PSK-SHA256                          : Disabled
            SAE                                 : Disabled
            SAE-EXT-KEY                         : Disabled
            OWE                                 : Disabled
            SUITEB-1X                           : Disabled
            SUITEB192-1X                        : Disabled
    SAE PWE Method                              : Hash to Element, Hunting and
```

```
 Pecking(H2E-HNP)
.
.
.
```

Cipher and AKMs based on client MAC address

To view the details of the cipher and AKMs based on the client MAC address, use this command:

```
Device# show wireless client mac-address 3089.4aXX.f0XX detail
Client MAC Address : 3089.4aXX.f0XX
.
.
.
Policy Type : WPA3
Encryption Cipher : GCMP-256
Authentication Key Management : SAE-EXT-KEY
.
.
.
Client MAC Address : 3089.4aXX.f0XX
.
.
.
Policy Type : WPA3
Encryption Cipher : GCMP-256
Authentication Key Management : FT-SAE-EXT-KEY
.
.
.
```

AKM support statistics report

To view the AKM support statistics report, use this command:

```
Device# show wireless stats client detail
Total WPA3 SAE attempts                          :71
Total WPA3 SAE successful authentications        : 9
  Total SAE-EXT-KEY successful authentications   : 3
Total WPA3 SAE authentication failures           : 22
  Total incomplete protocol failures             : 0
Total WPA3 SAE commit messages received          : 126
Total WPA3 SAE commit messages rejected                                  : 58
  Total unsupported group rejections                                     : 0
  Total PWE method mismatch for SAE Hash to Element commit received      : 0
  Total PWE method mismatch for SAE Hunting And Pecking commit received : 0
Total WPA3 SAE commit messages sent              : 175
Total WPA3 SAE confirm messages received         : 13
Total WPA3 SAE confirm messages rejected         : 4
  Total WPA3 SAE message confirm field mismatch  : 4
  Total WPA3 SAE confirm message invalid length  : 0
Total WPA3 SAE confirm messages sent             : 13
Total WPA3 SAE Open Sessions                     : 0
Total SAE Message drops due to throttling        : 0
Total WPA3 SAE Hash to Element commit received   : 111
Total WPA3 SAE Hunting and Pecking commit received : 15
```

AP beacon protection

Outlines AP beacon protection features, guiding users through configuration using GUI and CLI, and presents verification steps to confirm beacon protection status.

An AP beacon protection is a WLAN security feature that

- prevents attackers from modifying AP beacons and their capabilities
- uses a Beacon Integrity Key to generate a Message Integrity Code (MIC) in AP beacons, and
- enables clients to validate beacon authenticity and reject APs sent by attackers.

Expanded explanation

AP beacon protection works by transmitting a Beacon Integrity Key during the WPA2 or WPA3 four-way handshake. Genuine APs use this key to produce a MIC in their beacon frames. Clients verify the MIC to ensure beacons are authentic. The system rejects any beacon that fails validation.

A client device receives a beacon from an AP and checks the MIC. If the MIC is valid, the client associates with the AP; if not, the client ignores the beacon, preventing rogue AP attack

Configure AP beacon protection (GUI)

You can perform this task to enable AP beacon protection on your WLAN using the CLI and WPA3 security.

Before you begin

- Ensure that you are configuring a WLAN profile that supports WPA3 AKM (for example, SAE, FT-SAE, OWE, DOT1X-SHA256, and so on).
- Replace the sample AKMs with the specific WPA3 AKM you want to use.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.

   The **Add WLAN** window is displayed.
3. In the **General** tab, enter the **Profile Name**, **SSID**, and the **WLAN ID**.
4. Choose **Security** > **Layer 2**, select either the **WPA2 + WPA3** option or the **WPA3** option.

   The **Beacon Protection** check box appears in the WPA parameters section when you enable the WPA3 policy.
5. Check the **Beacon Protection** check box.

   Note

   Protected Management Frame (PMF) is required for Beacon Protection to be enabled.
6. Click **Apply to Device**.

You have enabled AP beacon protection on your WLAN. This provides enhanced protection against beacon frame attacks when WPA3 is used.

What to do next

=

Configure AP beacon protection (CLI)

You can perform this task to enable AP beacon protection on your WLAN using the CLI and WPA3 security.

Before you begin

- Ensure that you are configuring a WLAN profile that supports WPA3 AKM (for example, SAE, FT-SAE, OWE, DOT1X-SHA256, and so on).
- Replace the sample AKMs with the specific WPA3 AKM you want to use.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN profile and SSID. Enter the WLAN configuration mode.

   Example

   ```
   Device(config)# wlan wlan-profile-name wlan-id ssid-name
   ```

3. Disable adaptive 802.11r.

   Example

   ```
   Device(config-wlan)# no security ft adaptive
   ```

4. Configure the pre-shared key (PSK) either in the ASCII format or the HEX format using the **security wpa psk set-key {ascii | hex} {0 | 8}** *pre-shared-key* command.

   Example

   ```
   Device(config-wlan)# security wpa psk set-key ascii 0 123456789
   ```

5. Disable security Authentication and Key Management (AKM) for 802.1X.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

6. Configure SAE support.

   Example

   ```
   Device(config-wlan)# security wpa akm sae
   ```

7. Configure WPA3 support.

Example

```
Device(config-wlan)# security wpa wpa3
```

8. Configure AP beacon protection.

Example

```
Device(config-wlan)# security wpa wpa3 beacon-protection
```

9. Disable WPA2 security.

Example

```
Device(config-wlan)# no security wpa wpa2
```

10. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

You have enabled AP beacon protection on your WLAN. This provides enhanced protection against beacon frame attacks when WPA3 is used.

Verify AP beacon protection

Describes the show command to verify AP beacon protection

To view the AP beacon protection details, use this command:

```
Device# show wlan name wl-sae
WLAN Profile Name     : wl-sae
================================================
Identifier                                       : 7
Description                                      :
Network Name (SSID)                              : wl-sae
<...>
Security
    Security-2.4GHz/5GHz
        <...>
        Beacon Protection                        : Enabled
    Security-6GHz
        <...>
        Beacon Protection                        : Enabled
<...>
```

Multiple cipher support per WLAN

Details configuring multiple ciphers per WLAN, providing instructions for setup via GUI and CLI to enhance flexibility and security in wireless network deployments.

Multiple cipher support per WLAN is a WLAN security feature that

- allows multiple encryption ciphers to operate on a single WLAN,

- enables compatibility with various Authentication and Key Management (AKM)options, and
- supports broader device interoperability by accommodating mixed device requirements.

Mapping of cipher suites

Until Cisco IOS XE 17.14.1, only a single cipher was allowed in a WLAN, limiting the supported AKMs combinations. Only CCMP-128 cipher was used with multiple AKMs, while GCMP-128 was tightly coupled with the Suite-B-1x AKM and CCMP-256 or GCMP-256 were tightly coupled with the Suite-B-192-1x AKM.

With Cisco IOS XE 17.15.1 and later, support for multiple AKMs and cipher combinations on a single WLAN is introduced. One WLAN can now serve devices requiring different ciphers, such as GCMP-256 and CCMP-128.

The configured cipher suite(s) for a WLAN is mapped to the Pairwise Cipher Suite, Group Cipher Suite, and Management Cipher Suite broadcasted in the Beacons or Probe Responses.

Table 116: Mapping of cipher suites

Configured Cipher Suite	Pairwise Cipher Suite	Group Cipher Suite	Management Cipher Suite
CCMP-128 only	CCMP-128	CCMP-128	BIP-CMAC-128
GCMP-256 only	GCMP-256	GCMP-256 management	BIP-GMAC-256
CCMP-128 + GCMP-256	CCMP-128 or GCMP-256 (client chooses)	CCMP-128	BIP-CMAC-128

Configure multiple ciphers (GUI)

Enable support for multiple security ciphers on a WLAN using CLI commands.

Use these steps to configure a wireless LAN for multiple WPA2 ciphers, allowing enhanced security and compatibility for connecting devices.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.

   The **Add WLAN** window is displayed.
3. In the **General** tab, enter the **Profile Name**, **SSID**, and the **WLAN ID**.
4. Choose **Security** > **Layer2**, select one of the following options:
   - **WPA + WPA2**
   - **WPA2 + WPA3**
   - **WPA3**

   The **AES(CCMP128)** cipher is selected by default.

   The **Auth Key Mgmt (AKM)** section will be populated with the possible AKMs that are supported by the cipher that is selected in the **WPA2/WPA3 Encryption** section. Valid cipher and AKM combinations are displayed in the **Auth Key Mgmt (AKM)** section.
5. In the **WPA2/WPA3 Encryption** check the **GCMP256** check box, or the **AES(CCMP128)** check box, or a combination of both these check boxes, to display the AKMs in the same WLAN.

6. In the **Auth Key Mgmt (AKM)** section, check the AKM check boxes to enable the required AKMs. At least one AKM should be enabled.
7. Click **Apply to Device**.

Multiple WPA2 ciphers are now configured for the WLAN. This enhances network security and allows flexibility for supported clients.

Configure multiple ciphers (CLI)

Enable support for multiple security ciphers on a WLAN using CLI commands.

Use these steps to configure a wireless LAN for multiple WPA2 ciphers, allowing enhanced security and compatibility for connecting devices.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN profile and SSID. Enter the WLAN configuration mode.

   Example

   ```
   Device(config)# wlan wlan-profile-name wlan-id ssid-name
   ```

3. Disable adaptive 802.11r.

   Example

   ```
   Device(config-wlan)# no security ft adaptive
   ```

4. Configure the pre-shared key (PSK) either in the ASCII format or the HEX format using the **security wpa psk set-key {ascii | hex} {0 | 8}** *pre-shared-key* command.

   Example

   ```
   Device(config-wlan)# security wpa psk set-key ascii 0 123456789
   ```

5. Disable security AKM for 802.1X.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

6. Configure SAE support.

   Example

   ```
   Device(config-wlan)# security wpa akm sae
   ```

7. Configure SAE-EXT-KEY AKM support.

   Example

   ```
   Device(config-wlan)# security wpa akm sae ext-key
   ```

8. Configure WPA3 support.

   Example

   ```
   Device(config-wlan)# security wpa wpa3
   ```

9. Configure WPA2 cipher support. In this example, CCMP-128 cipher is configured using the **security wpa wpa2 ciphers {aes | ccmp256 | gcmp128 | gcmp256}** command.

   Example

   ```
   Device(config-wlan)# security wpa wpa2 ciphers aes
   ```

10. Configure another WPA2 cipher (multiple cipher support). In this example, GCMP-256 cipher is configured using the **security wpa wpa2 ciphers {aes | ccmp256 | gcmp128 | gcmp256}** command.

    Example

    ```
    Device(config-wlan)# security wpa wpa2 ciphers gcmp256
    ```

Multiple WPA2 ciphers are now configured for the WLAN. This enhances network security and allows flexibility for supported clients.

OWE support with GCMP-256 cipher

Explains OWE support leveraging the GCMP-256 cipher, including procedures for configuring Opportunistic Wireless Encryption AKM through GUI and CLI for improved wireless security.

In Cisco IOS XE 17.14.1 and earlier, OWE is supported only with the CCMP-128 cipher. Starting with Cisco IOS XE 17.15.1, you can configure OWE with both CCMP-128 and GCMP-256. When you configure both, the client selects the cipher suite during association.

Configure OWE AKM (GUI)

Enable Opportunistic Wireless Encryption (OWE) for a WLAN to enhance wireless security by allowing encrypted connections without requiring pre-shared keys.

Use OWE to provide secure, open wireless access using the WLAN configuration interface, leveraging WPA3 security and modern encryption standards for wireless networks.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.

   The **Add WLAN** window is displayed.

3. In the **General** tab, enter the **Profile Name**, **SSID**, and the **WLAN ID**.
4. Choose **Security** > **Layer 2** and click the **WPA3** option.
5. In the **WPA2/WPA3 Encryption** section, check the **GCMP256** check box, or the **AES(CCMP128)** check box, or a combination of both these check boxes. The **AES(CCMP128)** check box is selected by default.
6. In the **Fast Transition** section, from the **Status** drop-down list, select **Disabled**
7. In the **Auth Key Mgmt (AKM)** section, check the **OWE** check box.

   The **Transition Mode WLAN ID** field is displayed.
8. Enter the **Transition Mode WLAN ID**. The transition-mode WLAN ID ranges are the same as the WLAN ID ranges, that is, the valid range is between 0 and 4096.
9. Click **Apply to Device**.

OWE is enabled for the selected WLAN, allowing clients to connect securely without the need for a shared password.

Configure OWE AKM (CLI)

Describes the steps to configure the OWE AKM.

Enable Opportunistic Wireless Encryption (OWE), Authentication and Key Management (AKM), for a Wireless Local Area Network (WLAN) using the CLI.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Configure the WLAN profile and SSID. Enters the WLAN configuration mode.

   Example

   ```
   Device(config)# wlan wlan-profile-name wlan-id ssid-name
   ```
3. Disable adaptive 802.11r.

   Example

   ```
   Device(config-wlan)# no security ft adaptive
   ```
4. Configure the OWE AKM.

   Example

   ```
   Device(config-wlan)# security wpa akm owe
   ```

The WLAN profile is now configured to use Opportunistic Wireless Encryption AKM.

126 IP Source Guard

Topics:

- IP source guard

This chapter explains how to enable the feature on specific WLANs using GUI or commands, detailing IP/MAC binding behavior, configuration steps, and effects on forwarding to prevent IP address spoofing.

IP source guard

Explains how IP source guard features secure wireless networks by preventing unauthorized packets based on IP/MAC bindings.

A set of IP source guard features are Layer 2 security mechanisms that

- prevent the controller from forwarding packets with source IP addresses unknown to the wireless controller
- require explicit configuration per WLAN and are not enabled by default, and
- maintain an IP/MAC binding table to track and authorize valid wireless clients.

It supports both IPv4 and IPv6 wireless clients. The IPSG feature prevents the wireless controller from forwarding packets with unknown source IP addresses. This security feature is disabled by default. You must configure it explicitly for each WLAN. When you enable this feature, all wireless clients on the WLAN inherit the security setting.

Using the IP/MAC binding table, the wireless controller keeps track of IP and MAC address binding information for all wireless clients. The wireless controller collects binding information as part of the IP learning process. When this feature is enabled on a WLAN, the wireless controller forwards incoming packets from wireless clients only if a matching binding table entry exists for the source IP and MAC address combination. If the entry does not exist, the controller drops the packets.

Configure IP source guard (GUI)

Use this procedure to configure IP source guard using the GUI.

Enable IP source guard to prevent IP address spoofing on the selected WLAN using the GUI.

Before you begin

Ensure you have administrative access to the GUI. Confirm that the WLAN exists and is configured.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click on the WLAN.
3. In the **Advanced** tab, check the **IP Source Guard** checkbox.
4. Click **Update & Apply to Device**.

Configure IP source guard (CLI)

Use this procedure to configure IP source guard using commands.

Use this procedure to configure IP source guard on a WLAN using commands to protect against spoofed IP addresses.

Enabling IP source guard increases network security by ensuring that only valid IP sources are allowed on the WLAN.

Before you begin

Ensure you have administrator access to the device CLI. Have the WLAN name and ID information available.

Procedure

1. Specify the WLAN name and ID to use.

Example

```
Device(config)# wlan profile-name wlan-id ssid
```

> **Note**
>
> If a WLAN is not already configured, this step creates the WLAN.

2. Disable the WLAN.

   Example

   ```
   Device(config-wlan)# shutdown
   ```

3. Enable the IP source guard feature.

   Example

   ```
   Device(config-wlan)# ip verify source mac-check
   ```

4. Enable the WLAN.

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

---

127 Traffic Filtering on AP by Source IP ACL

Topics:

- Traffic filtering on AP by source IP ACL

Traffic filtering on AP by source IP ACL

Traffic filtering on access points by source IP ACL enables administrators to control network access by allowing or blocking incoming data packets based on their source IP address. This feature applies access control rules, managed by the controller, directly to the AP and supports both FlexConnect and Local modes for enhanced network security.

This feature enables APs to filter incoming data packets based on their source IP address. This is achieved when the controller pushes Access Control List (ACL) rules to the AP.

Its primary purpose is to:

- act as the listening device filtering incoming packets based on source IP addresses, while the controller manages and pushes the relevant ACL rules to the AP.
- enhance the security of AP services by allowing administrators to restrict access to trusted sources or block known malicious entities, regardless of whether the AP is operating in flex-connect mode or local mode.

This feature is supported in both FlexConnect mode and Local mode.

Feature History for Traffic Filtering on AP by Source IP ACL

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 117: Feature history for traffic filtering on AP by source IP ACL

Release	Feature Information
Cisco IOS XE 17.18.1	This feature enables APs to filter incoming data packets based on their source IP address. This is achieved when the controller pushes Access Control List (ACL) rules to the AP. This feature is supported in both FlexConnect mode and Local mode.

Configure AP management ACL on controller

Describes the CLI steps to define and configure ACL on the controller.

Follow these steps to define the AP management ACL rules, configure ACL on the controller, attach the site tag to the AP profile, and map the site tag to the AP.

> **Note**
>
> - The default action for AP MGMT ACL is allow. Therefore, there is no need to add **allow all** at the end of the ACL.
> - Ensure not to block the UDP source ports 5246/5247, or else the AP will lose CAPWAP registering, immediately after TLV is pushed to the AP. The only way to recovery it is to clear the configuration on the controller and factory reset the AP.

Define AP management ACL rules

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Define an extended IP access-list.

   Example

   ```
   Device(config)# ip access-list extended deny-host-ssh
   ```

   ```
   Device(config)# ipv6 access-list deny-host-ssh-v6
   ```

3. Define rules to deny SSH connection from IPv4 and IPv6 source addresses.

   Example

   ```
   Device(config-ext-nacl)# 10 deny tcp host 100.100.202.1 any eq 22
   Device(config-ext-nacl)# 11 deny tcp host 100.100.202.2 any eq 22
   ```

   ```
   Device(config-ipv6-acl)# sequence 10 deny tcp host 2001:1202:DED::1 any eq
   22
   ```

Rules are defined for AP management ACL.

Example

```
Device# configure terminal
Device(config)# ip access-list extended deny-host-ssh
Device(config-ext-nacl)# 10 deny tcp host 100.100.202.1 any eq 22
Device(config-ext-nacl)# 11 deny tcp host 100.100.202.2 any eq 22
Device(config)# ipv6 access-list deny-host-ssh-v6
Device(config-ipv6-acl)# sequence 10 deny tcp host 2001:1202:DED::1 any eq 22
```

Configure AP management ACL on the controller

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the AP profile.

   Example

   ```
   Device(config)# ap profile flex-ap-profile
   ```

3. Configure the AP management access list.

   Example

   ```
   Device(config-ap-profile)# access-list {access-list-v4 | access-list-v6}
   access-list-name
   ```

4. Exit sub-mode.

   Example

   ```
   Device(config-ap-profile)# exit
   ```

AP management ACL is configured on the controller.

Example

```
Device# configure terminal
Device(config)# ap profile flex-ap-profile
Device(config-ap-profile)# access-list access-list-v4 deny-host-ssh
Device(config-ap-profile)# access-list access-list-v6 deny-host-ssh-v6
Device(config-ap-profile)# exit
```

What to do next

Attach the site tag to the AP profile and then map the site tag to the AP. For example:

```
Device(config)# wireless tag site flex-site-tag
Device(config-site-tag)# description "flex site tag"
Device(config-site-tag)# ap-profile flex-ap-profile
Device(config-site-tag)# no local-site

Device(config)# ap 48XX.d5XX.e3XX
Device(config-ap-tag)# site-tag flex-site-tag
```

Verify mapped ACLs

Describes the show commands to verify the mapped ACLs.

Verify mapped ACL details

To verify the detailed AP profile configuration with mapped ACLs, on the controller, run the **show ap profile name** *ap-profile-name* **detailed** command:

```
Device# show ap profile name flex-ap-profile etailed
AP Profile Name                : flex-ap-profile
Description                    :
Country code                   : Not configured
Stats Timer                    : 180
Link Latency                   : DISABLED
Data Encryption                : DISABLED
LED State                      : ENABLED
…
…
…

AP broken antenna detection :
  Status                       : DISABLED
RLAN Configurations :
  Fast Switching                : DISABLED
Onboarding configuration       : Unicast
Pressure sensor :
  Mode                          : AUTO
Accelerometer Sensor           : ENABLED
AP statistics                   : DISABLED
Radio statistics                : DISABLED
AP Deployment Mode                   : Not Configured
Static path MTU                   : 0
AP Management IPv4 ACL               : deny-host-ssh
AP Management IPv6 ACL               : deny-host-ssh-v6
AP Profile Name                : test_ap_profile
Description                    :
Country code                   : Not configured
Stats Timer                    : 180
Link Latency                   : DISABLED
Data Encryption                : DISABLED
LED State                      : ENABLED
…
…
…
AP broken antenna detection :
  Status                       : DISABLED
RLAN Configurations :
  Fast Switching                : DISABLED
Onboarding configuration       : Unicast
Pressure sensor :
  Mode                          : AUTO
Accelerometer Sensor           : ENABLED
AP statistics                   : DISABLED
Radio statistics                : DISABLED
AP Deployment Mode                   : Not Configured
Static path MTU                   : 0
AP Management IPv4 ACL               : deny-host-ssh
AP Management IPv6 ACL               : deny-host-ssh-v6
```

Verify AP management ACL on AP

To verify the IPv4 AP management ACL on the AP, run the **show ip access-lists**command.

```
Cisco-AP# show ip access-lists
AP MGMT ACLs:
Extended IP access list deny-host-ssh
   1 deny tcp 100.100.202.1 0.0.0.0 range 0 65535 any eq 22
   2 deny tcp 100.100.202.2 0.0.0.0 range 0 65535 any eq 22
```

To verify IPv6 AP management ACL on the AP, run the **show ipv6 access-list** command.

```
Cisco-AP# show ipv6 access-list
AP MGMT ACLs:
Extended IPv6 access list deny-host-ssh-v6
    1 deny tcp host 2001:1202:ded::1 range 0 65535 any eq 22
```

Verify AP host firewall ACL drop count

To verify the statistics of the ACL drop count on the AP, run the **show datapath gateway 1213 statistics** command.

```
Cisco-AP# show datapath gateway l2l3 statistics
error stats:
Eth-Errs IP-Errs Drops MAC-filters Drop-by-ACL
       0       0     0           0           0
to host stats:
  IPv4   IPv6   ARP   Other
  108115 98915  46130     0
from host stats:
  IPv4   IPv6  ARP  Other
  136571 4288 12243     0
to routing stats:
IPv4 IPv6 ARP
   0    0   0
```

128 802.11w

Topics:

- Protected management frames with 802.11w
- Configure 802.11w

Protected management frames with 802.11w

Introduces protected management frames in 802.11w, explains IGTK and BIP integrity protocols, details SA teardown protection and associated procedures, and summarizes prerequisites and restrictions for deploying 802.11w.

A protected management frame is a wireless security feature that

- uses the 802.11w protocol to safeguard management frames
- prevents spoofing and forgery of authentication, de-authentication, association, and disassociation frames, and
- enhances the overall security of Wi-Fi networks by protecting key network management actions from attack.

While data frames can be encrypted, management frames were traditionally sent in the clear, making them vulnerable to interception and forgery. The 802.11w standard addresses this vulnerability by requiring cryptographic protection for certain management frames between client and access point.

An attacker attempting to disconnect a legitimate client by sending forged disassociation frames will fail on networks supporting protected management frames, as only cryptographically validated frames are accepted.

Types of management frames protected by 802.11w

The 802.11w protocol protects certain management frames by using the Protected Management Frames (PMF) service. These frames are classified as robust management frames and include:

- Disassociation frames
- De-authentication frames
- Robust Action frames

Robust Action frames protected by 802.11w include:

- Spectrum Management
- Quality of Service (QoS)
- Direct Link Setup (DLS)
- Block acknowledgement
- Radio Measurement
- Fast Basic Service Set (BSS) Transition
- Security Association (SA) Query
- Protected Dual of Public Action
- Vendor-specific Protected

Frames not included in this list are not protected by 802.11w.

Protections offered by 802.11w

When 802.11w is implemented, these protections are provided:

- **Client protection:** The AP adds cryptographic protection to de-authentication and dissociation frames, preventing spoofing in DOS attacks.
- **Infrastructure protection**: Security Association (SA) teardown protection is implemented using Association Comeback Time and SA-Query procedures to prevent spoofed association requests from disconnecting connected clients.

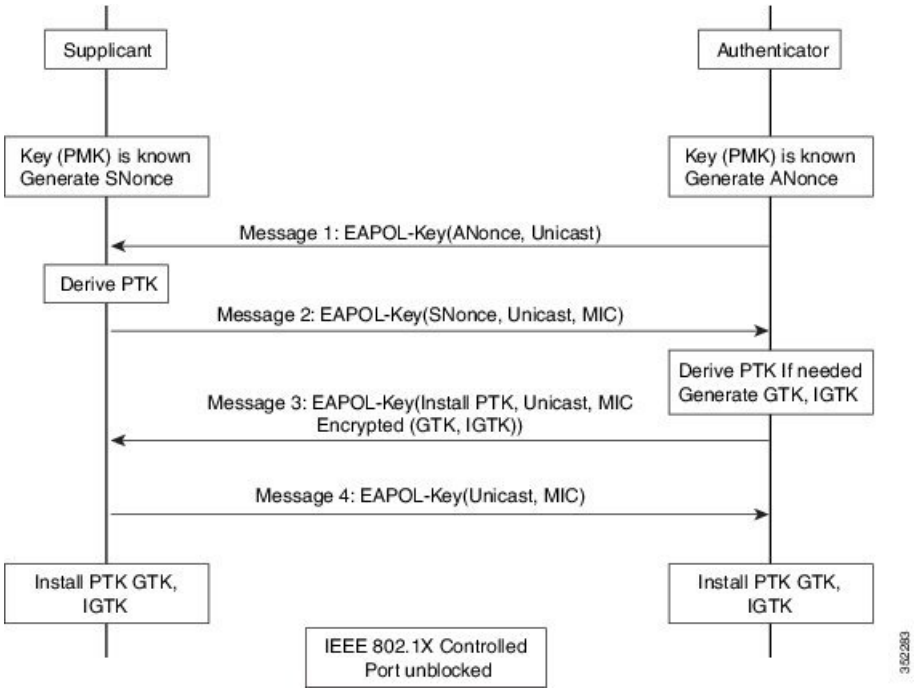
Integrity Group Temporal Key (IGTK)

The Integrity Group Temporal Key (IGTK) is a cryptographic key introduced in 802.11w to protect broadcast and multicast robust management frames.

An integrity group temporal key (IGTK) is a wireless security mechanism that

- protects broadcast and multicast robust management frames.
- uses random values assigned by the authenticator station (such as a wireless LAN controller), and
- secures MAC management protocol data units (MMPDUs) in 802.11w networks.

Figure 18: IGTK exchange in four-way handshake



802.11w introduced IGTKs to enhance the security of management frames in wireless networks.

How IGTK is used in 802.11w management frame protection

When you enable management frame protection, the AP encrypts the group temporal key (GTK) and IGTK values in an Extensible Authentication Protocol over LAN-Key (EAPoL-Key) frame. The AP includes this frame in the third message of the four-way handshake.

- IGTK is exchanged during the four-way handshake process.
- If the AP changes the GTK later, it sends the new GTK and the new IGTK to your client device using the Group Key Handshake

Imagine a wireless network as a secured meeting room, with announcements broadcast to everyone inside. The IGTK acts like a special group password that lets only authorized members hear these important messages. The authenticator (such as a wireless LAN controller) is like the meeting organizer, who gives each member a random, unique password at the door. When the password changes for added security, the organizer discreetly shares the new password with all members so that only legitimate participants can continue to hear future group announcements.

Broadcast or multicast integrity protocol (BIP)

Broadcast/Multicast Integrity Protocol (BIP) is defined by 802.11w to provide data integrity and replay protection for broadcast and multicast robust management frames.

A broadcast or multicast integrity protocol (BIP) is a wireless security mechanism that

- ensures data integrity of broadcast and multicast robust management frames
- provides replay protection for these frames after you establish an Integrity Group Temporal Key Security Association (IGTKSA), and
- adds a message integrity code (MIC) calculated using the shared IGTK key.

SA teardown protection

SA teardown protection is a wireless network security mechanism that prevents unauthorized disconnections by verifying t2he authenticity of association requests.

SA teardown protection is a wireless network security mechanism that

- prevents spoofed or replay attacks from disconnecting already associated clients
- uses Association Comeback Time and an SA-Query procedure to verify the authenticity of association requests, and
- ensures the AP only accepts new associations after the original security association is proven invalid.

How association comeback time and SA query procedures work

The Association Comeback Time and SA Query procedures prevent replay attacks from tearing down active wireless client sessions.

This process describes how Association Comeback Time and SA Query procedures protect wireless client sessions from replay-based association teardown attacks."

Summary

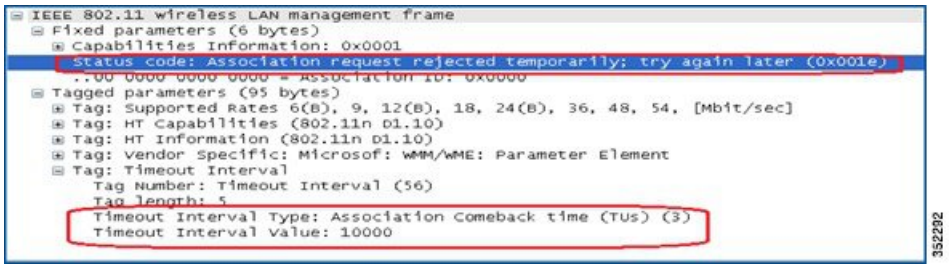
The key components involved in the process are:

- AP: implements security association (SA) teardown protection and manages association requests."
- Client device: Maintains a security association and sends and receives association and SA Query frames.
- SA Teardown Protection Mechanism: Provides the logic for handling replay and spoofed association attempts.

In this process, the AP and client device exchange association and SA Query frames. This exchange validates security associations and prevents unauthorized session teardown.

Workflow

Figure 19: Association Reject with Comeback Time



These stages describe how the Association Comeback Time and SA Query procedures operate to protect client sessions:

1. When an AP receives an Association Request from a client with an existing valid security association (SA) negotiated with 802.11w, the AP rejects the request with status code 30 (“Association request rejected temporarily; try again later”) and sends an Association Comeback Time.
2. The AP does not modify the existing association during the comeback interval.
3. If no ongoing SA Query session with the client exists, the AP sends SA Query requests repeatedly until it receives a valid SA Query response or the comeback time expires.
4. Receiving a matching SA Query response or a valid protected frame indicates a valid SA. The AP may then allow a new association attempt without more SA Query cycles.

Result

This process ensures that spoofed requests cannot disconnect valid clients, protecting against replay-based association teardown attacks.

Prerequisites for 802.11w

802.11w prerequisites.

- To configure 802.11w feature for optional and mandatory, you must have WPA and AKM configured.

> **Note**
>
> The RNS (Robust Secure Network) IE must be enabled with an AES Cipher.

Restrictions for 802.11w

Learn about restrictions for using 802.11w.

- You cannot use 802.11w with open WLANs, Wired Equivalent Privacy (WEP)-encrypted WLANs, or Temporal Key Integrity Protocol (TKIP)-encrypted WLANs.
- You can use 802.11w with Protected Management Frames (PMF) for non-Apple clients. For Apple iOS version 11 and earlier, request a fix from Apple to resolve association issues
- When clients do not use 802.11w PMF, the controller ignores disassociation frames or deauthentication frames they send. If a client uses PMF, its entry is deleted immediately when the controller receives such a frame. This process helps prevent denial-of-service attacks by malicious devices, since frames without PMF are not secure.

Configure 802.11w

Provides instructions for configuring, disabling, and monitoring 802.11w, including step-by-step procedures for GUI-based and other configurations to ensure effective implementation and management of 802.11w features.

Configure 802.11w (GUI)

Configure 802.11w using GUI.

To protect management traffic against spoofing and replay attacks, configure 802.11w settings after enabling WPA and AKM.

Enable protected management frames (PMF) on a WLAN to strengthen security for management communications.

Before you begin

Ensure WPA and AKM are configured on the target WLAN.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add** to create WLANs.

   The **Add WLAN** page is displayed.
3. In the **Security** > **Layer2** tab, navigate to the **Protected Management Frame** section.
4. Choose **PMF** as *Disabled*, *Optional*, or *Required*. By default, the PMF is *disabled*.

   If you choose **PMF** as *Optional* or *Required*, you can view these fields:

   - **Association Comeback Timer**–Enter a value between 1 and 10 seconds to configure 802.11w association comeback time.
   - **SA Query Time**–Enter a value between 100 to 500 milliseconds. This is required for clients to negotiate 802.11w PMF protection on a WLAN.
5. Click **Save & Apply to Device**.

802.11w PMF is enabled with the parameters you selected. This strengthens the security of management frames on the WLAN.

Configure 802.11w (CLI)

Configure 802.11w using commands.

Configuring 802.11w improves security for management frames on wireless LANs.

Enable 802.11w Protected Management Frames (PMF) on a WLAN using CLI commands.

Before you begin

- Configure WPA.
- Configure AKM (Authentication and Key Management) on the WLAN.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Configure a WLAN and enters configuration mode.

   Example

   ```
   Device(config)# wlan profile-name wlan-id ssid
   ```
3. Configure 802.1x support using the **security wpa akm dot1x-sha256** command.

Example

```
Device(config-wlan)#security wpa akm dot1x-sha256
```

4. Configure the 802.11w association comeback time.

Example

```
Device(config-wlan)# security pmf association-comeback comeback-interval
```

Example

```
Device(config-wlan)# security pmf association-comeback 10
```

5. Require clients to negotiate 802.11w PMF protection on a WLAN.

Example

```
Device(config-wlan)# security pmf mandatory
```

6. Configure time interval identified in milliseconds before which the SA query response is expected.

Example

```
Device(config-wlan)# security pmf saquery-retry-time timeout
```

Example

```
Device(config-wlan)# security pmf saquery-retry-time 100
```

If the device does not get a response, another SQ query is tried.

802.11w Protected Management Frames are enabled and mandatory on the specified WLAN, providing improved protection of management frames.

Disable 802.11w

Disable 802.11w using commands.

Disabling 802.11w may be necessary for compatibility with legacy or non-compliant devices.

Prevent wireless clients from using 802.11w Protected Management Frames on a specified WLAN.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a WLAN and enters configuration mode.

Example

```
Device(config)# wlan profile-name wlan-id ssid
```

3. Disable 802.1x support using the **no security wpa akm dot1x-sha256** command.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x-sha256
   ```

4. Configure the 802.11w association comeback time.

   Example

   ```
   Device(config-wlan)# security pmf association-comeback comeback-interval
   ```

   ```
   Device(config-wlan)# security pmf association-comeback 10
   ```

5. Disable client negotiation of 802.11w PMF protection on a WLAN.

   Example

   ```
   Device(config-wlan)# no security pmf mandatory
   ```

6. Disable SQ query retry.

   Example

   ```
   Device(config-wlan)# no security pmf saquery-retry-time 100
   ```

802.11w PMF protection is disabled for the specified WLAN.

Monitor 802.11w

Monitor 802.11w using commands.

Use these commands to monitor 802.11w.

Procedure

---

1. Display the WLAN parameters on the WLAN.

   The PMF parameters are displayed.

   ```
   Device# show wlan name wlan-name
                           . . . .
                           . . . .
                           Auth Key Management
                           802.1x                                  : Disabled
                           PSK                                     : Disabled
                           CCKM                                    : Disabled
   ```

```
                        FT dot1x                                : Disabled
                        FT PSK                                  : Disabled
                        FT SAE                                  : Disabled
                        Dot1x-SHA256                            : Enabled
                        PSK-SHA256                              : Disabled
                        SAE                                     : Disabled
                        OWE                                     : Disabled
                        SUITEB-1X                               : Disabled
                        SUITEB192-1X                            : Disabled
                        CCKM TSF Tolerance                             : 1000
                       FT Support                                    : Adaptive

                        FT Reassociation Timeout                : 20
                        FT Over-The-DS mode                     : Enabled
                       PMF Support                                  : Required

                        PMF Association Comeback Timeout        : 1
                        PMF SA Query Time                       : 500
                        . . . .
                        . . . .
```

2. Display the summary of the 802.11w authentication key management configuration on a client.

```
Device# show wireless client mac-address mac-address detail
                        . . . .
                        . . . .
                        Policy Manager State: Run
                        NPU Fast Fast Notified : No
                        Last Policy Manager State : IP Learn Complete
                        Client Entry Create Time : 497 seconds
                        Policy Type : WPA2
                        Encryption Cipher : CCMP (AES)
                        Authentication Key Management : 802.1x-SHA256
                        Encrypted Traffic Analytics : No
                        Management Frame Protection : No
                        Protected Management Frame - 802.11w : Yes
                        EAP Type : LEAP
                        VLAN : 39
                        Multicast VLAN : 0
                        Access VLAN : 39
                        Anchor VLAN : 0
                        WFD capable : No
                        Manged WFD capable : No
                        . . . .
                        . . . .
```

129 Management Frame Protection

Topics:

- Information About Management Frame Protection

Information About Management Frame Protection

Management Frame Protection details.

By default, 802.11 management frames are unauthenticated and hence not protected against spoofing. Infrastructure management frame protection (MFP) and 802.11w protected management frames (PMF) provide protection against such attacks.

Infrastructure MFP

Infrastructure MFP protects management frames by detecting adversaries that are invoking denial-of-service attacks, flooding the network with associations and probes, interjecting as rogue APs, and affecting network performance by attacking the QoS and radio measurement frames. Infrastructure MFP is a global setting that provides a quick and effective means to detect and report phishing incidents.

Specifically, infrastructure MFP protects 802.11 session management functions by adding message integrity check information elements (MIC IEs) to the management frames emitted by APs (and not those emitted by clients), which are then validated by other APs in the network. Infrastructure MFP is passive, can detect and report intrusions but has no means to stop them.

Infrastructure MFP consists of three main components:

- **Management frame protection**: The AP protects the management frames it transmits by adding a MIC IE to each frame. Any attempt to copy, alter, or replay the frame invalidates the MIC, causing any receiving AP configured to detect MFP frames to report the discrepancy. MFP is supported for use with Cisco Aironet lightweight APs.
- **Management frame validation**: In infrastructure MFP, the AP validates every management frame that it receives from other APs in the network. It ensures that the MIC IE is present (when the originator is configured to transmit MFP frames) and matches the content of the management frame. If it receives any frame that does not contain a valid MIC IE from a BSSID belonging to an AP that is configured to transmit MFP frames, it reports the discrepancy to the network management system. In order for the timestamps to operate properly, all controllers must be Network Time Protocol (NTP) synchronized.
- **Event reporting**: The AP notifies the controller when it detects an anomaly, and the controller aggregates the received anomaly events and can report the results through SNMP traps to the network management system.

Infrastructure MFP is disabled by default, and you can enable it globally. When you upgrade from a previous software release, infrastructure MFP is disabled globally if you have enabled AP authentication because the two features are mutually exclusive. When you enable infrastructure MFP globally, signature generation (adding MICs to outbound frames) can be disabled for selected WLANs, and validation can be disabled for selected APs.

> **Note**
>
> CCXv5 client MFP is no longer supported. Client MFP is enabled as optional by default on WLANs that are configured for WPA2. However, client MFP is not supported on Wave 2 APs or 802.11ax Wi-Fi6 APs, and there exist no clients that support CCXv5.

Supported Access Point Models

Cisco MFP is supported on the following AP models:

- Cisco Aironet 2802, 3802, and 4802 series access points
- Cisco Aironet 2800, 3800, 4800, and 1560 series access points

Unsupported Access Point Models

Cisco MFP is not supported on the following AP models:

- Cisco Aironet 1800 series access points
- Cisco 802.11ax access points
- All Cisco IOS access points

Restrictions for Management Frame Protection

Management Frame Protection restrictions.

- Lightweight access points support infrastructure MFP in local and monitor modes and in FlexConnect mode when the access point is connected to a controller.
- Client MFP is supported for use only with CCXv5 clients using WPA2 with TKIP or AES-CCMP.
- Client MFP is not supported on Cisco Wave 1 APs and Cisco Wave 2 APs.
- 802.11ax access points do not support MFP.
- Non-CCXv5 clients may associate to a WLAN, if client MFP is disabled or optional.
- Error reports generated on a FlexConnect access point in standalone mode cannot be forwarded to the controller and are dropped.
- Keys are generated using random number generator but you can improve the keys by changing to SHA.
- MFP key for each BSSID is not supported.

Configure Management Frame Protection (CLI)

Configure management frame protection using command-line interface (CLI) commands.

Configure Management Frame Protection (MFP) on a device using the command-line interface (CLI).

Before you begin

Use these steps to configure management frame protection.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure management frame protection.

   Example

   ```
   Device(config)# wireless wps mfp
   ```

3. Configure AP impersonation detection or set the MFP key refresh interval in hours.

Example

```
Device(config)# wireless wps mfp ap-impersonation
```

```
Device(config)# wireless wps mfp key-refresh-interval
```

key-refresh-interval: Set sthe MFP key refresh interval in hours.

The valid range is one to 24 hours. The default value is 24 hours.

4. Save the configuration, exit configuration mode, and return to privileged **EXEC** mode.

Example

```
Device(config)# end
```

The device protects wireless management traffic according to the configured settings.

Verifying Management Frame Protection Settings

Verify Management Frame Protection Settings.

To verify if the Management Frame Protection (MFP) feature is enabled or not, use the following command:

```
Device# show wireless wps summary
Client Exclusion Policy
  Excessive 802.11-association failures   : unknown
  Excessive 802.11-authentication failures: unknown
  Excessive 802.1x-authentication         : unknown
  IP-theft                                : unknown
  Excessive Web authentication failure    : unknown
  Failed Qos Policy                       : unknown

Management Frame Protection
  Global Infrastructure MFP state : Enabled
  AP Impersonation detection      : Disabled
  Key refresh interval            : 15
```

To view the MFP details, use the following command:

```
Device# show wireless wps mfp summary
Management Frame Protection
  Global Infrastructure MFP state : Enabled
  AP Impersonation detection      : Disabled
  Key refresh interval            : 15
```

To view the MFP statistics details, use the following command:

```
Device# show wireless wps mfp statistics
BSSID          Radio DetectorAP                          LastSourceAddr Error
        Count     FrameTypes

aabb.ccdd.eeff a     AP3800                              aabb.ccdd.eeff Invalid
MIC     10        Beacon, Probe Response
                                                                        Invalid
MIC     20        Beacon, Probe Response
```

To verify if access points support MFP validation and protection, use the following command:

```
Device# show wireless wps mfp ap summary
AP Name                          Radio MAC               Validation
   Protection
-------------------------------------------------------------------------------
AP002A.1087.CBF4                 00a2.eefd.bdc0          Enabled
   Enabled
AP58AC.78DE.9946                 00a2.eeb8.4ae0          Enabled
   Enabled
APb4de.3196.caac                 4c77.6d83.6b90          Enabled
   Enabled
```

130 IPv4 ACLs

Topics:

- Network security with ACLs
- How to configure ACLs
- Configuration examples for ACLs
- Monitor IPv4 ACLs

Network security with ACLs

Explains how access control lists (ACLs) are used to secure network traffic on a switch.

This chapter describes how to configure network security on the switch by using access control lists (ACLs), which in commands and tables are also referred to as access lists.

ACL overview

This section explains ACL overview.

An access control list (ACL) is a network security mechanism that

- defines permit and deny conditions for network packets
- filters inbound and outbound traffic as it traverses network devices, and
- enforces access and segmentation policies by controlling which packets are forwarded or dropped.

When a packet is received on an interface, the switch compares the fields in the packet against any applied ACLs to verify that the packet has the required permissions to be forwarded, based on the criteria specified in the access lists. The controller checks each packet against the conditions you set in the access list, one at a time. The controller accepts or rejects the packets based on the first matching condition. Arrange your access list conditions carefully because the controller stops checking once it finds a match. If you do not set any restrictions, the controller forwards the packet. If you set restrictions and none are matched, the controller drops the packet. The controller can use ACLs on all packets it forwards . There is an implicit "deny any host" rule.

Configure access list

Configure access lists on the controller to provide basic security for your network. If you do not configure ACLs, the switch might allow all packets onto every part of your network.

Control access and traffic types

You can use ACLs to control which hosts access various parts of your network and to decide which types of traffic are forwarded or blocked at router interfaces. For example, you can allow email traffic but block Telnet traffic.

Access control entries

This section explains access control entry details.

An access control entry (ACE) is a rule component that

- specifies a *permit* or *deny* action for network packets
- defines conditions such as source or destination IP, protocol, or port that packets must meet to match the rule, and
- determines packet handling within an ordered ACL.

The meaning of *permit* or *deny* depends on the context in which the ACL is used.

> **Note**
>
> The maximum number of ACEs that can be applied under an access policy (ACL) for central switching is 256 ACEs. For Flex Mode Local Switching, the maximum is 64 ACEs.

ACL supported types

This section provides ACL supported type details.

The switch supports IP ACLs and Ethernet (MAC) ACLs:

- IP ACLs filter IPv4 traffic, including TCP, user datagram protocol (UDP), internet group management protocol (IGMP), and internet control message protocol (ICMP).
- Ethernet ACLs filter non-IP traffic.

This switch also supports quality of service (QoS) classification ACLs.

Supported ACLs

This section provides supported ACLs details.

The controller supports three types of ACLs to filter traffic:

- Port ACL: Port ACLs control access to traffic entering a Layer 2 interface. You can apply port ACLs to a Layer 2 interface in both directions for each access list type: IPv4 and MAC.
- Router ACL: Router ACLs control access to routed traffic between VLANs. They are applied to Layer 3 interfaces in a specific direction (inbound or outbound).
- FQDN ACL: FQDN ACLs are encoded along with IPv6 ACLs and sent to the AP. FQDN ACLs are always custom ACLs. The AP performs DNS snooping and sends the IPv4 and IPv6 addresses to the controller.

> **Note**
>
> On switches running the LAN base feature set, router ACLs are supported only on SVIs. VLAN maps are not supported.

ACL precedence

This section provides ACL precedence details.

When VLAN maps, port ACLs, and router ACLs are configured on the same switch, the filtering precedence for ingress traffic is greatest for port ACL, followed by VLAN map, and then router ACL. For egress traffic, router ACLs have the highest filtering precedence, followed by VLAN maps, and then port ACLs. These examples describe simple use cases:

- When both an input port ACL and a VLAN map are applied, incoming packets received on ports with a port ACL are filtered by the port ACL. Other packets are filtered by the VLAN map.
- When an input router ACL and an input port ACL exist in a switch virtual interface (SVI), incoming packets received on ports with a port ACL are filtered by the port ACL. Incoming routed IP packets are filtered by the router ACL. Other packets are not filtered.

- When an output router ACL and an input port ACL exist in an SVI, incoming packets received on ports with a port ACL are filtered by the port ACL. Outgoing routed IP packets are filtered by the router ACL. Other packets are not filtered.
- When a VLAN map, an input router ACL, and an input port ACL exist in an SVI, incoming packets received on ports with a port ACL are filtered only by the port ACL. Incoming routed IP packets received on other ports are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.
- When a VLAN map, an output router ACL, and an input port ACL exist in an SVI, incoming packets received on ports with a port ACL are filtered only by the port ACL. Outgoing routed IP packets are filtered by both the VLAN map and the router ACL. Other packets are filtered only by the VLAN map.

Port ACLs

This section explains port ACL details.

A port ACL is an access control mechanism that

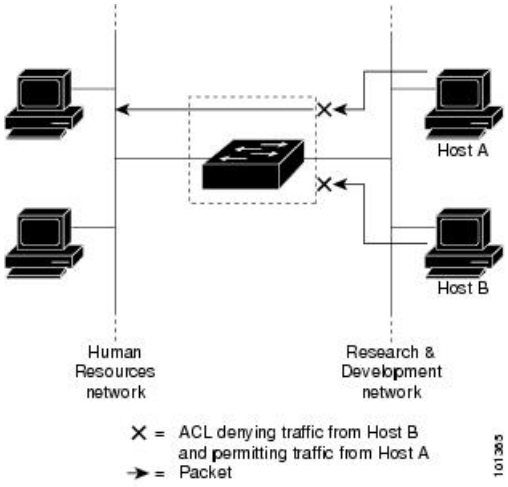
- applies to Layer 2 physical or EtherChannel interfaces on a switch
- supports the use of both IP and MAC-based access lists to filter traffic, and
- controls traffic in the inbound and outbound directions on the respective interface.

These access lists are supported:

- Standard IP access lists that use source addresses.
- Extended IP access lists that use source and destination addresses and may include protocol type information.
- MAC extended access list that use source and destination MAC addresses and may include protocol type information.

The switch examines ACLs on an interface and permits or denies packet forwarding based on how the packet matches the entries in the ACL. In this way, ACLs control access to a network or to part of a network.

Figure 20: Using ACLs to Control Traffic in a Network



This is an example of using port ACLs to control access to a network when all workstations are in the same VLAN. When you apply ACLs at the Layer 2 input, they allow Host A to access the Human Resources network while preventing Host B from accessing it. Port ACLs can only be applied to Layer 2 interfaces in the inbound direction.

When you apply a port ACL to a trunk port, the ACL filters traffic on all VLANs present on the trunk port. When you apply a port ACL to a port with a voice VLAN, it filters traffic on both data and voice VLANs.

With port ACLs, you can filter IP traffic by using IP access lists and non-IP traffic by using MAC addresses. You can filter both IP and non-IP traffic on the same Layer 2 interface by applying both an IP access list and a MAC access list to the interface.

Note

You can apply only one IP access list and one MAC access list to a Layer 2 interface. If an IP access list or MAC access list is already configured on a Layer 2 interface and you apply a new IP access list or MAC access list to the interface, the new ACL replaces the previously configured one.

Router ACLs

This section explains router ACL details.

A router ACL is a network security mechanism that

- can be applied to Layer 3 interfaces (such as SVIs, physical interfaces, and EtherChannels)
- filters traffic based on defined rules in specific directions (inbound or outbound), and
- supports both standard and extended IP matching criteria.

You can apply one router ACL in each direction on an interface.

The switch supports these access lists for IPv4 traffic:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses and optional protocol type information for matching operations.

As with port ACLs, the switch examines ACLs associated with features configured on each interface. When packets enter the switch on an interface, ACLs associated with all inbound features configured on that interface are examined. After routing, and prior to forwarding to the next hop, the switch examines all ACLs associated with outbound features configured on the egress interface.

ACLs permit or deny packet forwarding based on how the packet matches the entries in the ACL. They can be used to control access to a network or to part of a network.

ACEs and fragmented and unfragmented traffic

This section explains ACEs, fragmented and unfragmented traffic details.

A fragmented or unfragmented packet is a type of network traffic that

- may be divided into multiple fragments as it traverses a network
- only includes Layer 4 information in the first fragment (such as TCP/UDP port, ICMP type/code), and
- requires specific ACE matching behavior depending on the presence or absence of Layer 4 information in its fragments.

IP packets can be fragmented as they cross the network. The first fragment, which contains the beginning of the packet, includes Layer 4 information such as TCP or UDP port numbers, or ICMP type and code. All other fragments lack Layer 4 information.

Some access control entries (ACEs) do not check Layer 4 information, so they can be applied to all packet fragments. ACEs that test Layer 4 information cannot be applied to most fragments in a fragmented IP packet. If a fragment contains no Layer 4 information and the ACE tests Layer 4 information, matching rules are modified.

- Permit ACEs that check Layer 3 information in the fragment, including protocol type such as TCP or UDP, match the fragment regardless of missing Layer 4 details.

  Note

  For TCP ACEs with Layer 4 options, fragmented packets are dropped in accordance with RFC 1858.

- Deny ACEs that check Layer 4 information match a fragment only if the fragment contains Layer 4 information.

ACEs and fragmented and unfragmented traffic examples

This topic provides ACEs, fragmented and unfragmented traffic examples.

Consider access list 102, configured with these commands, applied to three fragmented packets:

```
Device(config)# access-list 102 permit tcp any host 10.1.1.1 eq smtp
Device(config)# access-list 102 deny tcp any host 10.1.1.2 eq telnet
Device(config)# access-list 102 permit tcp any host 10.1.1.2
Device(config)# access-list 102 deny tcp any any
```

Note

In the first and second ACEs, the *eq* keyword after the destination address checks if the TCP destination port matches the well-known port numbers for Simple Mail Transfer Protocol (SMTP) or Telnet.

- Packet A is a TCP packet from host 10.2.2.2, port 65000, sent to host 10.1.1.1 on the SMTP port. If the packet is fragmented, the first fragment matches the first ACE (a permit) because all Layer 4 information is included. The remaining fragments also match the first ACE, even if they do not include the SMTP port information, because the first ACE checks only Layer 3 information when applied to fragments. In this example, the fragment is a TCP packet sent to 10.1.1.1.
- Packet B is sent from host 10.2.2.2, port 65001, to host 10.1.1.2 on the Telnet port. If the packet is fragmented, the first fragment matches the second ACE (a deny) because it includes all Layer 3 and Layer 4 information. The remaining fragments in the packet do not match the second ACE because they are missing Layer 4 information. Instead, they match the third ACE (a permit).

  Because the first fragment was denied, host 10.1.1.2 cannot reassemble a complete packet. As a result, packet B cannot be reassembled on host 10.1.1.2. Permitted fragments still use network bandwidth and resources on host 10.1.1.2 during the attempted reassembly.
- Packet C is sent from host 10.2.2.2, port 65001, to host 10.1.1.3 on the FTP port. This packet is fragmented. If this packet is fragmented, the first fragment matches the fourth ACE (a deny). All other fragments also match the fourth ACE because it does not check Layer 4 information. Layer 3 information in each fragment shows they are sent to host 10.1.1.3, while previous permit ACEs check different hosts.

Standard and extended IPv4 ACLs

This section describes standard and extended IPv4 ACLs details.

An IPv4 access control list (ACL) is a packet-filtering feature that

- sequentially compares packets against a set of permit or deny conditions
- determines packet acceptance or rejection based on the first match, and
- enforces traffic control policies on network interfaces.

The switch stops testing after the first match. The order of the conditions is critical. If no conditions match, the switch denies the packet.

The software supports two types of ACLs for IPv4:

- Standard IP access lists use source addresses for matching operations.
- Extended IP access lists use source and destination addresses for matching operations and can include protocol-type information for more granular control.

Note

Only extended ACLs are supported; standard ACLs are not supported.

IPv4 ACL switch unsupported features

This section provides IPv4 ACL switch unsupported feature details.

Configuring IPv4 ACLs on this switch is similar to configuring IPv4 ACLs on other Cisco switches and routers.

The switch does not support the following ACL-related features:

- Non-IP protocol ACLs or bridge-group ACLs.
- IP accounting.
- Inbound and outbound rate limiting (except with QoS ACLs).
- Reflexive ACLs, URL redirect ACLs and dynamic ACLs are not supported. (except for some specialized dynamic ACLs used by the switch clustering feature).
- ACL logging for port ACLs and VLAN maps.

Access list numbers

This section provides access list number details.

The ACL number determines the type of access list you create.

Table 118: Access List Numbers

This lists the access-list number and corresponding access list type and shows whether or not they are supported in the switch. The switch supports IPv4 standard and extended access lists, numbers 1 to 199 and 1300 to 2699.

Access List Number	Type	Supported
1–99	IP standard access list	Yes
100–199	IP extended access list	Yes
200–299	Protocol type-code access list	No
300–399	DECnet access list	No

Access List Number	Type	Supported
400–499	XNS standard access list	No
500–599	XNS extended access list	No
600–699	AppleTalk access list	No
700–799	48-bit MAC address access list	No
800–899	IPX standard access list	No
900–999	IPX extended access list	No
1000–1099	IPX SAP access list	No
1100–1199	Extended 48-bit MAC address access list	No
1200–1299	IPX summary address access list	No
1300–1999	IP standard access list (expanded range)	Yes
2000–2699	IP extended access list (expanded range)	Yes

In addition to numbered standard and extended ACLs, you can create standard and extended named IP ACLs with the supported numbers. For example, you can name standard IP ACLs with numbers from 1 to 99, and extended IP ACLs with numbers from 100 to 199. With named ACLs, you can delete individual entries, but with numbered lists, you cannot.

Numbered standard IPv4 ACLs

This section provides numbered standard IPv4 ACL details.

When you create an ACL, an implicit deny statement is automatically present at the end. Any packet that does not match an earlier entry in the ACL is denied by this statement. For standard access lists, if you omit the mask in an IP host address ACL entry, the switch assumes a mask of 0.0.0.0.

- The switch rewrites standard access lists to move host match entries and entries with a don't care mask of 0.0.0.0 to the top of the list. Entries with non-zero don't care masks are placed below them. As a result, the ACEs may appear in a different order in show command output and in the configuration file than the order in which you entered them.
- After you create a numbered standard IPv4 ACL, you can apply it to VLANs, feature-id terminal lines (virtual teletype (VTY) lines), or interfaces.

Numbered extended IPv4 ACLs

This section explains numbered extended IPv4 ACLs.

A numbered extended IPv4 ACL is an access control mechanism that

- matches IPv4 traffic using both source and destination addresses
- optionally uses protocol type information for greater refinement, and
- requires that Access Control Entries (ACEs) be appended to the list since reordering or selective removal is unsupported.

When you are creating ACEs in numbered extended access lists, remember that after you create the ACL, any additions are placed at the end of the list.

The switch does not support dynamic or reflexive access lists. It also does not support filtering based on the type of service (ToS) minimize-monetary-cost bit.

Some protocols also have specific parameters and keywords that apply to that protocol.

You can define an extended TCP, UDP, ICMP, IGMP, or other IP ACL. The switch also supports these IP protocols:

- Authentication Header Protocol (**ahp**)
- Encapsulation Security Payload (**esp**)
- Enhanced Interior Gateway Routing Protocol (**eigrp**)
- generic routing encapsulation (**gre**)
- Internet Control Message Protocol (**icmp**) and Open Shortest Path First routing (**ospf**)
- Internet Group Management Protocol (**igmp**) and Payload Compression Protocol (**pcp**)
- any Interior Protocol (**ip**) and Protocol-Independent Multicast (**pim**)
- IP in IP tunneling (**ipinip**) and Transmission Control Protocol (**tcp**)
- KA9Q NOS-compatible IP over IP tunneling (**nos**) and User Datagram Protocol (**udp**)

Named IPv4 ACLs

This section provides named IPv4 ACL details.

You can identify IPv4 ACLs with an alphanumeric string (a name) rather than a number. You can use named ACLs to configure more IPv4 access lists in a router than if you were to use numbered access lists. If you identify your access list with a name rather than a number, the mode and command syntax are slightly different. However, at times, not all commands that use IP access lists accept a named access list.

Consider these guidelines before configuring named ACLs:

- Numbered ACLs are also available.
- A standard ACL and an extended ACL cannot have the same name.
- VLAN ACLs or VLAN maps are not supported.
- You can use standard or extended ACLs whether named or numbered, in VLAN maps.
- With IPv4 QoS ACLs, if you enter the class-map { match-all | match-any} class-map-name global configuration command, you can enter these match commands:

  1. **match access-group** *acl-name*

     > **Note**
     >
     > The ACL must be an extended named ACL.

  2. **match input-interface** *interface-id-list*
  3. **match ip dscp** *dscp-list*
  4. **match ip precedence** *ip-precedence-list*

Note

You cannot enter the **match access-group** *acl-index* command.

ACL logging

This topic explains ACL logging details.

ACL logging is a router logging feature that

- generates informational messages about packets permitted or denied by a standard IP access control list (ACL)
- records each message with details such as the action taken, the source IP address, and the access list number, and
- supports logging only for router ACLs (RACLs), not for ACLs used with Unicast Reverse Path Forwarding (uRPF).

The controller software can provide logging messages about packets permitted or denied by a standard IP access list. The level of messages logged to the console is controlled by the **logging console** commands, which control the syslog messages.

Note

Because routing is done in hardware and logging is done in software, if a large number of packets match a *permit* or *deny* ACE containing a **log** keyword, the software might not be able to match the hardware processing rate, and not all packets will be logged.

The first packet that triggers the ACL generates a logging message immediately. Subsequent packets are collected over 5-minute intervals before they are logged. The logging message includes the access list number, an indication of whether the packet was permitted or denied, the source IP address, and the number of packets from that source permitted or denied during the prior 5-minute interval.

Note

The logging facility might drop some message packets if there are too many to handle, or if more than one logging message must be processed in a second. This behavior prevents the router from crashing due to too many logging packets. Therefore, do not use the logging facility as a billing tool or as an accurate source for the number of matches to an access list.

Hardware and software treatment of IP ACLs

This section provides hardware and software treatment details of IP ACLs.

ACL processing is performed in hardware. If the hardware reaches its capacity to store ACL configurations, all packets on that interface are dropped.

The ACL scale for controllers is as follows:

- Cisco Catalyst 9800-40 wireless controller, Cisco Catalyst 9800-L wireless controller, Cisco Catalyst 9800-CL wireless controller (small and medium) support 128 ACLs with 128 access list entries (ACEs).
- Cisco Catalyst 9800-80 wireless controller and Cisco Catalyst 9800-CL wireless controller (large) support 256 ACLs and 256 ACEs.
- FlexConnect and Fabric mode APs support 96 ACLs.

Note

If an ACL configuration cannot be implemented in the hardware due to an out-of-resource condition on the controller, then only the traffic in that VLAN arriving on that controller is affected. Software forwarding of packets might adversely impact the performance of the switch or switch stack, depending on the number of CPU cycles that this consumes.

For router ACLs, other factors can cause packets to be sent to the CPU:

- Using the log keyword
- Generating ICMP unreachable messages

When traffic flows are both logged and forwarded, forwarding is done by hardware, but logging must be done by software. Because of the difference in packet handling capacity between hardware and software, if the sum of all flows being logged (both permitted flows and denied flows) is of great enough bandwidth, not all of the packets that are forwarded can be logged.

If a router ACL configuration cannot be applied in hardware, packets that arrive in a VLAN and must be routed are processed in software. However, packets are bridged in hardware. If ACLs cause large numbers of packets to be sent to the CPU, the switch performance can be negatively affected.

When you enter the show ip access-lists privileged EXEC command, the match count displayed does not account for packets that are access controlled in hardware. Use the show access-lists hardware counters and show platform acl counters hardware privileged EXEC commands to obtain basic hardware ACL statistics for switched and routed packets.

Router ACLs operate in this way:

- The hardware controls permit and deny actions of standard and extended ACLs (input and output) for security access control.
- If log has not been specified, and if ip unreachables is disabled, the hardware drops flows that match a deny statement in a security ACL. The flows matching a permit statement are switched in hardware.
- Adding the log keyword to an ACE in a router ACL causes a copy of the packet to be sent to the CPU for logging only. If the ACE is a permit statement, the packet is switched and routed in hardware.

IPv4 ACL interface considerations

This section provides IPv4 ACL interface consideration details.

When you apply the ip access-group interface configuration command to a Layer 3 interface (an SVI, a Layer 3 EtherChannel, or a routed port), the interface must have been configured with an IP address. Layer 3 access groups filter packets that are routed and packets that are received by Layer 3 processes on the CPU. They do not affect packets bridged within a VLAN.

- For inbound ACLs, after receiving a packet, the controller checks the packet against the ACL. If the ACL permits the packet, the controller continues to process the packet. If the ACL rejects the packet, the controller discards the packet.

- For outbound ACLs, after receiving and routing a packet to a controlled interface, the controller checks the packet against the ACL. If the ACL permits the packet, the controller sends the packet. If the ACL rejects the packet, the controller discards the packet.

If an undefined ACL has nothing listed in it, it is an empty access list.

By default, the input interface sends ICMP Unreachable messages whenever a packet is discarded, regardless of whether the packet was discarded because of an ACL on the input interface or because of an ACL on the output interface. ICMP Unreachables are normally limited to no more than one every half second per input interface. You can change this by using the ip icmp rate-limit unreachable global configuration command.

Restrictions for configuring IPv4 access control lists

This topic describes the restrictions for configuring IPv4 access control list restrictions.

General Network Security

These are restrictions for configuring network security with ACLs:

- A standard ACL and an extended ACL cannot have the same name.
- **AppleTalk** appears in the command-line help strings but is not supported as a matching condition for the **deny** and **permit** MAC access-list configuration mode commands.
- DNS traffic is permitted by default for clients that are awaiting web authentication, regardless of ACL entries.

IPv4 ACL Network Interfaces

These restrictions apply to IPv4 ACLs to network interfaces:

- When controlling access to an interface, you can use a named or numbered ACL.
- You do not have to enable routing to apply ACLs to Layer 2 interfaces.

MAC ACLs on a Layer 2 Interface

After you create a MAC ACL, you can apply it to a Layer 2 interface to filter non-IP traffic coming in that interface. When you apply the MAC ACL, consider these guidelines:

- You can apply no more than one IP access list and one MAC access list to the same Layer 2 interface. The IP access list filters only IP packets, and the MAC access list filters non-IP packets.
- A Layer 2 interface can have only one MAC access list. If you apply a MAC access list to a Layer 2 interface that has a MAC ACL configured, the new ACL replaces the previously configured one.

> **Note**
>
> The **mac access-group** interface configuration command is only valid when applied to a physical Layer 2 interface. You cannot use the command on EtherChannel port channels.

IP Access List Entry Sequence Numbering

- This feature does not support dynamic, reflexive, or firewall access lists.

How to configure ACLs

Configure ACLs.

Configure IPv4 ACLs (GUI)

Use this procedure to configure IPv4 ACLs using the GUI.

Define and apply IPv4 ACL rules for traffic management and security.

Procedure

1. Choose **Configuration** > **Security** > **ACL**.
2. Click **Add**.
3. In the **Add ACL Setup** dialog box, enter the details for these parameters.
   - ACL Name: Enter the name for the ACL.
   - ACL Type: IPv4 Standard.
   - Sequence: Enter the sequence number.
   - Action: Select ***Permit*** or ***Deny*** the packet flow from the drop-down list.
   - Source Type: Choose ***any***, ***Host*** or ***Network*** from which the packet is sent.
   - Log: Enable or disable logging.
4. Click **Add**.
5. Add the rest of the rules and click **Apply to Device**.

Configure IPv4 ACLs

Use this procedure to configure IPv4 ACLs.

Follow the procedure given below to use IP ACLs on the switch:

Procedure

1. Create an ACL by specifying an access list number or name and the access conditions.
2. Apply the ACL to interfaces or terminal lines. .

Create a numbered standard ACL (GUI)

Use this procedure to create a numbered standard ACL using the GUI.

Define and apply a numbered standard ACL using the device's web interface using GUI.

Procedure

1. Choose **Configuration** > **Security** > **ACL**.

2. On the **ACL** page, click **Add**.
3. In the **Add ACL Setup** window, enter the details for these parameters.
   - ACL Name: Enter the name for the ACL.
   - ACL Type: IPv4 Standard.
   - Sequence: Enter the sequence number.
   - Action: Select **Permit** or **Deny** access from the drop-down list.
   - Source Type: Choose **any** , **Host** or **Network**.
   - Log: Enable or disable logging, this is limited to ACLs associated to Layer 3 interface only.
4. Click **Add**.
5. Click **Save & Apply to Device**.

Create a numbered standard ACL (CLI)

Use this procedure to create a numbered standard ACL using commands.

Follow the procedure given below to create a numbered standard ACL using commands:

Procedure

1. Enable the privileged EXEC mode and enter your password if prompted.

   Example

   ```
   Device# enable
   ```

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Define a standard IPv4 access list by using a source address and wildcard.

   Example

   ```
   Device(config)# access-list 2 {deny | permit} source source-wildcard your_host
   ```

   The *access-list-number* is a decimal number from 1 to 99 or 1300 to 1999.

   Enter **deny** or **permit** to specify whether to deny or permit access if conditions are matched.

   The *source* is the source address of the network or host from which the packet is being sent. Specify the source as:

   - The 32-bit quantity in dotted-decimal format.
   - The keyword **any** abbreviates for *source* and *source-wildcard* as 0.0.0.0 255.255.255.255. You do not need to enter a source-wildcard.
   - The keyword **host** abbreviates source and *source-wildcard* as *source* 0.0.0.0.

The *source-wildcard* applies wildcard bits to the source address.

> **Note**
>
> Logging is supported only on ACLs attached to Layer 3 interfaces.

4. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

5. Verify your entries.

Example

```
Device# show running-config
```

6. Optional: Save your entries in the configuration file.

Example

```
Device# copy running-config startup-config
```

Create a numbered extended ACL (GUI)

Use this procedure to create a numbered extended ACL using the GUI.

Set up a numbered extended ACL to control network traffic based on source, destination, and protocol using the GUI.

Procedure

1. Choose **Configuration** > **Security** > **ACL**.
2. On the **ACL** page, click **Add**.
3. In the **Add ACL Setup** window, enter the details for these parameters.
   - ACL Name: Enter the name for the ACL.
   - ACL Type: IPv4 Extended.
   - Sequence: Enter the sequence number.
   - Action: Select **Permit** or **Deny** the packet flow from the drop-down list.
   - Source Type: Choose **any** , **Host** or **Network** from which the packet is sent.
   - Destination Type: Choose **any** , **Host** or **Network** to which the packet is sent.
   - Protocol: Select a protocol from the drop-down list.
   - Log: Enable or disable logging.

- DSCP: Enter to match packets with the DSCP value.

4. Click **Add**.
5. Click **Save & Apply to Device**.

Create a numbered extended ACL (CLI)

Use this procedure to create a numbered extended ACL using commands.

Follow the procedure given below to create a numbered extended ACL using commands:

Procedure

1. Enters global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Defines an extended IPv4 access list and the access conditions.

   Example

   ```
   Device# access-list 101 {deny | permit} ip host 10.1.1.2 any [precedence 0]
   [tos 0] [fragments] log [log-input | smartlog] [time-range time-range-name]
   [dscp dscp]
   ```

   The *access-list-number* is a decimal number from 100 to 199 or 2000 to 2699.

   Enter **deny** or **permit** to specify whether to deny or permit the packet if conditions are matched.

   For *protocol* , enter the name or number of an P protocol: **ahp**, **eigrp**, **esp**, **gre**, **icmp**, **igmp**, **igrp**, **ip**, **ipinip**, **nos**, **ospf**, **pcp**, **pim**, **tcp**, or **udp**, or an integer in the range 0 to 255 representing an IP protocol number. To match any Internet protocol (including ICMP, TCP, and UDP), use the keyword **ip**.

   > **Note**
   >
   > This step includes options for most IP protocols. For additional specific parameters for TCP, UDP, ICMP, and IGMP, see the following steps.

   The *source* is the number of the network or host from which the packet is sent.

   The *source-wildcard* applies wildcard bits to the source.

   The *destination* is the network or host number to which the packet is sent.

   The *destination-wildcard* applies wildcard bits to the destination.

   Source, source-wildcard, destination, and destination-wildcard can be specified as:

   - The 32-bit quantity in dotted-decimal format.
   - The keyword **any** for 0.0.0.0 255.255.255.255 (any host).
   - The keyword **host** for a single host 0.0.0.0.

The other keywords are optional and have these meanings:

- **precedence**: Enter to match packets with a precedence level specified as a number from 0 to 7 or by name: ***routine*** (0), ***priority*** (1), ***immediate*** (2), ***flash*** (3), ***flash-override*** (4), ***critical*** (5), ***internet*** (6), ***network*** (7).
- **fragments**: Enter to check non-initial fragments.
- **tos**: Enter to match by type of service level, specified by a number from 0 to 15 or a name: ***normal*** (0), ***max-reliability*** (2), ***max-throughput*** (4), ***min-delay*** (8).
- **time-range**: Specify the time-range name.
- **dscp**: Enter to match packets with the DSCP value specified by a number from 0 to 63, or use the question mark (?) to see a list of available values.

> **Note**
>
> Your controller must support the ability to:
>
> - Mark DCSP
> - Mark UP
> - Map DSCP and UP
>
> For more information on DSCP-to-UP Mapping , see:
>
> https://tools.ietf.org/html/draft-ietf-tsvwg-ieee-802-11-01

> **Note**
>
> If you enter a **dscp** value, you cannot enter **tos** or **precedence**. You can enter both a **tos** and a **precedence** value with no **dscp**.

3. Define an extended TCP access list and the access conditions.

Example

```
Device(config)# access-list 101 {deny | permit} tcp source source-wildcard
[operator-port] destination destination-wildcard 500 [established] [precedence
 0] [tos 0] [fragments] log [log-input | smartlog] [time-range time-range-name]
 [dscp dscp] [flag]
```

The parameters are the same as those described for an extended IPv4 ACL, with these exceptions:

Enter an *operator* and *port* to compare source (if positioned after *source source-wildcard*) or destination (if positioned after *destination destination-wildcard*) port. Possible operators include **eq** (equal), **gt** (greater than), **lt** (less than), **neq** (not equal), and **range** (inclusive range). Operators require a port number (range requires two port numbers separated by a space).

Enter the *port* number as a decimal number (from 0 to 65535) or the name of a TCP port. Use only TCP port numbers or names when filtering TCP.

The other optional keywords have these meanings:

- *flag*: Enter one of these flags to match by the specified TCP header bits: **ack** (acknowledge), **fin** (finish), **psh** (push), **rst** (reset), **syn** (synchronize), or **urg** (urgent).

4. Define an extended UDP access list and the access conditions.

Example

```
Device(config)# access-list 101 {deny | permit} udp source source-wildcard
[operator port] destination destination-wildcard 100 precedence 0] [tos 0]
[fragments] log [log-input | smartlog] [time-range time-range-name] [dscp
dscp
```

The UDP parameters are the same as those described for TCP except that the [operator [port]] port number or name must be a UDP port number or name, and the **flag** not valid for UDP.

5. Define an extended ICMP access list and the access conditions.

Example

```
Device(config)# access-list 101 {deny | permit} icmp source source-wildcard
destination destination-wildcard [ icmp-type | [[ icmp-type icmp-code ] | [
icmp-message ]] precedence 0] [tos 0] [fragments] log [log-input | smartlog]
 [time-range time-range-name] [dscp dscp]
```

The ICMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with the addition of the ICMP message type and code parameters. These optional keywords have these meanings:

- *icmp-type*: Enter to filter by ICMP message type, a number from 0 to 255.
- *icmp-code*: Enter to filter ICMP packets that are filtered by the ICMP message code type, a number from 0 to 255.
- *icmp-message*: Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name.

6. Optional: Define an extended IGMP access list and the access conditions.

Example

```
Device(config)# access-list 101 {deny | permit} igmp source source-wildcard
destination destination-wildcard [ igmp-type ] precedence 0] [tos 0]
[fragments] log [log-input | smartlog] [time-range time-range-name] [dscp
dscp]
```

The IGMP parameters are the same as those described for most IP protocols in an extended IPv4 ACL, with this optional parameter.

igmp-type: To match IGMP message type, enter a number from 0 to 15, or enter the message name: **dvmrp**, **host-query**, **host-report**, **pim**, or **trace**.

7. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

8. Optional: Save your entries in the configuration file.

Example

```
Device# copy running-config startup-config
```

Create named standard ACLs (GUI)

Use this procedure to create named standard ACLs using the GUI.

Use this procedure to set up and apply ACLs for network security management within the device's GUI.

Procedure

1. Click **Configuration** > **Security** > **ACL**.
2. Click **Add** to create a new ACL setup.
3. In the **Add ACL Setup** window, enter these parameters:
   - ACL Name: Enter the name for the ACL.
   - ACL Type: IPv4 Standard.
   - Sequence: The valid range is between 1 and 99 or 1300 and 1999.
   - Action: Select **Permit** or **Deny** access from the drop-down list.
   - Source Type: Choose **any**, **Host** or **Network**.
   - Log: Enable or disable logging, this is limited to ACLs associated to Layer 3 interface only.
4. Click **Add** to add the rule.
5. Click **Save & Apply to Device**.

Create named standard ACLs (CLI)

Use this procedure to create named standard ACLs using commands.

Define and apply named standard ACLs to permit or deny traffic from specified IPv4 source addresses using commands.

Procedure

1. Enter the privileged EXEC mode.

Example

```
Device# enable
```

2. Enter the global configuration mode.

Example

```
Device# configure terminal
```

3. Define a standard IPv4 access list using a name, and enter the access-list configuration mode.

   Example

   ```
   Device(config)# ip access-list standard 20
   ```

   The name can be a number from 1 to 99.

4. Specify one or more conditions denied or permitted to decide if the packet is forwarded or dropped in access-list configuration mode. Use one of the these:

   - **deny** { *source* [*source-wildcard*] | **host** *source* | **any** } [**log**]
   - **permit** { *source* [*source-wildcard*] | **host** *source* | **any** } [**log**]

   Example

   ```
   Device(config-std-nacl)# deny 192.168.0.0 0.0.255.255 255.255.0.0 0.0.255.255
   ```

   or

   ```
   Device(config-std-nacl)# permit 10.108.0.0 0.0.0.0 255.255.255.0 0.0.0.0
   ```

   - **host** *source*: A source and source wildcard of *source* 0.0.0.0.
   - **any**: A source and source wildcard of 0.0.0.0 255.255.255.255.

5. Returns to privileged EXEC mode.

   Example

   ```
   Device(config-std-nacl)# end
   ```

6. Verify your entries.

   Example

   ```
   Device# show running-config
   ```

7. Optional: Save your entries in the configuration file.

   Example

   ```
   Device# copy running-config startup-config
   ```

Create extended named ACLs (GUI)

Use this procedure to create extended named ACLs using the GUI.

Follow these steps to filter traffic based on granular criteria, such as source/destination, protocol, or DSCP using the GUI.

Procedure

1. Choose **Configuration** > **Security** > **ACL**.
2. Click **Add**.
3. In the **Add ACL Setup** window, enter these parameters.
   - ACL Name: Enter the name for the ACL.
   - ACL Type: IPv4 Extended.
   - Sequence: Enter the sequence number.
   - Action: Select *Permit* or *Deny* the packet flow from the drop-down list.
   - Source Type: Choose *any*, *Host* or *Network* from which the packet is sent.
   - Destination Type: Choose *any*, *Host* or *Network* to which the packet is sent.
   - Protocol: Select a protocol from the drop-down list.
   - Log: Enable or disable logging.
   - DSCP: Enter to match packets with the DSCP value.
4. Click **Add**.
5. Add the rest of the rules and click **Apply to Device**.

Create extended named ACLs (CLI)

Use this procedure to create extended named ACLs using commands.

Use this procedure to create extended named access control lists (ACLs) using CLI commands.

Procedure

1. Enable the privileged EXEC mode and enter your password if prompted.

   Example

   ```
   Device# enable
   ```

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Define an extended IPv4 access list using a name, and enter the access-list configuration mode.

   Example

   ```
   Device(config)# ip access-list extended 150
   ```

   The name can be a number from 100 to 199.

4. Specify the conditions allowed or denied in access-list configuration mode { **deny** | **permit** } *protocol* { *source* [*source-wildcard*] | **host** *source* | **any** } { *destination* [*destination-wildcard*] | host *destination* | **any** } [**precedence** *precedence*] [**tos***tos*] [**log**] [**time-range** *time-range-name*]

   Example

   ```
   Device(config-ext-nacl)# permit 0 any any
   ```

   Use the **log** keyword to get access list logging messages, including violations.

   - **host** *source*: A source and source wildcard of *source* 0.0.0.0.
   - **host** *destintation*: A destination and destination wildcard of *destination* 0.0.0.0.
   - **any**: A source and source wildcard or destination and destination wildcard of 0.0.0.0 255.255.255.255.

5. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-ext-nacl)# end
   ```

6. Verifiy your entries.

   Example

   ```
   Device# show running-config
   ```

7. Optional: Save your entries in the configuration file.

   Example

   ```
   Device# copy running-config startup-config
   ```

When you create extended ACLs, an implicit deny statement is added by default at the end of the ACL. If no match occurs before reaching the end, all traffic is denied. For standard ACLs, omitting the mask from an IP host address access list specification causes the system to use 0.0.0.0 as the mask.

New entries are always placed at the end of the ACL after you create it. You cannot add ACL entries at specific positions within an ACL. However, you can remove entries from a named ACL using the **no permit** and **no deny** access-list configuration mode commands.

Being able to selectively remove lines from a named ACL is one reason you might use named ACLs instead of numbered ACLs.

What to do next

After you create a named ACL, apply it to interfaces or to VLANs .

Apply an IPv4 ACL to an interface (GUI)

Use this procedure to apply an IPv4 ACL to an interface using GUI.

Use this procedure when you need to restrict or permit traffic through a device interface based on IPv4 ACL rules using the GUI.

Procedure

1. Choose **Configuration** > **Security** > **ACL**.
2. Click **Associating Interfaces**.
3. Choose the interface from the **Available Interfaces** list to view its ACL details on the right-hand side. You can change the ACL details, if required.
4. Click **Save & Apply to Device**.

Apply an IPv4 ACL to an interface (CLI)

Use to procedure to apply an IPv4 ACL to an interface using commands.

Restrict or permit packets on an interface by associating an ACL using commands.

This section describes how to apply IPv4 ACLs to network interfaces.

Beginning in privileged EXEC mode, follow the procedure given below to control access to an interface:

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Identify a specific interface for configuration, and enter the interface configuration mode.

   Example

   ```
   Device(config)#
   ```

   The interface is a layer 2 interface (port ACL) , or a layer 3 interface (router ACL).

3. Control access to the specified interface.

   Example

   ```
   Device(config-if)# ip access-group {access-list-number 2 | name} {in | out}
   ```

4. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-if)# end
   ```

5. Display the access list configuration.

   Example

   ```
   Device# show running-config
   ```

6. Optional: Save your entries in the configuration file.

Example

```
Device# copy running-config startup-config
```

Apply ACL to policy profile (GUI)

Use this procedure to apply ACL to policy profile using the GUI.

Use this procedure to apply an ACL to a policy profile using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. On the **Policy Profile** page, click **Add**.
3. In the **Add Policy Profile** window, click **Access Policies** tab.
4. In the **WLAN ACL** area, select the IPv4 ACL from the **IPv4 ACL** drop-down list.
5. Click **Apply to Device**.

Apply ACL to policy profile (CLI)

Use this procedure to apply ACL to policy profile using commands.

Use these steps to attach a preconfigured IPv4 ACL to a specific WLAN policy profile using commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a WLAN policy profile and enter the wireless policy configuration mode.

Example

```
Device(config)# wireless profile policy profile-policy
```

3. Configure an IPv4 ACL.

Example

```
Device(config-wireless-policy)# ipv4 acl test-acl
```

4. Return to the privileged EXEC mode.

Example

```
Device(config-wireless-policy)# end
```

Alternatively, you can also press *Ctrl-Z* to exit global configuration mode.

Configuration examples for ACLs

Demonstrates practical ACL implementations through configuration examples, including commented entries, applying ACLs to policy profiles, small network scenarios, numbered, extended, and named ACLs, time-based rules, and logging configurations.

Examples: including comments in ACLs

This section provides details about including comments in ACLs examples.

You can use the **remark** keyword to include comments (remarks) about entries in any IP standard or extended ACL. The remarks make the ACL easier for you to understand and scan. Each remark line is limited to 100 characters.

The remark can go before or after a permit or deny statement. Place remarks consistently to clarify which permit or deny statement each remark describes. If remarks are placed both before and after permit or deny statements, it may be unclear which remark applies to which statement.

To include a comment for IP numbered standard or extended ACLs, use the **access-list** *access-list number* **remark** *remark* global configuration command. To remove the remark, use the **no** form of this command.

In this example, the workstation that belongs to Jones is allowed access, and the workstation that belongs to Smith is not allowed access:

```
Device(config)# access-list 1 remark Permit only Jones workstation through
Device(config)# access-list 1 permit 171.69.2.88
Device(config)# access-list 1 remark Do not allow Smith through
Device(config)# access-list 1 deny 171.69.3.13
```

For an entry in a named IP ACL, use the **remark** access-list configuration command. To remove the remark, use the **no** form of this command.

In this example, the Jones subnet is not allowed to use outbound Telnet:

```
Device(config)# ip access-list extended telnetting
Device(config-ext-nacl)# remark Do not allow Jones subnet to telnet out
Device(config-ext-nacl)# deny tcp host 171.69.2.88 any eq telnet
```

Examples: apply an IPv4 ACL to a policy profile in a wireless environment

This section provides details about applying an IPv4 ACL to a policy profile in a wireless environment examples.

This example shows how to apply an IPv4 ACL to a policy profile in a wireless environment.

> **Note**
>
> All IPv4 ACLs must be associated to a policy profile.

This example uses extended ACLs to permit TCP traffic.

1. Creating an IPv4 ACL.

```
Device(config)# ip access-list extended <acl-name>
Device(config-ext-nacl)# 10 permit ip any 10.193.48.224 0.0.0.31
Device(config-ext-nacl)# 20 permit ip any any
```

2. Applying the IPv4 ACL to a policy profile.

```
Device(config)# wireless profile policy <policy-profile-name>
Device(config-wireless-policy)# shutdown
Device(config-wireless-policy)# ipv4 acl <acl-name>
Device(config-wireless-policy)# no shutdown
```

IPv4 ACL Configuration Examples

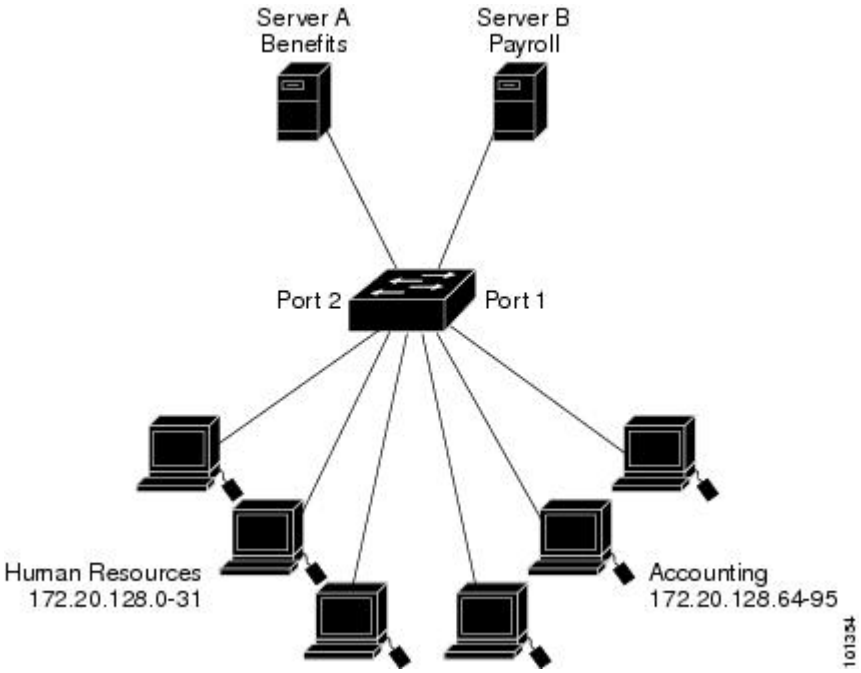
IPv4 ACL Configuration examples.

This section provides examples of configuring and applying IPv4 ACLs. For detailed information about compiling ACLs, see the *Cisco IOS Security Configuration Guide, Release 12.4* and to the Configuring IP Services" section in the "IP Addressing and Services" chapter of the *Cisco IOS IP Configuration Guide, Release 12.4*.

ACLs in a small networked office

This section provides ACLs in a small networked office details.

Figure 21: Using Router ACLs to Control Traffic



The diagram illustrates a small networked office. Routed Port 2 connects to Server A, which hosts benefits and other information accessible to all employees. Routed Port 1 connects to Server B, which contains confidential payroll data. All users can access Server A. Server B is restricted to authorized users.

Use router ACLs to do this in one of two ways:

- Create a standard ACL to filter traffic coming to the server from Port 1.
- Create an extended ACL to filter traffic coming from the server into Port 1.

Examples: ACLs in a small networked office

This section provides details about ACLs in a small networked office examples.

This example uses a standard ACL to filter traffic coming into Server B from a port. The ACL permits traffic only from accounting's source addresses 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic that comes out of routed Port 1 from the specified source address.

```
Device(config)# access-list 6 permit 172.20.128.64 0.0.0.31
Device(config)# end
Device# how access-lists

Standard IP access list 6
10 permit 172.20.128.64, wildcard bits 0.0.0.31
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 6 out
```

This example uses an extended ACL to filter traffic from Server B into a port. It permits traffic from any source address (in this case Server B) only to accounting destination addresses in the range 172.20.128.64 to 172.20.128.95. The ACL is applied to traffic going into routed Port 1, permitting it to go only to the specified destination addresses. With extended ACLs, enter the protocol (IP) before specifying source and destination information.

```
Device(config)# access-list 106 permit ip any 172.20.128.64 0.0.0.31
Device(config)# end
Device# show access-lists
```

```
Extended IP access list 106
10 permit ip any 172.20.128.64 0.0.0.31
Device(config)# interface gigabitethernet1/0/1
Device(config-if)# ip access-group 106 in
```

Example: numbered ACLs

This section provides details about numbered ACLs examples.

In this example, network 10.0.0.0 is a Class A network. Its second octet specifies a subnet, and its subnet mask is 255.255.0.0. The third and fourth octets in a 10.0.0.0 network address specify a particular host. Using access list 2, the switch accepts one address on subnet 48 and rejects all other addresses on that subnet. The last line of the list shows that the switch accepts addresses on all other subnets of network 10.0.0.0. The ACL is applied to packets entering a port.

```
Device(config)# access-list 2 permit 10.48.0.3
Device(config)# access-list 2 deny 10.48.0.0 0.0.255.255
Device(config)# access-list 2 permit 10.0.0.0 0.255.255.255
Device(config)#

Device(config-if)# ip access-group 2 in
```

Examples: extended ACLs

This section provides details about extended ACLs examples.

In this example, the first line permits any incoming TCP connections with destination ports greater than 1023. The second line permits incoming TCP connections to the Simple Mail Transfer Protocol (SMTP) port of host 128.88.1.2. The third line permits incoming ICMP messages for error feedback.

```
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 gt 1023
Device(config)# access-list 102 permit tcp any host 128.88.1.2 eq 25
Device(config)# access-list 102 permit icmp any any
Device(config)#

Device(config-if)# ip access-group 102 in
```

In this example, suppose that you have a network connected to the Internet, and you want any host on the network to be able to form TCP connections to any host on the Internet. However, you do not want IP hosts to be able to form TCP connections to hosts on your network, except to the mail (SMTP) port of a dedicated mail host.

SMTP uses TCP port 25 on one end of the connection and a random port number on the other end. The same port numbers are used throughout the life of the connection. Mail packets coming in from the Internet have a destination port of 25. Because the secure system of the network always accepts mail connections on port 25, incoming are separately controlled.

```
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 23
Device(config)# access-list 102 permit tcp any 128.88.0.0 0.0.255.255 eq 25
Device(config)#

Device(config-if)# ip access-group 102 in
```

Examples: named ACLs

This section provides details about named ACLs examples.

Creating named standard and extended ACLs

This example creates a standard ACL named *internet_filter* and an extended ACL named *marketing_group*. The *internet_filter* ACL allows all traffic from the source address 1.2.3.4.

```
Device(config)# ip access-list standard Internet_filter
Device(config-ext-nacl)# permit 1.2.3.4
Device(config-ext-nacl)# exit
```

The *marketing_group* ACL allows any TCP Telnet traffic to the destination address and wildcard 171.69.0.0 0.0.255.255 and denies any other TCP traffic. It permits ICMP traffic, denies UDP traffic from any source to the destination address range 171.69.0.0 through 179.69.255.255 with a destination port less than 1024, denies any other IP traffic, and provides a log of the result.

```
Device(config)# ip access-list extended marketing_group
Device(config-ext-nacl)# permit tcp any 171.69.0.0 0.0.255.255 eq telnet
Device(config-ext-nacl)# deny tcp any any
Device(config-ext-nacl)# permit icmp any any
Device(config-ext-nacl)# deny udp any 171.69.0.0 0.0.255.255 lt 1024
Device(config-ext-nacl)# deny ip any any log
Device(config-ext-nacl)# exit
```

The *Internet_filter* ACL is applied to outgoing traffic and the *marketing_group* ACL is applied to incoming traffic on a Layer 3 port.

```
Device(config)# interface gigabitethernet3/0/1

Device
Device(config-if)# ip address 2.0.5.1 255.255.255.0
Device(config-if)# ip access-group Internet_filter out
Device(config-if)# ip access-group marketing_group in
```

Deleting individual ACEs from named ACLs

This example shows how you can delete individual ACEs from the named access list *border-list*:

```
Device(config)# ip access-list extended border-list
Device(config-ext-nacl)# no permit ip host 10.1.1.3 any
```

Monitor IPv4 ACLs

Explains monitoring techniques for IPv4 ACLs, offering methods to observe ACL activity, assess effectiveness, and troubleshoot operations to maintain robust network security.

You can monitor IPv4 ACLs by displaying the ACLs configured on the switch and those applied to interfaces and VLANs .

Table 119: Commands for Displaying Access Lists and Access Groups

When you use the **ip access-group** interface configuration command to apply ACLs to a Layer 2 or 3 interface, you can display the access groups on the interface. You can also display the MAC ACLs applied to a Layer 2 interface. You can use the privileged EXEC commands as described in this table to display this information.

Command	Purpose
show access-lists [*number* | *name*]	Displays the contents of one or all current IP and MAC address access lists or a specific access list (numbered or named).
show ip access-lists [*number* | *name*]	Displays the contents of all current IP access lists or a specific IP access list (numbered or named).
show ip interface *interface-id*	Displays detailed configuration and status of an interface. If IP is enabled on the interface and ACLs have been applied by using the **ip access-group** interface configuration command, the access groups are included in the display.
show running-config [**interface** *interface-id*]	Displays the contents of the configuration file for the switch or the specified interface, including all configured MAC and IP access lists and which access groups are applied to an interface.

131 Downloadable ACL

Topics:

- Downloadable ACLs

Outlines the Downloadable ACL feature, covering its functionality, history, scale considerations, configuration steps in ISE and controllers, guidelines, and verification procedures for centralized ACL management using Cisco ISE.

Downloadable ACLs

Introduces the Downloadable ACL feature, detailing its capabilities, feature history, scale considerations, configuration in Cisco ISE and controllers, guidelines and restrictions, explicit authorization server setup, and methods for verifying dACL configuration.

A downloadable ACL is a type of access control list that

- restricts network access to users or devices based on predefined criteria
- is specified as a list of Access Control Entries (ACEs), and
- can be maintained centrally in Cisco ISE and downloaded to controllers.

Supporting reference information

You can easily maintain and update ACLs in Cisco ISE.

Each ACE has a matching condition based on packet header fields:

- IP addresses
- ports
- protocols
- combination of IP addresses, ports, and protocols
- Result (permit or deny)

ACLs are applied to each controller on a per-wireless-client basis.

Typically, you can configure ACLs in a controller itself. However, you can also configure ACLs on a connected Cisco ISE server and download them to the controller when a wireless client connects.

These ACLs are referred to as downloadable ACLs, per-user Dynamic ACLs, or dACLs.

You can easily maintain downloadable ACLs because they define or update ACLs in Cisco ISE and can be downloaded to all applicable controllers.

(In Cisco IOS-XE 17.8 and earlier releases, you must configure the name in Cisco ISE and define the ACL individually on each controller.)

Feature history for downloadable ACL

Downloadable ACL feature history.

This table provides release and related information about the feature explained in this section. This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Release	Feature	Feature Information
Cisco IOS XE Dublin 17.10.1	Downloadable ACL	The Downloadable ACL (dACL) feature defines and updates access control lists (ACLs) in one place (Cisco ISE) and allows ACL download to all the applicable controllers. In Cisco IOS-XE 17.8 and earlier releases, you had to configure the name in Cisco ISE and define the ACL individually in each of the controllers. The dACL feature is supported only in a centralized controller with Local mode APs. **Note** The dACL feature is not supported in RLAN environments.

Scale considerations for downloadable ACL

Downloadable ACL scale considerations.

This table provides the ACL scale numbers for controllers.

ACL Scale for Controllers

- Cisco Catalyst 9800-40 Wireless Controller (small or medium): Supports 128 ACLs with 128 ACEs.
- Cisco Catalyst 9800-80 Wireless Controller (large): Supports 256 ACLs and 256 ACEs.

Controllers	ACL Scale
Cisco Catalyst 9800-40 Wireless Controller (small or medium)	Supports 128 ACLs with 128 ACEs.
Cisco Catalyst 9800-80 Wireless Controller (large)	Supports 256 ACLs and 256 ACEs.

Restrictions for downloadable ACL

Outlines downloadable ACL guidelines and restrictions.

Restrictions on dACL usage

These restrictions apply to downloadable ACLs (dACL):

- dACL does not support FlexConnect local switching.
- You can use IPv6 dACLs only in Cisco ISE 3.0 or later.
- You can use the dACL feature only in a centralized controller that uses Local mode Access Points.

> **Note**
>
> The dACL feature is not supported in RLAN environments.

Configuring dACL name and definition in Cisco ISE

Configure dACL Name and Definition in Cisco ISE.

Before you configuring a dACL in a controller, you must configure the dACL name and definition in Cisco ISE. For more information, refer to Configure Per-User Dynamic Access Control Lists in ISE.

The RADIUS authorization response for a client assigned downloadable ACL includes the Cisco AV pair *ACS:CiscoSecure-Defined-ACL* with a value in this format: `#ACSACL#-IP-<ACL_NAME>-<ACL_HASH>`. For example, a dACL named *ACL_USER1* may be returned as `#ACSACL#-IP-ACL_USER1-65e89aab`. Use the ACL name and hash generated for that dACL by Cisco ISE. The example hash is provided only as a sample.

Configure dACL in a controller (CLI)

Configure dACL in a controller using commands.

Before you begin

Ensure you have configured the RADIUS server and the **aaa-override** command in the policy profile. For more information, see **Configuring AAA for Local Authentication (CLI)**.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the wireless profile policy.

   Example

   ```
   Device(config)# wireless profile policy policy-profile-name
   ```

3. Configure AAA override to apply policies coming from the Cisco ISE servers.

   Example

   ```
   Device(config-wireless-policy)# aaa-override
   ```

4. Enable the profile policy.

Example

```
Device(config-wireless-policy)# no shutdown
```

Configure explicit authorization server list (CLI)

Configure explicit authorization server list using commands.

Deploy an explicit authorization server list for network authorization.

This configuration is used to set up RADIUS servers for network authorization in a Cisco device.

Before you begin

Ensure you have the necessary RADIUS server details and access to the device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the RADIUS server name.

   Example

   ```
   Device(config)# radius server server-name
   ```

3. Specify the RADIUS server parameters.

   Example

   ```
   Device(config-radius-server)# address ipv4 ip-address
   ```

4. Specify the authorization and encryption key used between the device and the RADIUS server.

   Example

   ```
   Device(config-radius-server)# pac key key
   ```

5. Return to the configuration mode.

   Example

   ```
   Device(config-radius-server)# exit
   ```

6. Create a RADIUS server-group identification.

Example

```
Device(config)# aaa group server radius server-group-name
```

server-group-name refers to the server group name. The valid range is from 1 to 32 alphanumeric characters.

7. Create an authorization method list for web-based authorization.

Example

```
Device(config)# aaa authorization network authorization-list group
server-group-name
```

You must use the already created authorization method list.

8. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

The explicit authorization server list is now configured and ready for use.

Verify dACL configuration

Verifies dACL configuration.

To verify the dACL, use this command:

```
Device# show wireless client mac-address <client_mac> detail
                Local Policies:
                Service Template : wlan_svc_named-policy-profile_1_local
(priority 254)
                VLAN             : 16
                Absolute-Timer   : 1800
                Server Policies:
                ACS ACL          : xACSACLx-IP-tftpv4_2-62de6299
                ACS ACL          : xACSACLx-IPV6-tftpv6_2-62de8087
                Resultant Policies:
                ACS ACL          : xACSACLx-IP-tftpv4_2-62de6299
                ACS ACL          : xACSACLx-IPV6-tftpv6_2-62de8087
                VLAN Name        : VLAN0016
                VLAN             : 16
                Absolute-Timer   : 1800
```

To verify dACLs, use this commands:

```
Device# show ip access-lists xACSACLx-IP-tftpv4_2-62de6299
                Extended IP access list xACSACLx-IP-tftpv4_2-62de6299
                1 deny ip any host 9.8.29.13
                2 permit ip any any (58 matches)
```

```
Device# show ipv6 access-list xACSACLx-IPV6-tftpv6_2-62de8087
                IPv6 access list xACSACLx-IPV6-tftpv6_2-62de8087
                deny ipv6 any host 2001:9:8:29:3AAD:A27A:973A:97CC sequence 1
```

```
                permit ipv6 any any (2 matches) sequence 2
```

To view all the downloaded dACLs, use this command:

```
Device# show ip access-lists
```

132 DNS-Based Access Control Lists

Topics:

- DNS-based access control lists
- URL filters
- Configure ISE for central web authentication
- FlexConnect client IPv6 support with WebAuth pre and post ACL

This chapter explains how to configure and apply DNS-based security features and IPv6 web authentication to manage wireless client traffic. It provides procedures for URL filtering and dynamic IP learning to protect network resources across local and FlexConnect deployment modes.

DNS-based access control lists

Explains how DNS-based access control lists (ACLs) enable wireless network administrators to manage client access to network resources based on DNS domain names.

A DNS-based access control list is a wireless network security feature that

- allows administrators to permit or deny wireless client access to resources based on DNS domain names
- dynamically learns allowed or denied IP addresses through DNS snooping on access points, and
- enables pre-authentication and post-authentication URL filtering for enhanced client security.

DNS-based ACLs are used for wireless client devices. You can set pre-authentication ACLs on the Cisco Catalyst 9800 Series Wireless Controller to determine which data requests are allowed or blocked.

To enable DNS-based ACLs on the controller, configure the allowed or denied URLs for the ACLs. The URLs need to be pre-configured on the ACL.

During the registration phase, DNS-based ACLs allow the client to connect to the configured URLs. The controller is configured with the ACL name provided by the AAA server. If the AAA server returns the ACL name, the controller applies the ACL to the client for web redirection.

During client authentication, the AAA server returns the pre-authentication ACL (url-redirect-acl), which is the attribute name assigned to the ACL by the AAA server. The AP performs DNS snooping for each client until registration is complete and the client reaches the SUPPLICANT PROVISIONING state. When the controller receives the ACL configured with URLs, it sends a CAPWAP payload to the access point to enable DNS snooping for those URLs.

After URL snooping is enabled, the access point learns the IP address of the resolved domain name from the DNS response. If the domain name matches the configured URL, the access point parses the DNS response for the IP address and sends the IP address to the controller as a CAPWAP payload. The controller adds the IP address to the allowed list, which enables the client to access the configured URLs.

URL filtering allows access to the IP address for DNS ports 80 or 443.

During pre-authentication or post-authentication, the DNS ACL is applied to the client in the AP. If the client roams from one AP to another AP, tthe DNS-learned IP addresses from the previous access point remain valid on the new AP.

Note

Standard URL filtering is used for local mode, whereas enhanced URL filtering is used for FlexConnect local switching.

Note

Attach the URL filter to a policy profile for local mode. In FlexConnect mode, attach the URL filter to the flex profile; attaching it to a policy profile is not required.

Use the **urlfilter enhanced-list** command only with FlexConnect local switching mode. Do not configure **enhanced-list** in policy profiles for local mode or Flex central switching, as this may cause client connection issues.

Note

DNS-based URLs require an active DNS query from the client. For URL filtering to work, configure DNS settings correctly.

Note

The URL filter takes precedence over punt or redirect ACLs, as well as custom or static pre-authentication ACLs.

Restrictions on DNS-based access control lists

This section provides the restrictions of DNS-based access control list restrictions.

Restrictions for DNS-based ACLs

- In local modes, both Pre-authentication and Post-authentication filters are supported. In Flex (Fabric) mode, only the Pre-authentication filter is supported. ACL override pushed from ISE is not supported.
- FlexConnect Local Switching with External Web authentication, using URL filtering, is not supported until Cisco IOS XE Gibraltar 16.12.x.
- Fully qualified domain name (FQDN) ACLs and DNS-based ACLs are not supported on Cisco Wave 1 Access Points.
- The URL filter considers only the first 20 URLs, though you can add more. It uses regular regex patterns and permits wildcard characters only at the beginning or at the end of an URL. URL ACLs are defined and added to the FlexConnect policy profile, which is then associated with a WLAN. URL ACL creation follows a mechanism similar to local mode URL ACLs.
- In FlexConnect mode, the URL domain ACL works only when it is connected to a FlexConnect policy profile. The ACL can be attached to a WLAN by associating the policy profile with either the WLAN or local policies. However, you can override it using 'url-redirect-acl'. When the Cisco AV pair is received from ISE, the required policy for a client is pushed as part of the ADD MOBILE message.

- When an AP joins or when an existing URL ACL is modified and applied to a FlexConnect profile, the ACL definition, along with the mapped URL filter list, is pushed to the AP. The AP stores the URL ACL definition with the mapped ACL name and snoops DNS packets to learn the first IP address for each URL in the ACL. When the AP learns the IP addresses, it updates the controller of the URL and IP bindings. The controller records this information in the client database for future use.
- When a client roams to another AP during the pre-authentication state, the learned IP addresses are pushed to a new AP. Otherwise, these learned IP addresses are purged when a client enters the post-authentication state or when the TTL for the learned IP address expires.
- The **urlfilter enhanced-list** command is intended only for use with FlexConnect local switching mode. Do not configure **enhanced-list** in policy profiles for local mode or FlexConnect central switching, as this may cause client connection issues.

Restrictions on Wildcard Support in URLs

- The generic wildcard URL, such as ***.*** is not allowed.
- Wildcards between the domain names, such as ***a.cisco.com**, **a.cisco*.com**, **a.b.c.test*.apply.play** are not allowed.
- Multiple wildcards, such as **test.*.cisco.*.com** are not allowed in a URL.
- The wildcard ***.cisco.com** is allowed in the URL.
- The wildcard with a suffix such as **wpr.cisco.*** is a valid configuration.
- A maximum of 16 wildcard-based URLs must be configured for a given ACL.

Define ACLs

This section provides defining ACL details.

Extended ACLs are similar to standard ACLs, but they identify traffic more precisely.

- These commands allow you to define ACLs by name or by identification number.

```
Device(config)#ip access-list extended ?
<100-199> Extended IP access-list number
<2000-2699> Extended IP access-list number (expanded range)
WORD Access-list name
```

- This is the structure of a CLI ACL statement:

```
<sequence number> [permit/deny] <protocol> <address or any> eq <port number>
 <subnet> <wildcard>
```

- For example:

```
1 permit tcp any eq www 192.168.1.0 0.0.0.255
```

The sequence number specifies where to insert the Access Control List Entry (ACE) in the order of ACEs within the ACL. You can define your statements using sequence numbers such as 10, 20, 30, 40, and so on.

The controller GUI allows you to write a complete ACL by going to the Configuration > Security > ACL page. You can view a list of protocols to choose from and make changes to an existing ACL.

Methods for applying ACLs

This section provides methods for applying ACLs.

These are the ways to apply ACLs:

Security ACL: A security ACL defines the types of traffic to allow through the device and the types to block or drop. A security ACL can be applied in various scenarios:

- On SVI interfaces: The ACL is evaluated only against traffic routed through the interface.

```
Device(config)# interface Vlan<number>
```

```
Device(config-if)# ip access-group myACL in/out
```

- On a physical interface of the controller: The ACL is evaluated against all traffic passing through the interface. This method also restricts traffic on the controller management plane, along with SVI interface application.

```
Device(config)#interface GigabitEthernet1
```

```
Device(config-if)#ip access-group myACL in/out
```

In the wireless policy profile or WLAN. You can configure an ACL to apply to wireless client traffic in cases of central or local traffic switching. These ACLs are supported only in the inbound direction.

On the AP during FlexConnect local switching. The ACL is configured and applied from the policy profile on the controller. The ACL must be downloaded to the AP through the FlexConnect profile before it can be applied. Fabric mode APs in Software Defined Access also use Flex ACLs, even though these APs are not operating in FlexConnect mode.

A punt ACL or redirect ACL specifies which traffic is sent to the CPU for further processing instead of normal handling by the dataplane. For example, the Central Web Authentication (CWA) redirect ACL determines which traffic is intercepted and redirected to the web login portal. The ACL does not define traffic to drop or allow. It follows normal processing or forwarding rules and identifies what is sent to the CPU for interception.

A redirect ACL includes an implicit deny statement at the end. This security access list entry drops any traffic not explicitly allowed or sent to the CPU.

Configure ISE for central web authentication (GUI)

Use this procedure to configure ISE for central web authentication using GUI.

Perform these steps to configure ISE for Central Web Authentication using the GUI.

Procedure

1. Login to the Cisco Identity Services Engine (ISE). Click **Policy** and then click **Policy Elements**. Click **Results**.
2. Expand **Authorization** and click **Authorization Profiles**.
3. Click **Add** to create a new authorization profile for URL filter.
4. Enter a name for the profile in the **Name** field. For example, CentralWebauth.
5. Select **ACCESS_ACCEPT** option from the **Access Type** drop-down list. Alternatively, in the **Common Tasks** section, check **Web Redirection**.
6. Select the **Centralized Web Auth** option from the drop-down list. Specify the ACL and select the ACL value from the drop-down list.
7. In the **Advanced Attributes Setting** section, select **Cisco:cisco-av-pair** from the drop-down list.

> **Note**
>
> Multiple ACL can be applied on the controller based on priority. In L2 Auth + webauth multi-auth scenario, if the ISE returns ACL during L2 Auth then ISE ACL takes precedence over the default webauth redirect ACL. This leads to traffic running in webauth pending state, if ISE ACL has permit rule. To avoid this scenario, you need to set the precedence for L2 Auth ISE returned ACL. The default webauth redirect ACL priority is 100. To avoid traffic issue, you need to configure the redirect ACL priority above 100 for ACL returned by ISE.

8. Enter these one by one and click (+) icon after each of them:
   - url-redirect-acl=<sample_name>
   - url-redirect=<sample_redirect_URL>

     For example,

```
Cisco:cisco-av-pair = priv-lvl=15
Cisco:cisco-av-pair = url-redirect-acl=ACL-REDIRECT2
Cisco:cisco-av-pair = url-redirect=
https://9.10.8.247:port/portal/gateway?
sessionId=SessionIdValue&portal=0ce17ad0-6d90-11e5-978e-005056bf2f0a&daysToExpiry=value&action=cwa
```

9. Verify contents in the **Attributes Details** section and click **Save**.

View DNS-based access control lists

This section provides viewing DNS-based access control list details.

To view details of a specified wireless URL filter, use this command:

```
Device# show wireless urlfilter details <urllist_flex_preauth>
```

To view the summary of all wireless URL filters, use this command:

```
Device# show wireless urlfilter summary
```

To view the URL filter applied to the client in the resultant policy section, use this command:

```
Device# show wireless client mac-address <MAC_addr> detail
```

Configuration examples for DNS-based access control lists

This section provides DNS-based access control lists configuration examples.

FlexConnect Mode

Example: Defining URL Filter List

This example shows how to define URL list in FlexConnect mode:

```
Device# configure terminal
Device(config)# urlfilter enhanced-list urllist_flex_pre
```

```
Device(config-urlfilter-params)# url www.dns.com preference 1 action permit
Device(config-urlfilter-params)# end
```

Example: Applying URL Filter List to Flex Profile

This example shows how to apply an URL list to the FlexConnect profile in FlexConnect mode:

```
Device# configure terminal
Device(config)# wireless profile flex default-flex-profile
Device(config-wireless-flex-profile)# acl-policy acl_name
Device(config-wireless-flex-profile-acl)# urlfilter list urllist_flex_preauth
Device(config-wireless-flex-profile-acl)# end
```

Local Mode

Example: Defining Preauth URL Filter List

This example shows how to define URL filter list (pre-authentication):

```
Device# configure terminal
Device(config)# urlfilter list urllist_local_preauth
Device(config-urlfilter-params)# action permit
Device(config-urlfilter-params)# redirect-server-ipv4 9.1.0.101
Device(config-urlfilter-params)# redirect-server-ipv6 2001:300:8::82
Device(config-urlfilter-params)# url url1.dns.com
Device(config-urlfilter-params)# end
```

Example: Defining Postauth URL Filter List

This example shows how to define URL filter list (post-authentication):

```
Device# configure terminal
Device(config)# urlfilter list urllist_local_postauth
Device(config-urlfilter-params)# action permit
Device(config-urlfilter-params)# filter-type post-authentication
Device(config-urlfilter-params)# redirect-server-ipv4 9.1.0.101
Device(config-urlfilter-params)# redirect-server-ipv6 2001:300:8::82
Device(config-urlfilter-params)# url url1.dns.com
Device(config-urlfilter-params)# end
```

Example: Applying URL Filter List to Policy Profile

This example shows how to apply an URL list to the policy profile in local mode:

```
Device# configure terminal
Device(config)# wireless profile policy default-policy-profile
Device(config-wireless-policy)# urlfilter list pre-auth-filter
urllist_local_preauth
Device(config-wireless-policy)# urlfilter list post-auth-filter
urllist_local_postauth
Device(config-wireless-policy)# end
```

Verify DNS snoop agent (DSA)

This section provides how to verify DNS snoop agent (DSA) details.

To view details of the DNS snooping agent client, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
client
```

To view details of the DSA enabled interface, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
client enabled-intf
```

To view the pattern list in uCode memory, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
client hw-pattern-list
```

To view the OpenDNS string for the pattern list, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
client hw-pattern-list odns_string
```

To view the FQDN filter for the pattern list, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
client hw-pattern-list fqdn-filter <fqdn_filter_ID>
```

Note

The valid range of *fqdn_filter_ID* is from 1 to 16.

To view details of the DSA client, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
client info
```

To view the pattern list in CPP client, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
client pattern-list
```

To view the OpenDNS string for the pattern list, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
client pattern-list odns_string
```

To view the FQDN filter for the pattern list, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
client pattern-list fqdn-filter <fqdn_filter_ID>
```

> **Note**
>
> The valid range of *fqdn_filter_ID* is from 1 to 16.

To view details of the DSA datapath, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
datapath
```

To view details of the DSA IP cache table, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
datapath ip-cache
```

To view details of the DSA address entry, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
datapath ip-cache address {ipv4 <IPv4_addr> | ipv6 <IPv6_addr>}
```

To view details of all the DSA IP cache address, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
datapath ip-cache all
```

To view details of the DSA IP cache pattern, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
datapath ip-cache pattern <pattern>
```

To view details of the DSA datapath memory, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
datapath memory
```

To view the DSA regular expression table, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
datapath regexp-table
```

To view the DSA statistics, use this command:

```
Device# show platform hardware chassis active qfp feature dns-snoop-agent
datapath stats
```

URL filters

Explains how URL filters control wireless network web access, the types available, and their interaction with ACLs.

A URL filter is a wireless network access control feature that

- restricts access to websites based on specified URLs
- can operate before or after client authentication, and

- is applied together with an IP-based ACL for granular access control.

There are the two types of URL filters:

- Standard: Standard URL filters can be applied before client authentication (pre-auth) or after a successful client authentication (post-auth). Pre-auth filters are extremely useful for external web authentication because they allow access to the external login page and some internal websites before authentication takes place. After authentication, filters can block specific websites or allow only selected websites, while all other websites are blocked by default. After authentication, this type of URL filtering can be handled more flexibly by using Cisco DNS Layer Security (formerly called Umbrella). The standard URL filters apply the same action (permit or deny) for the whole list of URLs. It is either all permit or all deny. Standard URL filters work only on local mode APs.
- Enhanced: Enhanced URL filters allow you to specify a different action (deny or permit) for each URL in the list and include per-URL hit counters. Enhanced URL filters work only on FlexConnect mode APs.

In both types of URL filters, you can use a wildcard sub-domain such as `*.cisco.com`. URL filters are standalone features but are always applied together with an IP-based ACL. A maximum of 20 URLs are supported in a given URL filter. Because one URL can resolve to multiple IP addresses, each client can track up to 40 resolved IP addresses. URL filters track only DNS records. The controller or APs do not track the resolved IP address if the DNS response uses a CNAME alias record.

> **Note**
>
> If you have a POST-type URL filter and an ACL applied to a policy profile, the ACL blocks traffic to the URL if it does not include permit statements for the URLs. This issue can occur when the URL filter is POST with a permit statement but the ACL does not include a permit statement for the URLs. We recommend creating permit statements within the ACL for the IP addresses of the URLs rather than relying on the POST URL filter.

FlexConnect mode

Explains how to configure a URL filter list for the FlexConnect profile.

Define URL filter list (CLI)

Use this procedure to define URL filter list using commands.

Define a custom list of permitted or blocked URLs for use with FlexConnect local switching mode using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the URL filter enhanced list.

   Example

   ```
   Device(config)# urlfilter enhanced-list urllist_flex_preauth
   ```

> **Note**
>
> Use the **urlfilter enhanced-list** command only with FlexConnect local switching mode. Do not configure **enhanced-list** in policy profiles for local mode or FlexConnect central switching. This action may cause client connection issues.

Here, *list-name* refers to the URL filter list name. The list name must not exceed 32 alphanumeric characters.

3. Configure the action: permit (allowed list) or deny (blocked list).

Example

```
Device(config-urlfilter-enhanced-params)# url url-name preference [0-65535]
action {deny | permit}
```

4. Return to privileged EXEC mode.

Example

```
Device(config-urlfilter-params)# end
```

Apply URL filter list to FlexConnect profile (CLI)

Use this procedure to apply URL filter list to FlexConnect profile using commands.

Use this procedure to enforce web access policies on FlexConnect APs through defined URL filter lists using commands.

Procedure

---

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Create a new flex policy.

Example

```
Device(config)# wireless profile flex default-flex-profile
```

3. Configure ACL policy.

Example

```
Device(config-wireless-flex-profile)# acl-policy acl_name
```

4. Apply the URL list to the Flex profile.

Example

```
Device(config-wireless-flex-profile-acl)# urlfilter list urllist_flex_preauth
```

5. Return to privileged EXEC mode.

Example

```
Device(config-wireless-flex-profile-acl)# end
```

Local mode

Explains how to configure a URL filter list for the Local profile.

Define URL filter list (CLI)

Use this procedure to define URL filter list using commands.

Configure a URL filter list to control access to specific web resources by creating an allowed or blocked list, setting redirect servers, and applying actions for post-authentication filtering using commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the URL filter list.

Example

```
Device(config)# urlfilter list urllist_local_preauth
```

Here, *list-name* refers to the URL filter list name. The list name must not exceed 32 alphanumeric characters.

3. Configure the action: permit (allowed list) or deny (blocked list).

Example

```
Device(config-urlfilter-params)# action permit
```

4. Configure the URL list as post-authentication filter.

Example

```
Device(config-urlfilter-params)# filter-type post-authentication
```

> **Note**
>
> This step is applicable while configuring post-authentication URL filter only.

5. Configure the IPv4 redirect server for the URL list.

   Example

   ```
   Device(config-urlfilter-params)# redirect-server-ipv4 192.0.2.1
   ```

   Here, *IPv4-address* refers to the IPv4 address.

6. Configure the IPv6 redirect server for the URL list.

   Example

   ```
   Device(config-urlfilter-params)# redirect-server-ipv6 2001:DB8::82
   ```

   Here, *IPv6-address* refers to the IPv6 address.

7. Configure an URL.

   Example

   ```
   Device(config-urlfilter-params)# url url1.dns.com
   ```

   Here, *url* refers to the name of the URL.

8. Return to privileged EXEC mode.

   Example

   ```
   Device(config-urlfilter-params)# end
   ```

Apply URL filter list to policy profile (GUI)

Use this procedure to apply URL filter list to policy profile using GUI.

Assign URL filters to a policy profile to manage and restrict user web access using the GUI.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click on the **Policy Name**.
3. Go to **Access Policies** tab.
4. In the **URL Filters** section, select the filters from the **Pre Auth** and **Post Auth** drop-down lists.
5. Click **Update & Apply to Device**.

Apply URL filter list to policy profile (CLI)

Use this procedure to apply URL filter list to policy profile using commands.

Restrict or permit user access to network resources by applying specific URL filter lists to a designated policy profile using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure wireless policy profile.

   Example

   ```
   Device(config)# wireless profile policy default-policy-profile
   ```

   Here, *profile-policy* refers to the name of the WLAN policy profile.

3. Apply the URL list to the policy profile.

   Example

   ```
   Device(config-wireless-policy)# urlfilter list pre-auth-filter
   urllist_local_preauth
   ```

   ```
   Device(config-wireless-policy)# urlfilter list post-auth-filter
   urllist_local_postauth
   ```

   Here, *name* refers to the name of the pre-authentication or post-authentication URL filter list configured earlier.

   > **Note**
   >
   > When a client joins, the policy's configured URL filter applies to the client.

4. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

Configure ISE for central web authentication

Explains how to configure ISE for central web authentication.

Create authorization profiles (GUI)

Use this procedure to create authorization profiles.

Define authorization profiles that control network access through URL filters using the GUI.

Procedure

1. Login to the Cisco Identity Services Engine (ISE). Click **Policy**, and click **Policy Elements**.
2. Click **Results**.
3. Expand **Authorization**, and click **Authorization Profiles**.
4. Click **Add** to create a new authorization profile for URL filter.
5. In the Name field, enter a name for the profile. For example, *CentralWebauth*.
6. Select **ACCESS_ACCEPT** from the access type drop-down list.
7. In the Advanced Attributes Setting section, select **Cisco:cisco-av-pair** from the drop-down list.
8. Enter these parameters one by one and click (+) icon after each of them:
   - url-filter-preauth=<preauth_filter_name>
   - url-filter-postauth=<postauth_filter_name>

   For example,

```
Cisco:cisco-av-pair = url-filter-preauth=urllist_pre_cwa
Cisco:cisco-av-pair = url-filter-postauth=urllist_post_cwa
```

9. Verify contents in the attributes details section and click **Save**.

Map authorization profiles to authentication rule (GUI)

Use this procedure to map authorization profiles to authentication rule.

Define rules that determine how authentication requests are processed by mapping authorization profiles to authentication criteria in the Policy interface using the GUI.

Procedure

1. In the **Policy** > **Authentication** page, click **Authentication**.
2. Enter a name for your authentication rule.

   For example, *MAB*.
3. In the **If condition** field, select the plus (+) icon.
4. Choose **Compound condition** , and choose **WLC_Web_Authentication**.
5. Click the arrow located next to **and ...** in order to expand the rule further.
6. Click the + icon in the Identity Source field, and choose **Internal endpoints**.
7. Select **Continue** from the 'If user not found' drop-down list.

   This option allows a device to be authenticated even if its MAC address is not known.

8. Click **Save**.

Map authorization profiles to authorization rule (GUI)

Use this procedure to map authorization profiles to authorization rule.

Map an authorization profile to an authorization rule to control network access permissions based on selected identity and condition attributes using the GUI.

Procedure

1. Click **Policy** > **Authorization**. In the Rule Name field, enter a name.

   For example, *CWA Post Auth*.
2. In the Conditions field, select the plus (+) icon.
3. Select the **Identity Groups** area to view from the drop-down list. Choose **User Identity Groups** > **user_group**.
4. Click the plus (+) sign located next to **and ...** in order to expand the rule further.
5. In the Conditions field, select the plus (+) icon. Choose **Compound Conditions**, and choose to create a new condition.
6. From the settings icon, select **Add Attribute/Value** from the options.
7. In the Description field, select **Network Access** > **UseCase** as the attribute from the drop-down list. Choose the **Equals** operator.
8. From the right-hand field, choose **GuestFlow**.
9. In the Permissions field, select the plus (+) icon to select a result for your rule.

   You can choose **Standard** > **PermitAccess** option or create a custom profile to return the attributes that you like.

FlexConnect client IPv6 support with WebAuth pre and post ACL

Explains how to configure FlexConnect client IPv6 support with WebAuth pre and post ACL.

FlexConnect client IPv6 support with WebAuth pre and post ACL

This section explains FlexConnect client IPv6 support with WebAuth pre and post ACL details.

A WebAuth IPv6 ACL is an access control mechanism that

- enables an access point to enforce web authentication policies for IPv6 client traffic
- uses IOS IPv6 ACL definitions that are pushed to the AP in response to join events or policy changes, and
- differentiates pre-authentication and post-authentication access by assigning default or custom ACLs.

A change in the ACL policies of the FlexConnect profile can include adding a new ACL, deleting an ACL, or modifying an existing ACL.

ACL definitions are pushed to AP when any of these events occur:

- AP join.
- New ACL mapping in a new FlexConnect profile.
- Configuring IPv6 ACL definition in FlexConnect profile.

Default Local Web Authentication ACLs

The pre-defined default LWA IPv6 ACL is pushed to AP and assigned to the data plane.

Default External Web Authentication ACL

The default EWA ACLs are derived from the redirect portal address configured in the parameter map.

The types of default EWA ACLs include:

- Security ACL which is pushed and assigned to the AP.
- Intercept ACL which is pushed and assigned to the data plane.

FQDN ACL

- FQDN ACL is encoded along with IPv6 ACL and sent to AP.
- FQDN ACL is always a custom ACL.

These applies to FlexConnect and Local mode:

- If you are migrating from AireOS, you would explicitly need to execute these commands:

```
redirect append ap-mac tag ap_mac
```

```
redirect append wlan-ssid tag wlan
```

```
redirect append client-mac tag client_mac
```

- If the login page has any resource that needs to be fetched from the server, you will need to include those resource URLs in URL filtering.
- If you are trying to access an IPv6 URL and you have an IPv4 web server, the controller redirects the client to an internal page. Domain redirection is not supported in this context. It is recommended to have a dual-stack web server and configure virtual IPv6 address in the global parameter map.

Enable pre-authentication ACL for LWA and EWA (GUI)

Use this procedure to enable pre-authentication ACL for LWA and EWA using GUI.

Allow or restrict traffic before authentication on specified WLANs using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**, the **SSID** and the **WLAN ID**.
4. Choose **Security** > **Layer2** tab. Uncheck the **WPAPolicy**, **AES** and **802.1x** check boxes.
5. Choose **Security** > **Layer3** tab. Select the Web Auth Parameter Map from the **Web Auth Parameter Map** drop-down list and authentication list from the **Authentication List** drop-down list.
6. Click **Show Advanced Settings** and under the **Preauthenticated ACL** settings, and select the IPv6 ACL from the **IPv6** drop-down list.
7. Choose **Security** > **AAA** tab. Select the authentication list from the **Authentication List** drop-down list.

8. Click **Apply to Device**.

Enable pre-authentication ACL for LWA and EWA (CLI)

Use this procedure to enable pre-authentication ACL for LWA and EWA using commands.

Configure pre-authentication access control for WLAN client web authentication scenarios using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-demo 1 ssid-demo
   ```

   - *wlan-name*: Enter the profile name. The range is from 1 to 32 alphanumeric characters.
   - *wlan-id*: Enter the WLAN ID. The range is from 1 to 512.
   - *SSID-name*: Enter the Service Set Identifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID.

   > **Note**
   >
   > If you have already configured WLAN, enter **wlan wlan-name** command.

3. Create a pre-authentication ACL for web authentication and disable the WPA security.

   Example

   ```
   Device(config-wlan)# ipv6 traffic-filter web preauth_v6_acl
   ```

   ```
   Device(config-wlan)# no security wpa
   ```

4. Disable WPA2 ciphers for AES.

   Example

   ```
   Device(config-wlan)#no security wpa wpa2 ciphers aes
   ```

5. Disable security AKM for dot1x.

Example

```
Device(config-wlan)#no security wpa akm dot1x
```

6. Configure web authentication.

Example

```
Device(config-wlan)# security web-auth
```

7. Enable authentication list for WLAN.

Example

```
Device(config-wlan)# security web-auth authentication-list wcm_dot1x
```

8. Map the parameter map.

Example

```
Device(config-wlan)# security web-auth parameter-map param-custom-webconsent
```

9. Shutdown the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

Enable post-authentication ACL for LWA and EWA (GUI)

Use this procedure to enable post-authentication ACL for LWA and EWA using GUI.

Enable post-authentication access control for LWA and EWA WLANs to enforce security policies after successful user authentication using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**. The **Profile Name** is the profile name of the policy profile.
4. Enter the **SSID** and the **WLAN ID**.
5. Click **Apply to Device**.

Enable post-authentication ACL for LWA and EWA (CLI)

Use this procedure to enable post-authentication ACL for LWA and EWA using commands.

Configure security policies that restrict client access after authentication using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create policy profile for the WLAN.

   Example

   ```
   Device(config)# wireless profile policy test1
   ```

   The *profile-name* is the profile name of the policy profile.

3. Create a named WLAN ACL.

   Example

   ```
   Device(config-wireless-policy)# ipv6 acl testacl
   ```

4. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

   Alternatively, you can also press ***Ctrl-Z*** to exit global configuration mode.

Enable DNS ACL for LWA and EWA (GUI)

Use this procedure to enable DNS ACL for LWA and EWA (GUI).

Enable DNS-based access control for LWA and EWA WLANs by configuring the required settings using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**. The **Profile Name** is the profile name of the policy profile.
4. Enter the **SSID** and the **WLAN ID**.
5. Click **Apply to Device**.

Enable DNS ACL for LWA and EWA (CLI)

Use this procedure to enable DNS ACL for LWA and EWA using commands.

Enable DNS ACLs to control DNS access for WLAN policy profiles that use LWA or EWA using commands.

> **Note**
>
> Post-authentication DNS ACL is not supported.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create policy profile for the WLAN.

   Example

   ```
   Device(config)# wireless profile policy test1
   ```

   The *profile-name* is the profile name of the policy profile.

3. Return to the privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   Alternatively, you can also press ***Ctrl-Z*** to exit global configuration mode.

Verify FlexConnect client IPv6 support with WebAuth pre and post ACL

This section provides verifying FlexConnect client IPv6 support with WebAuth pre and post ACL details.

To verify the client state after L2 authentication, use this command:

```
Device# show wireless client summary
Number of Local Clients: 1

MAC Address    AP Name                          WLAN  State            Protocol
 Method     Role
-------------------------------------------------------------------------------------------
1491.82b8.f8c1 AP4001.7A03.544C                  4      Webauth Pending   11n(5)
   None       Local
Number of Excluded Clients: 0
```

To verify the IP state, discovery, and MAC, use this command:

```
Device# show wireless dev da ip
IP                                          STATE       DISCOVERY   MAC
-------------------------------------------------------------------------------
15.30.0.4                                   Reachable   ARP
1491.82b8.f8c1
```

```
2001:15:30:0:d1d7:ecf3:7940:af60            Reachable   IPv6 Packet
1491.82b8.f8c1
fe80::595e:7c29:d7c:3c84                    Reachable   IPv6 Packet
1491.82b8.f8c1
```

133 Allowed List of Specific URLs

Topics:

- Allowed lists of specific URLs

Allowed lists of specific URLs

Introduces the concept of an allowed list for specific URLs and provides instructions for adding URLs to the list, addresses issues with portals resolving to multiple IP addresses, and explains verification procedures for URL allowed-listing.

An allowed list of specific URLs is a wireless controller feature that

- lets you add URLs to a list on the controller or access point so designated sites remain accessible
- enables web authentication use cases such as captive portals and walled gardens, and
- allows users to reach permitted sites before authenticating, even without full internet access.

Additional reference information

You do not need to authenticate to access URLs in the allowed list. If you try to reach a site not in the allowed list, you are redirected to the login page.

Accessing a support site

If "support.example.com" is added to the allowed list, users can access this site for help or information prior to logging into the network.

Add URL to allowed list

Learn how to add a URL to the allowed list.

To configure a URL filter that allows specific URLs while blocking others.

This procedure is used in network environments where URL filtering is necessary to control access to web resources.

Before you begin

Ensure you have administrative access to the device and that URL filtering is supported.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the URL filter profile.

   Example

   ```
   Device(config)# urlfilter list url-allowedlist-nbn
   ```

3. Set the action for the URL filter list.

   Example

   ```
   Device(config-urlfilter-params)# action permit
   ```

   Configures the list as an allowed list, and the **deny** command configures the list as a blocked list.

4. Configure the IP address of the redirect servers to which user requests will be redirected in case of denied requests.

Example

```
Device(config-urlfilter-params)# redirect-server-ipv4 X.X.X.X
```

5. Configure the URL to be allowed.

Example

```
Device(config-urlfilter-params)# url www.cisco.com
```

Note

redirect-server-ipv4 and **redirect-server-ipv6** is applicable only in the local mode, specifically in post-authentication. For any further tracking or displaying any warning messages, the denied user request is redirected to the configured server.

But the **redirect-server-ipv4** and **redirect-server-ipv6** configurations do not apply to pre-authentication scenario as you will be redirected to the controller for the redirect login URL for any denied access.

You can associate the allowed URL with the ACL policy in flex profile.

Associating the allowed URL with the ACL policy in flex profile:

```
Device(config)# wireless profile flex default-flex-profile
            Device(config-wireless-flex-profile)# acl-policy user_v4_acl
            Device(config-wireless-flex-profile-acl)# urlfilter list
url_allowedlist_nbn
            Device(config-wireless-flex-profile-acl)# exit
            Device(config-wireless-flex-profile)# description "default flex
profile"
```

```
Device(config)# urlfilter enhanced-list urllist_pre_cwa
                Device(config-urlfilter-enhanced-params)# url url1.dns.com
preference 1 action permit
                Device(config-urlfilter-enhanced-params)# url url2.dns.com
preference 2 action deny
                Device(config-urlfilter-enhanced-params)# url url3.dns.com
preference 3 action permit
```

```
Device(config)# wlan wlan5 5 wlan5
                    Device(config-wlan)#ip access-group web user_v4_acl
                    Device(config-wlan)#no security wpa
                    Device(config-wlan)#no security wpa
                    Device(config-wlan)#no security wpa wpa2 ciphers aes
                    Device(config-wlan)#no security wpa akm dot1x
                    Device(config-wlan)#security web-auth
                    Device(config-wlan)#security web-auth authentication-list
 default
                  Device(config-wlan)#security web-auth parameter-map global

                    Device(config-wlan)#no shutdown
```

Portal resolving to multiple IP addresses

This section lists the network controller behavior and configuration steps required when a portal resolves to multiple IP addresses.

You can use two IP addresses with the controller. However, the Web Auth Parameter Map allows pre-authentication to only one IP address. If an external portal, such as Cisco Spaces, resolves to multiple IP addresses or other HTTP resources require pre-authentication, you need a URL filter. The URL filter dynamically permits traffic to the configured URLs by snooping DNS requests and adding each resolved IP address to the intercept (redirect) and security (pre-auth) ACLs. This process enables you to access clients when necessary.

Additional information

- For FlexConnect deployments with local switching, apply the URL filter to the client at the AP for DNS snooping to work properly.
- Configuring the Web Auth Parameter Map automatically creates the following ACLs:
  - Redirect or intercept ACL (WA-v4-int): handles URL redirection.
  - Security ACL (WA-sec-): permits pre-auth access for HTTP or HTTPS, DNS, DHCP, and other required services.

Configuration steps for URL filter in FlexConnect profile

If you miss this step, the AP cannot dynamically add IP addresses. Clients may fail to redirect to the portal page when they request a secondary portal IP address.

> **Note**
>
> Tip: Use visuals, diagrams, or tables as needed to clarify configuration mapping, ACL lists, or traffic flow.

Associating the allowed URL with the ACL policy in a Flex profile

```
                Device(config)# wireless profile flex default-flex-profile
                Device(config-wireless-flex-profile)# acl-policy WA-v4-<ip>
(security ACL)
                Device(config-wireless-flex-profile-acl)# urlfilter list
url_allowedlist_nbn
                Device(config-wireless-flex-profile-acl)# exit
                Device(config-wireless-flex-profile)# description "default
flex profile"
```

Verify URLs on the allowed list

Verify URLs on the Allowed List.

```
Device# show wireless urlfilter summary
Black-list    - DENY
White-list    - PERMIT
Filter-Type   - Specific to Local Mode

URL-List                          ID  Filter-Type  Action   Redirect-ipv4
Redirect-ipv6
```

```
_____________________________________________________________________________________
url-whitelist                    1    PRE-AUTH     PERMIT   192.0.2.1

Device#

Device# show wireless urlfilter details url-whitelist
List Name..................... : url-whitelist
Filter ID..................... : : 1
Filter Type................... : PRE-AUTH
Action........................ : PERMIT
Redirect server ipv4...... : 192.0.2.1
Redirect server ipv6...... :
Configured List of URLs
   URL........................ : www.example.com
```

134 Cisco Umbrella WLAN

Topics:

- Cisco Umbrella WLAN

This chapter explains how to register and configure an embedded wireless controller with the cloud DNS-security service, covering certificate import, parameter-map setup, DNS encryption, UDP timeouts, policy application, and verification commands to enforce DNS-layer protection.

Cisco Umbrella WLAN

Introduces Cisco Umbrella WLAN, detailing core integration concepts, the process for registering the Embedded Wireless Controller to a Cisco Umbrella account, and explaining DNS enforcement restrictions for Umbrella WLAN deployments.

A Cisco Umbrella WLAN is a cloud-based security solution for wireless networks that

- provides DNS-level protection by detecting and blocking both known and emergent threats
- enables granular policy configuration per user group, network, device, and IP address, and
- offers real-time security dashboards and scheduled reporting.

Additional features and policy configuration

- Allows policy configuration at a single point for each user group and supports local policy, AP group, and WLAN priority order.
- Publishes visual, real-time activity dashboards with aggregated reports.
- Supports up to 60 content categories and lets administrators add custom allowed and blocked list entries.
- Enables scheduled email reporting.
- Supports custom Umbrella profiles (one global, up to 15 custom profiles).
- Flexconnect mode allows DNS handling within the AP, supporting multiple profiles in Flex mode.
- Provides communication over IPv6, though device registration occurs only through IPv4.

Policy priority order

When you configure multiple policies in Cisco Umbrella WLAN, you enforce them in order of priority:

1. **Local policy** – This is the most specific policy and is applied first. Local policies override any broader group or network rules for individual devices or users.
2. **AP group policy** – If no local policy exists, the rules defined for the AP group are applied next.
3. **WLAN policy** – If there are no local or AP group policies, the general policy for the entire wireless network (WLAN) is applied.

To summarize, the most specific rules take priority over broader ones. If a local rule exists, it is enforced first. If not, the system checks for an AP group rule. If neither exists, the general WLAN policy is used.

Register Controller to Cisco Umbrella account

This process explains how Controller are registered with Cisco Umbrella accounts, including the required components, step-by-step stages, and considerations for DNS query handling and profile application.

This section describes the process followed to register the Controller to the Cisco Umbrella account. Here are the prerequisites:

- Have an account with Cisco Umbrella.
- Have an API token from Cisco Umbrella.

Summary

Registering Controller to Cisco Umbrella accounts enables centralized DNS security management and policy enforcement for wireless devices. This integration requires account access, API tokens, and specific parameter mappings.

The key components involved in the process are:

- embedded wireless controllercontroller: Wireless network devices that will be registered with Cisco Umbrella for DNS query management.
- **Cisco Umbrella account**: The cloud-based platform providing DNS security services and device identity management.
- **API token**: A credential generated from Cisco Umbrella and used for authenticating and mapping devices during registration.
- **Umbrella parameter map**: Controller-side configuration map that links API tokens to device profiles.
- **DHCP servers and DNS servers**: Network services facilitating IP assignment and DNS query redirection for WLAN clients.

This process involves obtaining an API token, applying it to the controller, and verifying device registration in Cisco Umbrella.

Workflow

These stages describe the registration of the Controller to Cisco Umbrella Account.

1. **Obtain prerequisites:** Ensure you have a Cisco Umbrella account and generate an API token from the Umbrella dashboard.
2. **Configure umbrella parameter map on the controller:** Create an Umbrella parameter map on the controller embedded wireless controller, assigning the API token.
3. **Register the device:** Register the controller to the Cisco Umbrella server using the configured parameter map. Cisco Umbrella responds with a device ID, mapped in a one-to-one relationship with the parameter map name.
4. **Verify device registration:** In the Cisco Umbrella dashboard, confirm that the controller embedded wireless controller appears under Device Name with its identity information.
5. **Apply the API token configuration:** Register the Cisco Umbrella API token on the network through controller configuration.
6. **Redirect DNS queries:** Once registration and parameter mapping are complete, DNS queries from WLAN clients not matching local RegEx domain settings are redirected to the Umbrella DNS resolver, with encryption applied based on DNScrypt options in the parameter map.
7. **Apply profiles and handle mobility:** Apply wireless Cisco Umbrella profiles to WLAN or AP groups if device registration is successful. For Layer 3 mobility, always apply Cisco Umbrella on the anchor controller.
8. **Distribute Umbrella server IPs through DHCP:** When two DNS servers are configured under DHCP, two Cisco Umbrella server IPs are sent to clients using DHCP option 6; only one IP is sent if a single DNS server is configured.

Result

Controller are successfully registered with Cisco Umbrella accounts. Wireless entities can be managed with Umbrella profiles, and DNS queries are securely redirected according to policy. Device identity, profile application, and DNS traffic handling are centrally managed.

For more information on the Cisco Umbrella configurations, see the Integration for ISR 4K and ISR 1100 – Security Configuration Guide.

Restrictions for Umbrella WLAN

Cisco Umbrella WLAN relies on DNS traffic for protection enforcement; certain scenarios may bypass these safeguards.

- Wireless Cisco Umbrella profiles can be applied only to WLAN or AP groups after successful device registration.

- In Layer 3 mobility environments, Cisco Umbrella must be enabled on the anchor Controller Embedded wireless controllers.
- DHCP configuration determines how many Cisco Umbrella DNS server IPs are distributed to clients.

DNS dependency restrictions

Cisco Umbrella WLAN protections are only effective when client devices perform DNS queries through the network. If DNS is bypassed, Umbrella protection does not apply.

- If an application or host connects directly using an IP address, bypassing DNS resolution, Umbrella WLAN protections are not enforced.

These restrictions apply to all deployments of Cisco Umbrella WLAN where DNS traffic is required for policy enforcement.

Umbrella WLAN leverages DNS queries to monitor and enforce security policies. Direct IP or proxy-based connections circumvent DNS-based protections.

Clients that bypass DNS, or use web proxies that resolve server addresses without DNS queries, may not be protected by Umbrella WLAN security policies.

Ensure all client devices perform DNS queries through the WLAN and avoid proxy configurations that bypass DNS to maintain optimal Umbrella protection.

Configure Cisco Umbrella WLAN

Guides Cisco Umbrella WLAN configuration, providing step-by-step instructions for importing CA certificates, creating local domain RegEx parameter maps, configuring Umbrella parameters and GUI settings, enabling or disabling DNScrypt, adjusting UDP session timeouts, establishing Flex Profiles and Policy Profiles (CLI and GUI), and verifying configuration.

To configure Cisco Umbrella on the Controller, you need to meet these prerequisites:

- Obtain the API token from the Cisco Umbrella dashboard.
- You must have the root certificate to establish an HTTPS connection with the Cisco Umbrella registration server: api.opendns.com. Import the root certificate from digicert.com to the Controller using the command **crypto pki trustpool import terminal** .

Import CA certificate to the trust pool

Learn how to import a CA certificate to the trust pool.

Before you begin

Ensure you have the root certificate and an established HTTPS connection with the Cisco Umbrella registration server.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Perform either of these steps:

- Import the root certificate directly from the Cisco website. The Trustpool bundle contains the root certificate of *digicert.com* together with other CA certificates.

```
Device(config)# crypto pki trustpool import url
http://www.cisco.com/security/pki/trs/ios.p7b
```

- Import the root certificate by executing the **import terminal** command.

```
Device(config)# crypto pki trustpool import terminal
```

- Import the root certificate by pasting the CA certificate from the digicert.com. Enter the PEM-formatted CA certificate from these location. See the Related Information section to download the CA certificate.

```
-----BEGIN CERTIFICATE-----
MIIE6jCCA9KgAwIBAgIQCjUI1VwpKwF9+K1lwA/35DANBgkqhkiG9w0BAQsFADBhMQswCQYDVQQG
EwJVUzEVMBMGA1UEChMMRGlnaUNlcnQgSW5jMRkwFwYDVQQLExB3d3cuZGlnaWNlcnQuY29tMSAw
HgYDVQQDExdEaWdpQ2VydCBHbG9iYWwgUm9vdCBDQTAeFw0yMDA5MjQwMDAwMDBaFw0zMDA5MjMy
MzU5NTlaME8xCzAJBgNVBAYTAlVTMRUwEwYDVQQKEwxEaWdpQ2VydCBJbmMxKTAnBgNVBAMTIERp
Z2lDZXJ0IFRMUyBSU0EgU0hBMjU2IDIwMjAgQ0ExMIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIB
CgKCAQEAwUuzZUdwvN1PWNvsnO3DZuUfMRNUrUpmRh8sCuxkB+Uu3Ny5CiDt3+PE0J6aqXodgojl
EVbbHp9YwlHnLDQNLtKS4VbL8Xlfs7uHyiUDe5pSQWYQYE9XE0nw6Ddng9/n00tnTCJRpt8OmRDt
V1F0JuJ9x8piLhMbfyOIJVNvwTRYAIuE//i+p1hJInuWraKImxW8oHzf6VGo1bDtN+I2tIJLYrVJ
muzHZ9bjPvXj1hJeRPG/cUJ9WIQDgLGBAfr5yjK7tI4nhyfFK3TUqNaX3sNk+crOU6JWvHgXjkkD
Ka77SU+kFbnO8lwZV21reacroicgE7XQPUDTITAHk+qZ9QIDAQABo4IBrjCCAaowHQYDVR0OBBYE
FLdrouqoqoSMeeq02g+YssWVdrn0MB8GA1UdIwQYMBaAFAPeUDVW0Uy7ZvCj4hsbw5eyPdFVMA4G
A1UdDwEB/wQEAwIBhjAdBgNVHSUEFjAUBggrBgEFBQcDAQYIKwYBBQUHAwIwEgYDVR0TAQH/BAgw
BgEB/wIBADB2BggrBgEFBQcBAQRqMGgwJAYIKwYBBQUHMAGGGGh0dHA6Ly9vY3NwLmRpZ2ljZXJ0
LmNvbTBABggrBgEFBQcwAoY0aHR0cDovL2NhY2VydHMuZGlnaWNlcnQuY29tL0RpZ2lDZXJ0R2xv
YmFsUm9vdENBLmNydDB7BgNVHR8EdDByMDegNaAzhjFodHRwOi8vY3JsMy5kaWdpY2VydC5jb20v
RGlnaUNlcnRHbG9iYWxSb290Q0EuY3JsMDegNaAzhjFodHRwOi8vY3JsNC5kaWdpY2VydC5jb20v
RGlnaUNlcnRHbG9iYWxSb290Q0EuY3JsMDAGA1UdIAQpMCcwBwYFZ4EMAQEwCAYGZ4EMAQIBMAgG
BmeBDAECAjAIBgZngQwBAgMwDQYJKoZIhvcNAQELBQADggEBAHert3onPa679n/gWlbJhKrKW3EX
3SJH/E6f7tDBpATho+vFScH90cnfjK+URSxGKqNjOSD5nkoklEHIqdninFQFBstcHL4AGw+oWv8Z
u2XHFq8hVt1hBcnpj5h232sb0HIMULkwKXq/YFkQZhM6LawVEWwtIwwCPgU7/uWhnOKK24fXSuhe
50gG66sSmvKvhMNbg0qZgYOrAKHKCjxMoiWJKiKnpPMzTFuMLhoClw+dj20tlQj7T9rxkTgl4Zxu
YRiHas6xuwAwapu3r9rxxZf+ingkquqTgLozZXq8oXfpf2kUCwA/d5KxTVtzhwoT0JzI8ks5T1KE
SaZMkE4f97Q=
-----END CERTIFICATE-----
```

3. Optional: Import the root certificate by entering the quit command.

Example

```
Device(config)# quit
```

> **Note**
>
> You will receive a message after the certificate has been imported.

Related information

https://www.digicert.com/CACerts/DigiCertTLSRSASHA2562020CA1.crt

Create a local domain RegEx parameter map

Learn how to create a Local Domain RegEx Parameter Map using commands.

To create a regex parameter map for matching specific domain patterns.

This procedure is used in network configurations where domain name filtering is required based on regex patterns.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a regex parameter map.

   Example

   ```
   Device(config)# parameter-map type regex parameter-map-name
   ```

3. Configure the regex pattern to match.

   Example

   ```
   Device(config-profile)# pattern regex-pattern
   ```

   Example

   ```
   Device(config-profile)# pattern www.google.com
   ```

   Note

   These patterns are supported:

   - Begins with `.*`. For example: **`.*facebook.com`**
   - Begins with `.*` and ends with `*`. For example: **`.*google*`**
   - Ends with `*`. For example: **`www.facebook*`**
   - No special character. For example: **`www.facebook.com`**

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-profile)# end
   ```

The local domain RegEx parameter map is created and configured successfully.

Configure parameter map name in WLAN (GUI)

Use this procedure to configure the parameter map name in a WLAN using the graphical user interface (GUI) to apply security and access policies for wireless clients..

Assign a parameter map to a WLAN, enabling custom policy enforcement for DNS traffic and access restrictions.

This procedure ensures that your WLAN profile enforces the intended policies for DNS and wireless client access using Cisco Umbrella integration.

Before you begin

Confirm that required parameter maps are defined

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy** .
2. Click on the Policy Profile Name. The **Edit Policy Profile** window is displayed.
3. Choose the **Advanced** tab.
4. In the **Umbrella** settings, from the **Umbrella Parameter Map** drop-down list, choose the parameter map.
5. Enable or disable **Flex DHCP Option for DNS** and **DNS Traffic Redirect** toggle buttons.
6. Click Update & Apply to Device.

The WLAN profile is updated with the selected parameter map, and the associated DNS and access policies are enforced for connected wireless clients.

Configure the Umbrella parameter map

Configure the Umbrella Parameter Map using commands.

To configure the Umbrella Parameter Map for network security and management.

This configuration is used to manage and secure network traffic through the Umbrella service, which provides DNS-layer security.

Before you begin

Ensure you have the necessary permissions and access to the device for configuration.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create an Umbrella global or customized parameter map.

   Example

   ```
   Device(config)# parameter-map type umbrella global parameter-map-name
   ```

Example

```
Device(config)# parameter-map type umbrella custom_pmap
```

Creates an umbrella global or customized parameter map.

3. Configure an Umbrella token.

Example

```
Device(config-profile)# token token-value
```

Example

```
Device(config-profile)# token 5XXXXXXXXCXXXXXXXAXXXXXXXFXXXXCXXXXXXXX
```

4. Configure local domain RegEx parameter maps.

Example

```
Device(config-profile)# local-domain regex-parameter-map-name
```

Example

```
Device(config-profile)# local-domain dns_wl
```

Configures local domain RegEx parameter map.

5. Configure the Anycast address.

Example

```
Device(config-profile)# resolver IPv6 10:1:1:1::10
```

The default address is applied when there is no specific address configured.

6. Return to privileged EXEC mode.

Example

```
Device(config-profile)# end
```

The Umbrella parameter map is now configured and ready for use in managing network security.

Enable or disable DNScrypt (GUI)

Learn how to enable or disable DNScrypt encryption for DNS packets using the graphical user interface (GUI). This procedure helps manage DNS security and filtering settings on your device

Use this task when you need to control whether DNS queries are encrypted, or to specify exceptions and update your Umbrella settings.

Configure DNScrypt to enhance or relax DNS traffic encryption as needed to adapt your organization's security requirements.

Before you begin

- Obtain the registration token from Cisco Umbrella.
- Determine domains to whitelist if needed.

Procedure

1. Choose **Configuration** > **Security** > **Threat Defence** > **Umbrella**.
2. Enter the **Registration Token** received from Umbrella. Alternatively, you can click on **Click here to get your Token** to get the token from Umbrella.
3. Enter the **Whitelist Domains** that you want to exclude from filtering.
4. Check or uncheck the **Enable DNS Packets Encryption** check box to encrypt or decrypt the DNS packets.
5. Click **Apply**.

The DNScrypt setting is updated. DNS packets is now be encrypted or unencrypted based on your selection, and any specified domains will be excluded from filtering.

Enable or disable DNScrypt

Learn how to enable or disable DNScrypt using commands.

To configure DNScrypt settings for enhanced DNS security.

This procedure is used to enable or disable DNScrypt, which provides encryption for DNS queries, enhancing privacy and security.

Before you begin

Ensure you have access to the device's global configuration mode.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create an umbrella global parameter map.

   Example

   ```
   Device(config)# parameter-map type umbrella global
   ```

3. Enable DNScrypt

   Example

   ```
   Device(config-profile)# no dnscrypt
   ```

   By default, the DNScrypt option is enabled.

> **Note**
>
> Cisco Umbrella DNScrypt is not supported when DNS-encrypted responses are sent in the data-DTLS encrypted tunnel (either mobility tunnel or AP CAPWAP tunnel).

4. Optional: Return to privileged EXEC mode.

Example

```
Device(config-profile)# end
```

The DNScrypt settings have been successfully configured.

Configure timeout for UDP sessions

Learn how to configure the timeout for UDP sessions using commands.

To set a timeout value for UDP sessions to manage session persistence and resource allocation.

This configuration is useful in environments where UDP sessions need to be managed effectively to prevent resource exhaustion.

Before you begin

Ensure you have access to the device and the necessary privileges to enter global configuration mode.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Create an umbrella global parameter map.

Example

```
Device(config)# parameter-map type umbrella global
```

3. Configure timeout value for UDP sessions.

Example

```
Device(config-profile)# udp-timeout timeout_value
```

The *timeout_value* ranges from 1 to 30 seconds.

> **Note**
>
> The **public-key** and **resolver** parameter-map options are automatically populated with the default values. So, you need not change them.

4. Optional: Return to privileged EXEC mode.

Example

```
Device(config-profile)# end
```

The UDP session timeout is configured successfully, allowing for better management of UDP resources.

Configure parameter Map Name in WLAN (GUI)

Configure Parameter Map Name in WLAN using GUI.

Set the parameter map name on a WLAN to apply specific security and policy settings.

Use this task when you need to assign or modify a parameter map for a WLAN on your wireless network device, ensuring the correct DNS and security parameters are active.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click on the Policy Profile Name. The **Edit Policy Profile** window is displayed.
3. Choose the **Advanced** tab.
4. In the **Umbrella** settings, from the **Umbrella Parameter Map** drop-down list, choose the parameter map.
5. Enable or disable **Flex DHCP Option for DNS** and **DNS Traffic Redirect** toggle buttons.
6. Click **Update & Apply to Device**.

The selected parameter map and options are applied to the WLAN Policy Profile. DNS and Umbrella security settings will operate based on the parameter map configuration.

Configure parameter map name in WLAN

Configure Parameter Map Name in WLAN using commands.

Configure a parameter map name to enable Cisco Umbrella OpenDNS integration on a WLAN.

This procedure is used when secure DNS filtering and policy enforcement are required on WLANs. The parameter map defines the Umbrella integration settings for the profile.

Before you begin

Ensure that the WLAN profile has been created and the network device is running the appropriate software version for Umbrella OpenDNS support.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. Create a policy profile for the WLAN.

   Example

   ```
   Device(config)# wireless profile policy policy-profile-name
   ```

   The *default-policy-profile* is the profile name of the policy profile.

3. Configure the Umbrella parameter map in the policy profile.

   Example

   ```
   Device(config-wireless-policy)# umbrella-param-map umbrella-name
   ```

   Configures the Cisco Umbrella OpenDNS feature for the WLAN using the specified parameter map.

   The *global* parameter map contains policy and key settings for Umbrella integration.

4. Exit wireless policy configuration and return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

   Alternatively, you can press Ctrl-Z.

The Umbrella parameter map name is now configured for the WLAN. DNS protection policies will be applied to wireless clients associating with this profile.

Configure the Umbrella Flex profile (CLI)

Configure the Umbrella Flex Profile using commands.

To configure the Umbrella Flex Profile for enhanced network management.

This procedure is used in environments where Umbrella services are integrated with wireless profiles to provide security and policy management.

Before you begin

Ensure you have the necessary permissions to configure the wireless profiles and Umbrella settings.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Create a new flex policy and enter flex profile configuration mode.

   Example

   ```
   Device(config)# wireless profile flex flex-profile-name
   ```

   The *flex-profile-name* is the flex profile name.

3. Configure the Umbrella flex feature.

   Example

   ```
   Device(config-wireless-flex-profile)# umbrella-profile umbrella-profile-name
   ```

   Use the **no** form of this command to negate the command or to set the command to its default.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-flex-profile)# end
   ```

   Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The Umbrella Flex Profile is now configured and ready for use in the network.

Configure the Umbrella Flex profile (GUI)

Use this task to configure the Umbrella Flex Profile through the graphical user interface to apply your desired parameter map settings to devices.

Configure Umbrella Flex Profile settings using the GUI.

Perform this task to set up or modify Umbrella Flex Profiles, applying the appropriate parameter maps to target devices.

Procedure

---

1. Choose **Configuration > Tags & Profiles > Flex** .
2. Click a **Flex Profile Name** . The **Edit Flex Profile** dialog box appears.
3. Under the **Umbrella** tab, click the **Add** button.
4. Select a name for the parameter map from the **Parameter Map Name** drop-down list and click **Save** .
5. Click the **Update & Apply to Device** button. The configuration changes are successfully applied.

The Umbrella Flex Profile is configured and the changes are successfully applied to your devices.

Configure Umbrella Flex parameters (CLI)

Configure Umbrella Flex Parameters using commands.

To configure Umbrella Flex parameters for WLAN policy profiles.

This procedure is used in environments where Umbrella services are integrated with WLAN policy profiles to manage DNS traffic effectively.

Before you begin

Ensure you have access to the device and the necessary permissions to configure WLAN policy profiles.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile and enter the wireless policy profile configuration mode.

   Example

   ```
   Device(config)# wireless profile policy profile-policy-name
   ```

   The *profile-policy-name* is the WLAN policy profile name.

3. Configure the Umbrella DHCP option for DNS.

   Example

   ```
   Device(config-wireless-policy-profile)# [no] flex umbrella dhcp-dns-option
   ```

   By default, the option is enabled.

4. Configure the DNS traffic redirection mode to Umbrella.

   Example

   ```
   Device(config-wireless-policy-profile)# [no] flex umbrella mode {force |
   ignore}
   ```

   You can either forcefully redirect the traffic or choose to ignore the redirected traffic to Umbrella. The default mode is **ignore**.

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

   Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The Umbrella Flex parameters are now configured for the WLAN policy profile.

Configure the Umbrella Flex policy profile (GUI)

CUse this task to configure an Umbrella Flex Policy Profile through the graphical user interface (GUI) to establish DNS redirection and related network policies for managed devices.

Define and deploy DNS redirection and other policy settings using Cisco Umbrella for your organization's network devices.

Perform this task whenever you want to enforce or update DNS-related policies and parameters for network devices using the GUI

Procedure

1. Choose **Configuration > Tags & Profiles > Policy** .
2. Click the **Add** button. The **Add Policy Profile** dialog box appears.
3. In the **Advanced** tab, and under the **Umbrella** section, complete the following:
   a) Select the parameter map from the **Umbrella Parameter Map** drop-down list. Click the **Clear** hyperlink to clear the selection.
   b) Click the field adjacent to **Flex DHCP Option for DNS** to **Disable** the option. By default it is **Enabled** .
   c) Click the field adjacent to **DNS Traffic Redirect** to set the option to **Force** . By default it is set to **Ignore** .
4. Click the **Apply to Device** button.

The specified Umbrella Flex Policy Profile is created and applied, enforcing your DNS and policy settings for selected devices.

Verify the Cisco Umbrella configuration

Verify the Cisco Umbrella Configuration.

To view the Umbrella configuration details, use the command:

```
Device# show umbrella config
Umbrella Configuration
=======================
Token: 5XXXXXXABXXXXXFXXXXXXXXXXDXXXXXXXXXXXXABXX
API-KEY: NONE
OrganizationID: xxxxxxx
Local Domain Regex parameter-map name: dns_bypass
DNSCrypt: Not enabled
Public-key: NONE
UDP Timeout: 5 seconds
Resolver address:
1. 10.1.1.1
2. 5.5.5.5
3. XXXX:120:50::50
4. XXXX:120:30::30
```

To view the device registration details, use the command:

```
Device# show umbrella deviceid
Device registration details
Param-Map Name                  Status          Device-id
global                          200 SUCCESS     010aa4eXXXXXXX8d
vj-1                            200 SUCCESS     01XXXXXXXf4541e1
GUEST                           200 SUCCESS     010a4f6XXXXXXX42
EMP                             200 SUCCESS     0XXXXXXXXd106ecd
```

To view the detailed description for the Umbrella device ID, use the command:

```
Device# show umbrella deviceid detailed
Device registration details
```

```
 1.global
      Tag               : global
      Device-id         : 010aa4eXXXXXXX8d
      Description       : Device Id recieved successfully
      WAN interface     : None
 2.vj-1
      Tag               : vj-1
      Device-id         : 01XXXXXXXf4541e1
      Description       : Device Id recieved successfully
      WAN interface     : None
```

To view the Umbrella DNSCrypt details, use the command:

```
Device# show umbrella dnscrypt
DNSCrypt: Enabled
   Public-key:
B111:XXXX:XXXX:XXXX:3E2B:XXXX:XXXX:XXXE:XXX3:3XXX:DXXX:XXXX:BXXX:XXXB:XXXX:FXXX

   Certificate Update Status: In Progress
```

To view the Umbrella global parameter map details, use the command:

```
Device# show parameter-map type umbrella global
```

To view the regex parameter map details, use the command:

```
Device# show parameter-map type regex <parameter-map-name>
```

To view the Umbrella statistical information, use the command:

```
Device# show platform hardware chassis active qfp feature umbrella datapath
stats
```

To view the wireless policy profile Umbrella configuration, use the command:

```
Device#show wireless profile policy detailed vj-pol-profile | s Umbrella
Umbrella information
Cisco Umbrella Parameter Map : vj-2
DHCP DNS Option : ENABLED
Mode : force
```

To view the wireless flex profile Umbrella configuration, use the following command:

```
Device#show wireless profile flex detailed vj-flex-profile | s Umbrella
Umbrella Profiles :
vj-1
vj-2
global
```

To view the Umbrella details on the AP, use the following command:

```
AP#show client opendns summary
Server-IP role
208.67.220.220 Primary
208.67.222.222 Secondary

Server-IP role
2620:119:53::53 Primary
2620:119:35::35 Secondary
```

```
Wlan Id DHCP OpenDNS Override Force Mode
0 true false
1 false false
...

15 false false
Profile-name Profile-id
vj-1 010a29b176b34108
global 010a57bf502c85d4
vj-2 010ae385ce6c1256
AP0010.10A7.1000#

Client to profile command

AP#show client opendns address 50:3e:aa:ce:50:17
Client-mac Profile-name
50:3E:AA:CE:50:17 vj-1
AP0010.10A7.1000#
```

135 RADIUS Server Load Balancing

Topics:

- RADIUS server load balancing

Introduces RADIUS server load balancing concepts and explains how to distribute authentication requests across multiple servers for improved performance and redundancy.

RADIUS server load balancing

Explains RADIUS server load balancing, its prerequisites and restrictions, and provides instructions for enabling load balancing on named RADIUS server groups.

RADIUS server load balancing is a transaction distribution mechanism that

- distributes batches of AAA transactions across multiple RADIUS servers in a server group
- assigns each batch to the server with the lowest number of outstanding transactions in its queue, and
- ensures active utilization of all servers rather than using standby configuration.

Load balancing operation

By default, if two RADIUS servers are configured in a server group, only one is used. The other server acts as standby, if the primary server is declared as dead, the secondary server receives all the load.

If you need both servers to perform transactions actively, you need to enable Load Balancing.

> **Note**
>
> By default, load balancing is not enabled on the RADIUS server group.

If you enable load balancing in a RADIUS server group with two or more RADIUS servers, the Server A and Server B receives a AAA transaction. The transaction queues are checked in Server A and Server B. The server with less number of outstanding transactions are assigned the next batch of AAA transaction.

The process of assigning a batch of transactions is as follows:

1. The first transaction is received for a new batch.
2. All server transaction queues are checked.
3. The server with the lowest number of outstanding transactions is identified.
4. The identified server is assigned the next batch of transactions.

The batch size is a user-configured parameter. Changes in the batch size may impact CPU load and network throughput. As batch size increases, CPU load decreases, and network throughput increases. However, if a large batch size is used, all available server resources may not be fully utilized. As batch size decreases, CPU load increases and network throughput decreases.

> **Note**
>
> There is no set number for large or small batch sizes. A batch with more than 50 transactions is considered large and a batch with fewer than 25 transactions is considered small.

> **Note**
>
> If a server group contains ten or more servers, we recommend that you set a high batch size to reduce CPU load.

You can configure load balancing either per-named RADIUS server group or for the global RADIUS server group. The load balancing server group must be referred to as "radius" in the authentication, authorization, and accounting (AAA) method lists. All public servers that are part of the RADIUS server group are then load balanced.

You can configure authentication and accounting to use the same RADIUS server or different servers. In some cases, the same server can be used for preauthentication, authentication, or accounting transactions for a session. The preferred server, which is an internal setting and is set as the default, informs AAA to use the same server for the start and stop record for a session regardless of the server cost. When using the preferred server setting, ensure that the server that is used for the initial transaction (for example, authentication), the preferred server, is part of any other server group that is used for a subsequent transaction (for example, accounting).

The preferred server is not used if one of the following criteria is true:

- The **load-balance method least-outstanding ignore-preferred-server** command is used.
- The preferred server is dead.
- The preferred server is in quarantine.
- The want server flag has been set, overriding the preferred server setting.

The want server flag, an internal setting, is used when the same server must be used for all stages of a multistage transaction regardless of the server cost. If the want server is not available, the transaction fails.

You can use the **load-balance method least-outstanding ignore-preferred-server** command if you have either of the following configurations:

- Dedicated authentication server and a separate dedicated accounting server
- Network where you can track all call record statistics and call record details, including start and stop records and records that are stored on separate servers
- The **least outstanding** load-balancing method in RADIUS is only recommended while doing 802.1X without accounting.

If you have a configuration where authentication servers are a superset of accounting servers, the preferred server is not used.

Note

If a third-party RADIUS load balancer is used and RADIUS packets are routed based on the NAS source port, it is recommended to move to any other rule based on the following Attribute-Value Pairs (AVPs):

- If the load balancer uses NAS source port in the Access-Request to load balance, rules may not work as expected as the source port in NAS might change during transaction.
- If the load balancer compares AVPs between Access-Challenge and Access-Request to route packets, you will need to use the AVP value of t-State.
- If the load balancer compares AVPs in Access-Request from NAS, you will need to use one or a combination of the following AVPs:
  - t-State value
  - Calling-Station-ID and NAS IP or Identifier

Prerequisites for RADIUS server load balancing

Lists the prerequisites for RADIUS Server Load Balancing.

This reference lists the prerequisites you must meet before configuring RADIUS Server Load Balancing.

- Configure Authentication, Authorization, and Accounting (AAA) on the RADIUS server.
- Configure AAA RADIUS server groups.
- Configure RADIUS for functions such as authentication, accounting, or static route download.

RADIUS server load balancing restrictions

Learn about RADIUS Server Load Balancing restrictions that affect your implementation.

Use this reference to understand RADIUS Server Load Balancing restrictions and limitations before implementing this feature.

- Incoming RADIUS requests, such as Packet of Disconnect (POD) requests are not supported.
- Load balancing is not supported on proxy RADIUS servers and private server groups.
- Central Web Authentication (CWA) does not support load balancing. CWA and CoA are linked to accounting. With load balancing enabled, the system may send accounting data to a different server than the one processing the CoA request.
- RADIUS Load Balancing does not support client stickiness during the same session. Do not use it for authentication mechanisms like CWA or CoA. It may impact server clusters when client session ownership changes across servers.
- The **Least-outstanding** load-balancing method is not recommended for RADIUS in CWA or NAC scenarios that require CoA and multiple RADIUS transactions. When implementing 802.1X with accounting, this method may lead to inconsistencies, as the system does not send all accounting packets to the same server, potentially impacting the ISE.

Enable load balancing for a named RADIUS server group (CLI)

Learn how to enable load balancing for a named Remote Authentication Dial-In User Service (RADIUS) server group to distribute authentication and accounting requests across multiple servers.

Enable load balancing to distribute authentication and accounting requests across multiple RADIUS servers in a named server group for improved performance and reliability.

Load balancing for RADIUS server groups helps distribute the authentication load across multiple servers, reducing the risk of server overload and improving response times. This configuration is particularly useful in high-traffic environments where multiple RADIUS servers are available.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter server group configuration mode.

   Example

   ```
   Device(config)# aaa group server radius group-name
   ```

   Example

   ```
   Device(config)# aaa group server radius rad-sg
   ```

3. Configure the IP address of the RADIUS server for the group server.

   Example

   ```
   Device(config-sg-radius)# server ip-address auth-port port-number acct-port
   port-number
   ```

   Example

   ```
   Device(config-sg-radius)# server 192.0.2.238 auth-port 2095 acct-port 2096
   ```

4. Enable the least-outstanding load balancing for a named server group.

   Example

   ```
   Device(config-sg-radius)# load-balance method least-outstanding batch-size
   number
   ```

   Example

   ```
   Device(config-sg-radius)# load-balance method least-outstanding batch-size
   30
   ```

5. Exit server group configuration mode and enter privileged EXEC mode.

Example

```
Device(config-sg)# end
```

Load balancing is now enabled for the named RADIUS server group, allowing authentication and accounting requests to be distributed across multiple RADIUS servers based on the least-outstanding method.

136 AAA Dead-Server Detection

Topics:

- AAA dead-server detection

AAA dead-server detection

Introduces AAA dead-server detection concepts, covering an overview of the feature along with prerequisites and operational restrictions for deploying dead-server detection in network environments.

A dead-server detection feature is an AAA capability that

- allows configuration of criteria for marking RADIUS servers as dead
- uses tunable deadtime and dead-criteria values to determine server status, and
- provides faster failover and packet processing by quickly detecting and isolating non-responding servers.

You can configure the deadtime (in minutes) per server group or globally. This defines how long a server marked as DEAD remains in that state.

Deadtime—The period (in minutes) that a server, once marked as DEAD, remains in that state before being re-evaluated. This value can be set per server group or globally.

Dead-criteria—The specific conditions, such as timeout intervals or number of missed responses, that must be met before a server is considered DEAD. To declare a server as DEAD, you need to configure **dead-criteria** and configure the conditions that determine when a RADIUS server is considered unavailable or dead.

Criteria for marking a RADIUS server as dead

AAA dead-server detection uses two main criteria for marking a RADIUS server as unavailable: the minimum time since the last valid response, and the number of consecutive timeouts. When both criteria are met, the server is marked as dead and remains in that state for the configured dead time. After dead time expires, the controller re-evaluates the server's state and updates clients accordingly.

This approach ensures that only unresponsive servers are marked as dead. It minimizes authentication delays and prevents unnecessary switches between dead and alive status due to slow responses.

For configuration details and packet handling rules, see the definitions above.

Additional reference information

- Both time and tries criteria must be met for a server to be marked as dead.
- The detection process helps avoid unnecessary state changes between dead and alive for temporarily slow servers.
- Both authentication and accounting packets are counted when applying the dead-criteria.

Example of AAA dead-server detection

If a controller has multiple RADIUS servers and one stops responding for the configured dead time and number of consecutive timeouts, it is marked as dead. Requests are then redirected to working servers, improving reliability.

Counter-example: Without AAA dead-server detection

Without AAA dead-server detection, a controller may continually attempt to communicate with a non-responsive RADIUS server, increasing authentication delays and denying access to users.

Prerequisites for AAA dead-server detection

AAA Dead-Server Detection prerequisites.

- You must have access to a RADIUS server.
- You must be familiar with configuring a RADIUS server.

- You must be familiar with configuring Authentication, Authorization, and Accounting (AAA).
- Before a server can be marked as dead, yo configure **radius-server dead-criteria time** *time-in-seconds* **tries** *number-of-tries* to mark the server as DOWN.

  Also, you must configure the **radius-server deadtime** *time-in-mins* to retain the server in DEAD status.

Restrictions for AAA dead-server detection

AAA Dead-Server Detection restrictions.

Original transmissions are not counted in the number of consecutive timeouts that must occur on the controller before the server is marked as dead--only the number of retransmissions are counted.

Configure AAA dead-server detection (CLI)

Provides instructions for configuring AAA dead-server detection using CLI, including steps to set up, manage, and verify dead-server detection to ensure effective monitoring and recovery of unresponsive AAA servers.

Enable detection of dead RADIUS servers to maintain network reliability.

This configuration is essential in environments using RADIUS servers for authentication. It ensures you can effectively manage server availability.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable the AAA access control model.

   Example

   ```
   Device(config)# aaa new-model
   ```

3. Define how long, in minutes, a server marked as DEAD will remain in that state.

   Example

   ```
   Device(config)# radius-server deadtime time-in-mins
   ```

   Example

   ```
   Device(config)# radius-server deadtime 5
   ```

   When the deadtime expires, the controller marks the server as UP (ALIVE) and notifies registered clients about the state change.

   time-in-mins – Valid values range from 1 to 1440 minutes. Default value is zero. To return to the default value, use the **no radius-server deadtime** command.

   The **radius-server deadtime** command can be configured globally or per AAA group server level.

You can use the **show aaa dead-criteria** or **show aaa servers** command to check for dead-server detection. If the default value is zero, deadtime is not configured.

4. Declare a server as DEAD and configure the conditions that determine when a RADIUS server is considered unavailable or dead.

   Example

   ```
   Device(config)# radius-server dead-criteria [time time-in-seconds] [tries number-of-tries]
   ```

   Example

   ```
   Device(config)# radius-server dead-criteria time 5 tries 4
   ```

   time-in-seconds : Time in seconds during which no response is received from the RADIUS server to consider it as dead. Valid values range from 1 to 120 seconds.

   > **Note**
   >
   > The total time to declare the server as dead depends on the configured time in seconds, retries, and pending transactions.
   >
   > The calculated resulting time may range from ten to sixty seconds based on these criteria.

   number-of-tries : Number of transmits to RADIUS server without responses before marking the server as dead. Valid values range from 1 to 100.

5. Exit configuration mode and enter privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

You have now configured the AAA Dead-Server Detection, allowing you to manage RADIUS server availability effectively.

Verify AAA dead-server detection

This section lists the commands to verify AAA dead-server detection, configuration, and status across devices.

This topic describes how to verify AAA Dead-Server Detection using various show commands.

To verify dead-criteria, use this command:

```
Device# show run | s dead-criteria
radius-server dead-criteria time 20 tries 20
```

To verify the dead-criteria details, use this command:

```
Device# sh aaa dead-criteria radius <server>
sh aaa dead-criteria radius 192.0.2.55
RADIUS Server Dead Criteria:
```

```
Server Details:
Address : 192.0.2.55
Auth Port : 1645
Acct Port : 1646
Server Group : radius
Dead Criteria Details:
Configured Retransmits : 3
Configured Timeout : 5
Estimated Outstanding Access Transactions: 2
Estimated Outstanding Accounting Transactions: 0
Dead Detect Time : 30s
Computed Retransmit Tries: 6
Statistics Gathered Since Last Successful Transaction
Max Computed Outstanding Transactions: 3
Max Computed Dead Detect Time: 90s
Max Computed Retransmits : 18
```

To verify the state of servers, number of requests being processed, and so on, use the command:

```
Device# show aaa servers | s WNCD
Platform State from WNCD (1) : current UP
Platform State from WNCD (2) : current UP
Platform State from WNCD (3) : current UP
Platform State from WNCD (4) : current UP
Platform State from WNCD (5) : current UP, duration 773s, previous duration
0s
Platform Dead: total time 0s, count 0
Quarantined: No
```

137 ISE Simplification and Enhancements

Topics:

- Security utilities configuration options
- Captive bypassing
- DHCP option 55 and 77 parameters
- Captive portals

This chapter provides step-by-step configurations to simplify RADIUS/AAA setup, enable captive portal and bypass behaviors, configure DHCP options for ISE profiling, and tune EAP timeouts to enforce secure wireless authentication.

Security utilities configuration options

Introduces security configuration utilities and provides instructions for configuring multiple RADIUS servers with procedures to verify AAA and RADIUS server configurations.

You can set up the RADIUS server using this simplified method:

wireless-default radius server *ip* **key** *secret*

- Configures AAA authorization for network services, including authentication for web authentication and Dot1x.
- Enables local authentication with default authorization.
- Applies the default redirect ACL for Central Web Authentication (CWA).
- Creates a global parameter map with a virtual IP and enables captive bypass portal.
- Configures all default AAA settings required when configuring a RADIUS server.
- Sets method-list configuration and enables RADIUS accounting by default.
- Disables RADIUS aggressive failovers by default and sets the RADIUS request timeout to five seconds.
- Implements a default access control list (ACL) for CWA URL redirect.

```
aaa new-model
                aaa authentication webauth default group radius
                aaa authentication dot1x default group radius
                aaa authorization network default group radius
                aaa accounting identity default start-stop group radius
                !
                aaa server radius dynamic-author
                client <IP> server-key cisco123
                !
                radius server RAD_SRV_DEF_<IP>
                description Configured by wireless-default
                address ipv4 <IP> auth-port 1812 acct-port 1813
                key <key>
                !
                aaa local authentication default authorization default
                aaa session-id common
                !
                ip access-list extended CISCO-CWA-URL-REDIRECT-ACL-DEFAULT
                remark " CWA ACL to be referenced from ISE "
                deny udp any any eq domain
                deny tcp any any eq domain
                deny udp any eq bootps any
                deny udp any any eq bootpc
                deny udp any eq bootpc any
                deny ip any host <IP>
                permit tcp any any eq www
                !
                parameter-map type webauth global
                captive-bypass-portal
                virtual-ip ipv4 192.0.2.1
                virtual-ip ipv6 2001:DB8::1
                !
                wireless profile policy default-policy-profile
                aaa-override
                local-http-profiling
                local-dhcp-profiling
```

```
                accounting
```

Key benefits

- Simplifies deployment of RADIUS server configuration for wireless controllers.
- Eliminates the need for step-by-step manual configuration across multiple CLI commands.
- Ensures consistent and secure default settings for AAA and authentication services.

Configure multiple RADIUS servers

Learn how to configure multiple radius servers using commands.

To configure multiple RADIUS servers for authentication in a network.

Before you begin

Ensure you have the necessary IP addresses and shared secrets for the RADIUS servers.

Procedure

1. Enter global configuration mode

   Example

   ```
   Device# configure terminal
   ```

2. Configure a RADIUS server

   Example

   ```
   Device(config)# wireless-default radius server ip key secret
   ```

   Example

   ```
   Device(config)# wireless-default radius server 192.2.58.90 key cisco123
   ```

   You can configure up to ten RADIUS servers.

3. Return to privileged EXEC mode

   Example

   ```
   Device(config)# end
   ```

   Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Multiple RADIUS servers are now configured and ready for use in the network.

Verify AAA and Radius server configurations

Verify AAA and Radius Server Configurations.

To view details of AAA server, use these command:

```
Device# show run aaa
!
aaa new-model
aaa authentication webauth default group radius
aaa authentication dot1x default group radius
aaa authorization network default group radius
aaa accounting Identity default start-stop group radius
!
aaa server radius dynamic-author
 client 192.0.2.10 server-key cisco123
!
radius server RAD_SRV_DEF_192.0.2.10
 description Configured by wireless-default
 address ipv4 192.0.2.10 auth-port 1812 acct-port 1813
 key cisco123
!
aaa local authentication default authorization default
aaa session-id common
!
!
ip access-list extended CISCO-CWA-URL-REDIRECT-ACL-DEFAULT
remark " CWA ACL to be referenced from ISE "
deny udp any any eq domain
deny tcp any any eq domain
deny udp any eq bootps any
deny udp any any eq bootpc
deny udp any eq bootpc any
deny ip any host 192.0.2.10
permit tcp any any eq www
!
parameter-map type webauth global
  captive-bypass-portal
  virtual-ip ipv4 192.0.2.1
  virtual-ip ipv6 2001:DB8::1
!
wireless profile policy default-policy-profile
   aaa-override
   local-http-profiling
   local-dhcp-profiling
   accounting
```

Note

The **show run aaa** output may change when new commands are added to this utility.

Captive bypassing

Configuring Captive Bypassing for WLAN in LWA and CWA (GUI), Configuring Captive Bypassing for WLAN in LWA and CWA

A captive bypass is a network behavior that

- enables client devices to automatically determine Internet connectivity and initiate authentication flows,
- may enable users to bypass captive portal splash pages through protocols such as WISPr.

- involves interaction between devices (such as Apple iOS) and Controllers, affecting web redirection and authentication.

Additional reference information

WISPr is a draft protocol that enables users to roam between different wireless service providers. Devices like Apple iOS use HTTP WISPr requests to check if they are connected to the Internet.

When direct Internet connectivity is not detected, these mechanisms prompt the device to open a web browser so the user can enter credentials for Internet access. Authentication occurs in the background whenever the device connects to a new SSID.

When an Apple iOS device connects, it sends a WISPr request to the Controller, which checks the user agent details and then triggers an HTTP request with web authentication interception. After verifying the iOS version and browser details, the Controller allows the client device to bypass captive portal settings and access the Internet.

This HTTP request triggers web authentication interception on the Controller just as any other page requests do. This leads to the standard web authentication process. If web authentication uses splash page features (such as a URL from a configured RADIUS server), the splash page might not appear for the user because WISPr requests occur at short intervals. Once a query reaches the server, web redirection or splash page display running in the background is cancelled, and the device processes the page request, which may prevent the splash page from being displayed as intended.

Example

For instance, Apple introduced an iOS feature that detects captive portals by sending a web request upon connecting to a wireless network. On iOS 6 and earlier, the request is sent to http://www.apple.com/library/test/success.html. On iOS 7 and later, multiple possible URLs are used. If a response is received, Internet access is assumed and no further interaction occurs. If no response is received, Internet access is assumed to be blocked and the device's Captive Network Assistant (CNA) auto-launches a pseudo-browser to prompt for portal login. The CNA may not function properly if it is redirected to an ISE captive portal. The Controller can suppress the pop-up of the pseudo-browser. Network administrators can configure the Controller to bypass WISPr detection. In this configuration, web authentication interception only occurs when a user intentionally requests a web page, ensuring the splash page is loaded in the user's browser context and not triggered by background connectivity probes.

Analogy: Sneaking into an amusement park

Captive bypassing is like someone sneaking into an amusement park through a side door, instead of showing their ticket at the main entrance. Some devices use automated checks to find shortcuts and access the internet without going through the intended login page, just as clever visitors might bypass security by finding an easier way in. Network controls are needed to make sure everyone enters through the main gate and checks in properly.

Configure captive bypassing for WLAN in LWA and CWA (GUI)

Use this task to enable captive bypass on a WLAN for local web authentication (LWA) or central web authentication (CWA) using the GUI.

Enable client devices to bypass captive portal authentication on the selected WLAN.

Before you begin

Use this task when you need certain clients to connect without passing through the captive portal, typically for pre-authorized devices.

Ensure you have administrative access to the device and the parameter map is appropriately configured.

Procedure

1. Choose **Configuration** > **Security** > **Web Auth** .
2. In the **Webauth Parameter Map** tab, click the parameter map name. The **Edit WebAuth Parameter** window is displayed.

3. Select **Captive Bypass Portal** check box.
4. Click **Update & Apply to Device** .

The selected WLAN now allows captive bypass in accordance with your configuration.

Configure captive bypassing for WLAN in LWA and CWA

Use this procedure to configure captive bypassing for WLAN in LWA and CWA using commands.

Enable captive bypassing for WLAN to allow users to bypass the captive portal.

This configuration is applicable in environments where users need to access the network without going through a captive portal.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create the parameter map.

   Example

   ```
   Device(config)# parameter-map type webauth parameter-map-name
   ```

   The *parameter-map-name* must not exceed 99 characters.

3. Configure captive bypassing.

   Example

   ```
   Device(config)# captive-bypass-portal
   ```

4. Specify the WLAN name and ID.

   Example

   ```
   Device(config)# wlan profile-name wlan-id ssid-name
   ```

   - *profile-name* is the WLAN name which can contain 32 alphanumeric characters.
   - *wlan-id* is the wireless LAN identifier. The valid range is from 1 to 512.
   - *ssid-name* is the SSID which can contain 32 alphanumeric characters.

5. Enable the web authentication for the WLAN.

   Example

   ```
   Device(config-wlan)# security web-auth
   ```

6. Map the parameter map.

   Example

   ```
   Device(config-wlan)# security web-auth parameter-map parameter-map-name
   ```

   If the parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.

7. Optional: Return to privileged EXEC mode.

   Example

   ```
   Device(config-wlan)# end
   ```

   Alternatively, you can also press *Ctrl-Z* to exit global configuration mode.

The WLAN is now configured with captive bypassing enabled, allowing users to bypass the captive portal.

DHCP option 55 and 77 parameters

Describes DHCP options 55 and 77 functionality and provides configuration procedures for sending these options to ISE, including EAP request timeout settings via GUI and CLI.

The DHCP sensors use these DHCP options on the ISE for both native and remote device profiling:

- Option 12: Hostname
- Option 6: Class Identifier

These options must also be sent to ISE for accurate profiling:

- Option 55: Parameter Request List
- Option 77: User Class

Configure the DHCP options 55 and 77 to ISE using the GUI

Use this task to configure DHCP options 55 and 77 for integration with Cisco Identity Services Engine (ISE) using the graphical interface.

Ensure DHCP options 55 and 77 are sent to ISE to enable device profiling and policy enforcement

These options allow ISE to collect device-specific DHCP information for accurate profiling on wireless networks.

Before you begin

Confirm that your network devices are compatible with ISE profiling features.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. On the **Policy Profile** page, click **Add** to view the **Add Policy Profile** window.
3. Click **Access Policies** tab, choose the **RADIUS Profiling** and **DHCP TLV Caching** check boxes to configure radius profiling and DHCP TLV Caching on a WLAN.

4. Click **Save & Apply to Device**

---

he network device now sends DHCP options 55 and 77 to ISE, enabling accurate device profiling and network access policy enforcement.

Configure the device to send DHCP options 55 and 77 to ISE (CLI)

Use this procedure to configure your device using CLI commands to send DHCP options 55 and 77 to Cisco Identity Services Engine (ISE) for effective client profiling.

Enable the device to send specific DHCP options to ISE for improved client profiling.

This configuration is essential for environments where client profiling is necessary for network access control and monitoring.

Before you begin

Ensure that the device is running a compatible version of the software that supports DHCP options 55 and 77.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enters the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy profile-policy
   ```

   The *profile-policy* is the name of the WLAN policy profile being configured.

3. Configure DHCP TLV caching on a WLAN.

   Example

   ```
   Device(config-wireless-policy)# dhcp-tlv-caching
   ```

   Enables caching of DHCP TLV information for better profiling.

4. Configure client radius profiling on a WLAN.

   Example

   ```
   Device(config-wireless-policy)# radius-profiling
   ```

   Enables RADIUS profiling to enhance client identification.

5. Optional: Return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

Alternatively, you can also press *Ctrl-Z* to exit global configuration mode.

The device is configured to include DHCP options 55 and 77 in client profiling information sent to Cisco ISE.

Configure the EAP request timeout using the GUI

Configure EAP Request Timeout using GUI.

Use the steps to configure the EAP request timeout through the GUI:

Procedure

---

1. Choose **Configuration** > **Security** > **Advanced EAP**.
2. In the **EAP-Identity-Request Timeout** field, specify the amount of time (in seconds) in which the device attempts to send an EAP identity request to wireless clients using local EAP.
3. In the **EAP-Identity-Request Max Retries** field, specify the maximum number of times that the device attempts to retransmit the EAP identity request to wireless clients using local EAP.
4. Set **EAP Max-Login Ignore Identity Response** to **Enabled** state to limit the number of clients that can be connected to the device with the same username. You can log in up to eight times from different clients (PDA, laptop, IP phone, and so on) on the same device. The default state is **Disabled**.
5. In the **EAP-Request Timeout** field, specify the amount of time (in seconds) in which the device attempts to send an EAP request to wireless clients using local EAP.
6. In the **EAP-Request Max Retries** field, specify the maximum number of times that the device attempts to retransmit the EAP request to wireless clients using local EAP.
7. In the **EAPOL-Key Timeout** field, specify the amount of time (in seconds) in which the device attempts to send an EAP key over the LAN to wireless clients using local EAP.
8. In the **EAPOL-Key Max Retries** field, specify the maximum number of times that the device attempts to send an EAP key over the LAN to wireless clients using local EAP.
9. In the **EAP-Broadcast Key Interval** field, specify the time interval between rotations of the broadcast encryption key used for clients and click **Apply**.

> **Note**
>
> After configuring the EAP-Broadcast key interval to a new time period, you must shut down or restart the WLAN for the changes to take effect. Once the WLAN is shut down or restarted, the M5 and M6 packets are exchanged when the configured timer value expires.

Configure EAP request timeout

Use this procedure to configure EAP request timeout using commands.

Set the timeout for EAP requests to manage client exclusion effectively.

This configuration is useful in environments where EAP authentication is used, and it is necessary to handle clients that do not respond in a timely manner.

Before you begin

Ensure you have access to the device and are in the appropriate configuration mode.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable exclusion on timeout and no response.

   Example

   ```
   Device(config)# wireless wps client-exclusion dot1x-timeout
   ```

   By default, this feature is enabled. To disable, append a *no* at the beginning of the command.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   Alternatively, you can also press *Ctrl-Z* to exit global configuration mode.

The EAP request timeout is now configured, allowing for better management of client exclusions.

Configure the EAP request timeout in wireless security

Use this procedure to configure EAP request timeout in wireless security using commands.

Set the EAP request timeout to ensure timely authentication in wireless security.

This configuration is essential in environments where EAP is used for authentication, as it affects the responsiveness of the authentication process.

Before you begin

Ensure you have access to the device and are in privileged EXEC mode.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. Configure the EAP request retransmission timeout value in seconds.

   Example

   ```
   Device(config)# wireless security dot1x request timeout timeout
   ```

Configures the EAP request retransmission timeout value in seconds.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The EAP request timeout is now configured, which will help in managing the authentication process in wireless security.

Captive portals

Outlines captive portal implementation concepts and provides comprehensive configuration procedures through GUI and CLI, with verification commands to confirm portal functionality.

A captive portal is a network authentication mechanism that

- intercepts and redirects user web requests to a designated login or acceptance page,
- allows administrators to enforce authentication or policy acceptance before granting network access, and
- supports configuration of multiple web authentication URLs based on access point (AP), WLAN, or global scope.

Order of precedence

Captive portal configurations let you define different web authentication URLs—including external URLs—for the same SSID, based on the specific access point or WLAN. By default, the system uses a global URL for user authentication, but administrators may override this at the WLAN or AP level to customize user experience.

Order of precedence

1. AP-level configuration (highest priority)
2. WLAN-level configuration
3. Global configuration (default)

Restrictions

- This feature is supported on standalone controllers only.
- Export-Anchor configurations are not supported.

Configure captive portal (GUI)

Configure Captive Portal using GUI.

Configure a secure WLAN that uses captive portal authentication for guest or user access.

Configure a captive portal when you require user authentication on your Wi-Fi network for guests or employees.

Before you begin

Confirm the web authentication parameter map and necessary authentication lists are already created.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**, the **SSID**, and the **WLAN ID**.
4. In the **Security** > **Layer2** tab, uncheck the **WPA Policy**, **AES** and **802.1x** check boxes.
5. In the **Security** > **Layer3** tab, choose the parameter map from the **Web Auth Parameter Map** drop-down list and authentication list from the **Authentication List** drop-down list.
6. In the **Security** > **AAA** tab, choose the Authentication list from the **Authentication List** drop-down list.
7. Click **Apply to Device**.
8. Choose **Configuration** > **Security** > **Web Auth**.
9. Choose a **Web Auth Parameter Map**.
10. In the **General** tab, enter the **Maximum HTTP connections**, **Init-State Timeout(secs)** and choose **webauth** from the **Type** drop-down list.
11. In the **Advanced** tab, under the **Redirect to external server** settings, enter the **Redirect for log-in** server.
12. Click **Update & Apply**.

The captive portal WLAN is available, and users associating with this SSID will be redirected for authentication as defined in the parameter map.

Configure captive portal

Use this procedure to configure captive portal using commands.

Set up a captive portal to manage user access to the network.

The captive portal is used to authenticate users before granting them access to the network, typically in public Wi-Fi environments.

Before you begin

Ensure that the WLAN is properly configured and that you have the necessary access rights.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. Configure the WLAN profile.

   Example

   ```
   Device(config)# wlan profile-name
   ```

   Enables or disables all WLANs and creates the WLAN identifier. The *profile-name* and the SSID network name should be up to 32 alphanumeric characters.

3. Configure the WLAN web ACL.

Example

```
Device(config-wlan)# ip access-group web IPv4-ACL-Name
```

WLAN needs to be disabled before performing this operation.

4. Disable WPA security.

Example

```
Device(config-wlan)# no security wpa
```

5. Disable security AKM for dot1x.

Example

```
Device(config-wlan)# no security wpa akm dot1x
```

Disables security AKM for dot1x.

6. Disable WPA2 ciphers for AES.

Example

```
Device(config-wlan)# no security wpa wpa2 ciphers aes
```

Disables WPA2 ciphers for AES.

7. Enable web authentication for WLAN.

Example

```
Device(config-wlan)# security web-auth authentication-list
authentication-list-name
```

```
Device(config-wlan)# security web-auth parameter-map parameter-map-name
```

Enables web authentication for WLAN. Here,

- **authentication-list**

  authentication-list-name : Sets the authentication list for IEEE 802.1x.

- **authorization-list**

  authorization-list-name : Sets the override-authorization list for IEEE 802.1x.

- **on-macfilter-failure** : Enables Web authentication on MAC filter failure.

- **parameter-map**

  parameter-map-name : Configures the parameter map.

> **Note**
>
> When **security web-auth** is enabled, you get to map the default **authentication-list** and global **parameter-map** . This is applicable for authentication-list and parameter-map that are not explicitly mentioned.

8. Optional: Enable the WLAN.

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

9. Exit from the WLAN configuration.

   Example

   ```
   Device(config-wlan)# exit
   ```

10. Create a parameter map and enter parameter-map webauth configuration mode.

    Example

    ```
    Device(config)# parameter-map type webauth parameter-map-name
    ```

    Creates a parameter map and enters parameter-map webauth configuration mode.

11. Configure the webauth type parameter.

    Example

    ```
    Device(config-params-parameter-map)# type webauth parameter-map-name
    ```

12. Configure the WEBAUTH timeout in seconds.

    Example

    ```
    Device(config-params-parameter-map)# timeout init-state sec <timeout-seconds>
    ```

    Valid range for the time in sec parameter is 60 seconds to 3932100 seconds.

13. Configure the URL string for redirect during login.

    Example

    ```
    Device(config-params-parameter-map)# redirect for-login <URL-String>
    ```

    Example

    ```
    Device(config-params-parameter-map)# redirect for-login
    https://172.16.100.157/portal/login.html
    ```

    Configures the URL string for redirect during login.

14. Exit the parameters configuration.

Example

```
Device(config-params-parameter-map)# exit
```

15. Configure policy tag and enter policy tag configuration mode.

Example

```
Device(config)# wireless tag policy policy-tag-name
```

16. Attach a policy profile to a WLAN profile.

Example

```
Device(config-policy-tag)# wlan wlan-profile-name policy policy-profile-name
```

17. Save the configuration and exit configuration mode.

Example

```
Device(config-policy-tag)# end
```

The captive portal is now configured and ready to manage user access to the network.

Example: Captive portal configuration

Captive Portal Configuration example.

The example shows how you can have APs at different locations, broadcasting the same SSID but redirecting clients to different redirect portals:

Configuring multiple parameter maps pointing to different redirect portal:

```
parameter-map type webauth parMap1
type webauth
timeout init-state sec 21600
redirect for-login
https://172.16.12.3:8080/portal/PortalSetup.action?portal=cfdbce00-2ce2-11e8-b83c-005056a06b27
redirect portal ipv4 172.16.12.3
!
!
parameter-map type webauth parMap11
type webauth
timeout init-state sec 21600
redirect for-login
https://172.16.12.4:8443/portal/PortalSetup.action?portal=094e7270-3808-11e8-9797-02421e4cae0c
redirect portal ipv4 172.16.12.4
!
```

Associating these parameter maps to different WLANs:

```
wlan edc1 1 edc
ip access-group web CPWebauth
```

```
no security wpa
no security wpa akm dot1x
no security wpa wpa2 ciphers aes
security web-auth
security web-auth authentication-list cp-webauth
security web-auth parameter-map parMap11
no shutdown
wlan edc2 2 edc
ip access-group web CPWebauth
no security wpa
no security wpa akm dot1x
no security wpa wpa2 ciphers aes
security web-auth
security web-auth authentication-list cp-webauth
security web-auth parameter-map parMap1
no shutdown
```

Note

All WLANs have identical SSIDs.

Associating WLANs to different policy tags:

```
wireless tag policy policy_tag_edc1
wlan edc1 policy policy_profile_flex
wireless tag policy policy_tag_edc2
wlan edc2 policy policy_profile_flex
```

Assigning these policy tags to the desired APs:

```
ap E4AA.5D13.14DC
policy-tag policy_tag_edc1
site-tag site_tag_flex
ap E4AA.5D2C.3CAC
policy-tag policy_tag_edc2
site-tag site_tag_flex
```

138 RADIUS DTLS

Topics:

- RADIUS DTLS

RADIUS protocol provides centralized security for users attempting to gain management access to a network.

RADIUS DTLS

Describes RADIUS DTLS as a client or server protocol that provides centralized security for users attempting to gain management access to a network.

RADIUS DTLS is a client or server protocol that

- provides centralized security for users attempting to gain management access to a network
- delivers a complete Authentication, Authorization, and Accounting (AAA) solution, and
- uses RADIUS-over-DTLS (UDP) for secure communication.

RADIUS DTLS components and configuration

The Remote Authentication Dial-In User Service (RADIUS) is a widely deployed authentication and authorization protocol.

RADIUS DTLS includes these components:

- **RADIUS DTLS Port**: Used for authentication and accounting with default DTLS server port 2083
- **Shared Secret**: Uses RADIUS/DTLS as the shared secret when DTLS is enabled for a specific server
- **PAC Handling**: Downloads PAC from ISE for CTS communication and encrypts CTS attributes with the PAC key
- **Session Management**: Depends on DTLS server responses and closes sessions for ideal timeout or invalid responses
- **Load Balancing**: Supports multiple DTLS servers and load balancing methods

You can change the RADIUS DTLS port number using **DTLS port** *port_number*. For more information, see the Configuring RADIUS DTLS Port Number section.

RADSEC consists of two types: RADIUS-over-TLS (using TCP) and RADIUS-over-DTLS (using UDP). Cisco IOS-XE support RADIUS-over-DTLS (UDP) but does not support RADIUS-over-TLS (TCP), as outlined in RFC 7360.

RADIUS DTLS timeout and retry configuration includes these parameters:

- **Connection Timeout**: Default connection timeout is 5 seconds
- **Connection Retries**: Default connection retries is 5
- **Idle Timeout**: Default idle timeout is 60 seconds

After the encrypted RADIUS packet is sent, you need to start the retransmission timer. If you do not get a response before the retransmission timer expires, the packet is re-encrypted and re-transmitted. You can continue for number of times as per the **DTLS retries** configuration or till the default value. Once the number of tries exceeds the limit, the server becomes unavailable and responses are sent back to the AAA clients.

As the RADIUS DTLS is UDP based, you need to retry the connection after a specific timeout interval for a specific number of retries. After all retries are exhausted, the DTLS connection performs these actions:

- Is marked as unsuccessful.
- Looks up for the next available server for processing the RADIUS requests.

When the idle timer expires and no transactions exists since the last idle timeout, the DTLS session remains closed. After you establish the DTLS session, you can start the idle timer. If you start the idle timer for 30 seconds and one of the RADIUS DTLS packet is sent, then after 30 seconds, the idle timer expires and checks for number of RADIUS DTLS transactions. If the idle timer value exceeds zero, the idle timer resets the transaction counter and restarts the timer.

You can configure RADIUS servers with and without DTLS. It is recommended to create AAA server groups with DTLS enabled servers and non-DTLS servers. However, you will not find any such restriction while configuring AAA server groups.

Suppose you choose a DTLS server, the DTLS server establishes connection and RADIUS request packet is sent to the DTLS server. If the DTLS server does not respond after all RADIUS retries, it would fall over to the next configured server in the same server group. If the next server is a DTLS server, the processing of the RADIUS request packet continues with the next server. If the next server is a non-DTLS server, the processing of RADIUS request packet does not happen in that server group. Then the server group failover occurs and the same sequence continues with the next server group, if the next server group is available.

Note

You need to use either only DTLS or non-DTLS servers in a server group.

Requirement: Port RADIUS DTLS support to BINOS platforms

Outlines the prerequisite for porting RADIUS DTLS support from IOS to BINOS platforms where AAA servers run.

Support for IOS and BINOS AAA

Port RADIUS DTLS support to BINOS after completing implementation in IOS, since the AAA server runs on both IOS and BINOS platforms.

Configure RADIUS DTLS server (CLI)

Configure RADIUS DTLS server on a wireless controller.

Enable secure RADIUS communication using DTLS encryption on a wireless controller.

DTLS (Datagram Transport Layer Security) provides secure communication for RADIUS authentication in wireless networks. This configuration is used to protect RADIUS traffic between the wireless controller and RADIUS server.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Specify the RADIUS server name.

   Example

   ```
   Device(config)# radius server server-name
   ```

Example

```
Device(config)# radius server R1
```

4. Configure DTLS parameters.

Example

```
Device(config-radius-server)# dtls
```

5. Return to privileged EXEC mode.

Example

```
Device(config-radius-server)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RADIUS server is now configured with DTLS security, enabling encrypted communication for authentication requests.

Configure RADIUS DTLS connection timeout

Configure RADIUS DTLS connection timeout on a wireless controller.

Set the timeout value for DTLS connections to RADIUS servers to ensure proper connection management and security.

DTLS (Datagram Transport Layer Security) connection timeout configuration is necessary when using encrypted communication between the wireless controller and RADIUS servers. This setting determines how long the controller waits before timing out a DTLS connection attempt.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device# enable
```

2. Enter global configuration mode.

Example

```
Device# configure terminal
```

3. Specify the RADIUS server name.

Example

```
Device(config)# radius server server-name
```

Example

```
Device(config)# radius server R1
```

4. Configure RADIUS DTLS connection timeout.

Example

```
Device(config-radius-server)# dtls connectiontimeout timeout-value
```

Example

```
Device(config-radius-server)# dtls connectiontimeout 1
```

The *timeout-value* refers to the DTLS connection timeout value. The valid range is from 1 to 65535.

5. Return to privileged EXEC mode.

Example

```
Device(config-radius-server)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RADIUS DTLS connection timeout is now configured with the specified timeout value for the designated RADIUS server.

Configure RADIUS DTLS idle timeout

Configure RADIUS DTLS idle timeout on a wireless controller.

Set up RADIUS DTLS idle timeout to control the duration before idle DTLS connections are terminated.

RADIUS DTLS idle timeout configuration is necessary to manage connection resources and ensure proper cleanup of inactive DTLS sessions between the wireless controller and RADIUS server.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device# enable
```

2. Enter global configuration mode.

Example

```
Device# configure terminal
```

3. Specify the RADIUS server name.

Example

```
Device(config)# radius server server-name
```

Example

```
Device(config)# radius server R1
```

4. Configure RADIUS DTLS idle timeout.

Example

```
Device(config-radius-server)# dtls idletimeout idle-timeout-value
```

Example

```
Device(config-radius-server)# dtls idletimeout 2
```

The DTLS idle timeout value valid range is from 1 to 65535.

5. Return to privileged EXEC mode.

Example

```
Device(config-radius-server)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RADIUS DTLS idle timeout is configured and will terminate idle DTLS connections after the specified timeout period.

Configure source interface for RADIUS DTLS server

Configure source interface for RADIUS on a wireless controller.

Configure the source interface for RADIUS DTLS server to establish secure communication between the wireless controller and RADIUS server.

When configuring RADIUS authentication with DTLS encryption, you need to specify the source interface that the wireless controller will use to communicate with the RADIUS server. This ensures proper routing and security for authentication traffic.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device# enable
```

2. Enter global configuration mode.

Example

```
Device# configure terminal
```

3. Specify the RADIUS server name.

Example

```
Device(config)# radius server server-name
```

Example

```
Device(config)# radius server R1
```

4. Configure source interface for RADIUS DTLS server.

Example

```
Device(config-radius-server)# dtls ip radius source-interface Ethernet-Internal
 interface-number
```

Example

```
Device(config-radius-server)# dtls ip radius source-interface Ethernet-Internal
 0
```

The *interface-number* refers to the Ethernet-Internal interface number. The default value is 0.

5. Return to privileged EXEC mode.

Example

```
Device(config-radius-server)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The source interface for RADIUS DTLS server is now configured, enabling secure RADIUS communication through the specified Ethernet-Internal interface.

Configure RADIUS DTLS port number (CLI)

Configure RADIUS DTLS port number on a wireless controller.

Configure the RADIUS DTLS port number to establish secure RADIUS communication using DTLS protocol.

RADIUS DTLS provides secure authentication and accounting communication between the wireless controller and RADIUS server. Configure the DTLS port when you need to specify a custom port number for DTLS communication.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device# enable
```

2. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Specify the RADIUS server name.

   Example

   ```
   Device(config)# radius server server-name
   ```

   Example

   ```
   Device(config)# radius server R1
   ```

4. Configure RADIUS DTLS port number.

   Example

   ```
   Device(config-radius-server)# dtls port port-number
   ```

   Example

   ```
   Device(config-radius-server)# dtls port 2
   ```

   The *port-number* refers to the DTLS port number.

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config-radius-server)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RADIUS DTLS port number is configured and the controller can now use the specified port for secure DTLS communication with the RADIUS server.

Configure RADIUS DTLS connection retries (CLI)

Configure RADIUS DTLS connection retries on a wireless controller.

Set the number of RADIUS DTLS connection retry attempts to ensure reliable authentication server connectivity.

RADIUS DTLS connection retries help maintain authentication service availability when network conditions cause connection failures. Configure this setting when you need to control how many times the wireless controller attempts to reconnect to the RADIUS server before giving up.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Specify the RADIUS server name.

   Example

   ```
   Device(config)# radius server server-name
   ```

   Example

   ```
   Device(config)# radius server R1
   ```

4. Configure RADIUS connection retries.

   Example

   ```
   Device(config-radius-server)# dtls retries retry-number
   ```

   Example

   ```
   Device(config-radius-server)# dtls retries 3
   ```

   The *retry-number* refers to the DTLS connection retries. The valid range is from 1 to 65535.

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config-radius-server)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RADIUS DTLS connection retry count is configured and the wireless controller will attempt the specified number of reconnections before failing over or giving up.

Configure RADIUS DTLS trustpoint (CLI)

Configure RADIUS DTLS trustpoint on a wireless controller.

Establish secure DTLS communication between the wireless controller and RADIUS server by configuring appropriate trustpoints for authentication.

RADIUS DTLS trustpoints provide secure communication channels for RADIUS authentication traffic. Configure both client and server trustpoints to ensure proper certificate validation during DTLS handshakes.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Specify the RADIUS server name.

   Example

   ```
   Device(config)# radius server server-name
   ```

   Example

   ```
   Device(config)# radius server R1
   ```

4. Configure trustpoint for client and server.

   Example

   ```
   Device(config-radius-server)# dtls trustpoint client client-trustpoint-name
   dtls
   ```

   Example

   ```
   Device(config-radius-server)# dtls trustpoint server server-trustpoint-name
   dtls
   ```

   Example

   ```
   Device(config-radius-server)# dtls trustpoint client client1 dtls
   ```

   Example

   ```
   Device(config-radius-server)# dtls trustpoint server server1 dtls
   ```

5. Return to privileged EXEC mode.

Example

```
Device(config-radius-server)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The RADIUS DTLS trustpoints are now configured, enabling secure DTLS communication between the wireless controller and RADIUS server for authentication traffic.

Configure RADIUS DTLS match-server-identity

Configure RADIUS DTLS trustpoint on a wireless controller.

This task configures RADIUS DTLS match-server-identity parameters to establish secure authentication between the wireless controller and RADIUS server.

RADIUS DTLS (Datagram Transport Layer Security) provides encrypted communication for RADIUS authentication. The match-server-identity feature validates the RADIUS server's certificate against its hostname or IP address to ensure secure connections.

Procedure

---

1. Configure the RADSEC certification validation parameters.

   Example

   ```
   dtls match-server-identity hostname <name>
   ```

2. Configure the RADSEC certification validation parameters.

   Example

   ```
   dtls match-server-identity ip-address <IPv4 or IPv6>
   ```

The RADIUS DTLS match-server-identity parameters are configured, enabling secure certificate validation for RADSEC communication.

Configure DTLS Dynamic Author (CLI)

Configure DTLS Dynamic Author on a wireless controller.

Enable DTLS support for RADIUS dynamic authorization (RFC 3576) on the wireless controller.

DTLS Dynamic Author provides secure communication for RADIUS Change of Authorization (CoA) and disconnect messages. This configuration is needed when implementing dynamic policy changes and session management in wireless networks.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device# enable
```

2. Enter global configuration mode.

Example

```
Device# configure terminal
```

3. Configure local server profile for RFC 3576 support.

Example

```
Device(config)# aaa server radius dynamic-author
```

4. Configure DTLS source parameters.

Example

```
Device(config-locsvr-da-radius)# dtls
```

5. Return to privileged EXEC mode.

Example

```
Device(config-locsvr-da-radius)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

DTLS Dynamic Author is now configured on the wireless controller, enabling secure RADIUS dynamic authorization communications.

Enable DTLS for client (CLI)

Enable DTLS for client on a wireless controller.

Enable DTLS encryption for client communications to secure RADIUS dynamic authorization messages.

DTLS (Datagram Transport Layer Security) provides encryption for UDP-based communications between the wireless controller and RADIUS clients. This configuration enhances security for RFC 3576 dynamic authorization operations.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device# enable
```

2. Enter global configuration mode.

Example

```
Device# configure terminal
```

3. Configure local server profile for RFC 3576 support.

Example

```
Device(config)# aaa server radius dynamic-author
```

4. Enable DTLS for the client.

Example

```
Device(config-locsvr-da-radius)# client IP-address dtls
```

Example

```
Device(config-locsvr-da-radius)# client 10.104.49.14 dtls
```

5. Return to privileged EXEC mode.

Example

```
Device(config-locsvr-da-radius)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

DTLS is now enabled for the specified client, providing encrypted communication for RADIUS dynamic authorization messages.

Configure client trustpoint for DTLS (CLI)

Configure client trustpoint for DTLS on a wireless controller.

Establish secure DTLS communication between the wireless controller and RADIUS dynamic authorization clients.

Client trustpoint configuration for DTLS is required when implementing secure communication for RFC 3576 dynamic authorization services on wireless controllers.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device# enable
```

2. Enter global configuration mode.

Example

```
Device# configure terminal
```

3. Configure local server profile for RFC 3576 support.

Example

```
Device(config)# aaa server radius dynamic-author
```

4. Configure client trustpoint for DTLS.

Example

```
Device(config-locsvr-da-radius)# client IP-address dtls client-tp
client-tp-name
```

Example

```
Device(config-locsvr-da-radius)# client 10.104.49.14 dtls client-tp
client_tp_name
```

5. Return to privileged EXEC mode.

Example

```
Device(config-locsvr-da-radius)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The client trustpoint for DTLS is now configured, enabling secure communication between the wireless controller and the specified RADIUS dynamic authorization client.

Configure DTLS idle timeout (CLI)

Configure DTLS idle timeout on a wireless controller.

Set the DTLS idle timeout interval to control when inactive DTLS connections are terminated.

DTLS idle timeout configuration is necessary to manage inactive DTLS connections and optimize resource utilization on wireless controllers.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device# enable
```

2. Enter global configuration mode.

Example

```
Device# configure terminal
```

3. Configure local server profile for RFC 3576 support.

Example

```
Device(config)# aaa server radius dynamic-author
```

4. Configure DTLS idle time.

Example

```
Device(config-locsvr-da-radius)# client IP_addr dtls idletimeout
timeout-interval  {client-tp client_tp_name | server-tp server_tp_name}client
IP-address dtls idletimeout timeout-interval client-tp client-tp-name
```

Example

```
Device(config-locsvr-da-radius)# client 10.104.49.14 dtls idletimeout 62
client-tp dtls_ise
```

The *timeout-interval* refers to the idle timeout interval. The valid range is from 60 to 600.

5. Return to privileged EXEC mode.

Example

```
Device(config-locsvr-da-radius)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The DTLS idle timeout is now configured and will terminate inactive DTLS connections after the specified timeout interval.

Configure server trustpoint for DTLS (CLI)

Configure server trustpoint for DTLS on a wireless controller.

Enable secure DTLS communication between the wireless controller and RADIUS clients by configuring a server trustpoint.

Server trustpoints are required when implementing DTLS (Datagram Transport Layer Security) for RFC 3576 Change of Authorization (CoA) messages between RADIUS clients and the wireless controller. This ensures encrypted and authenticated communication.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device# enable
```

2. Enter global configuration mode.

Example

```
Device# configure terminal
```

3. Configure local server profile for RFC 3576 support.

Example

```
Device(config)# aaa server radius dynamic-author
```

4. Configure server trust point for the DTLS client.

Example

```
Device(config-locsvr-da-radius)# client IP-address dtls server-tp
server-trustpoint-name
```

Example

```
Device(config-locsvr-da-radius)# client 10.104.49.14 dtls server-tp dtls_client
```

5. Return to privileged EXEC mode.

Example

```
Device(config-locsvr-da-radius)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The server trustpoint for DTLS is now configured, enabling secure communication between the wireless controller and the specified RADIUS client for Change of Authorization messages.

Verify RADIUS DTLS server configuration

Provides command to view information about the DTLS enabled servers.

To view information about the DTLS enabled servers, use this command:

```
Device# show aaa servers
DTLS: Packet count since last idletimeout 1,
Send handshake count 3,
Handshake Success 1,
Total Packets Transmitted 1,
Total Packets Received 1,
Total Connection Resets 2,
Connection Reset due to idle timeout 0,
Connection Reset due to No Response 2,
Connection Reset due to Malformed packet 0,
```

Clear RADIUS DTLS specific statistics

Clear DTLS server statistics to reset monitoring counters.

Clear RADIUS DTLS specific statistics to reset monitoring counters and start fresh data collection.

Use this task when you need to reset DTLS server statistics for monitoring purposes or troubleshooting.

Procedure

Clear the RADIUS DTLS specific statistics.

Example

```
Device# clear aaa counters servers radius {<server-id> | all}
```

Note

Here, *server-ID* refers to the server ID displayed by **show aaa servers**. The valid range is from 0 to 2147483647.

The RADIUS DTLS specific statistics are cleared and reset to zero.

139 Policy Enforcement and Usage Monitoring

Topics:

- Policy enforcement and usage monitoring

Outlines policy enforcement and usage monitoring concepts, guides through configuration using the command-line interface, and provides methods for verifying policy application and effectiveness.

Policy enforcement and usage monitoring

Explains how policy enforcement and usage monitoring provide dynamic QoS policy application and real-time data rate control for 802.11 clients.

A policy enforcement and usage monitoring system is a network management mechanism that

- applies dynamic QoS policies and enforces upstream and downstream TCP or UDP data rate limits on 802.11 clients
- monitors each client's data usage without disrupting ongoing sessions, and
- supports multiple wireless authentication methods such as 802.1X, PSK, and web authentication.

APs send client statistics to the controller periodically. This includes bandwidth consumption. The AAA server receives Accounting-Interim messages at configured intervals to accumulate consumption data for each client. When a client exceeds the configured data threshold, the AAA server sends a Change-of-Authorization (CoA) message to the controller. This process ensures continuous client connectivity. It maintains session persistence during policy changes.

> **Note**
>
> Only FlexConnect local switching mode is supported.

Configure policy enforcement and enable CoA (CLI)

Use this procedure to configure policy enforcement and enable change-of-authorization using command.

Configure network device policy enforcement and enable dynamic reauthorization for RADIUS clients.

This task is required when implementing RADIUS-based policy control on supported network devices and enabling Change-of-Authorization capabilities for dynamic policy updates.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a local server RADIUS profile in the controller.

   Example

   ```
   Device(config)# aaa server radius dynamic-author
   ```

3. Configure a server key for a RADIUS client.

   Example

   ```
   Device(config-locsvr-da-radius)# client client-ip-addr server-key key
   ```

4. Optional: Display the AAA CoA packet statistics.

Example

```
Device# show aaa command handler
```

The device is now configured to enforce policy and support change-of-authorization for RADIUS clients. You can verify configuration or monitor status as needed.

Example: Configure policy enforcement and usage monitoring

This section lists a sample configuration for applying policy enforcement and usage monitoring on a wireless network group.

Policy enforcement and usage monitoring is applied on a group where a class-map is created for quality of service (QoS) policies. This is done using Change of Authorization (CoA).

Here is a sample configuration for policy enforcement and usage monitoring:

```
aaa new-model
  radius server radius_free
  address ipv4 10.0.0.1 auth-port 1812 acct-port 1813
  key cisco123
  exit

aaa new-model
  aaa server radius dynamic-author
  client 10.0.0.1 server-key cisco123
aaa new-model
  aaa group server radius rad_eap
  server name radius_free
  exit
aaa new-model
  dot1x system-auth-control
  aaa authentication dot1x eap_methods group rad_eap
  dot1x system-auth-control
class-map client_dscp_clsmapout
match dscp af13
exit
class-map client_dscp_clsmapin
match dscp af13
exit
policy-map qos_new
  class client_dscp_clsmapout
  police 512000 conform-action transmit exceed-action drop
  policy-map qos_nbn
  class client_dscp_clsmapin
  police 16000000 conform-action transmit exceed-action drop
wlan test1 3 test2
  broadcast-ssid
  security wpa wpa2 ciphers aes
  security dot1x authentication-list eap_methods
no shutdown
exit
wireless profile policy named-policy-profile
shutdown
  vlan 10
  aaa-override
  no central association
  no central dhcp
```

```
  no central switching
  no shutdown
wireless tag policy named-policy-tag
  wlan test1 policy named-policy-profile
wireless profile flex FP_name_001
  native-vlan-id 10
wireless tag site ST_name_001
  no local-site
  flex-profile FP_name_001
  exit
ap test-ap
  policy-tag named-policy-tag
  site-tag ST_name_001
  exit
aaa authorization network default group radius
exit
```

Verify policy usage and enforcement

This section lists CLI commands to view detailed information about policy usage and enforcement.

To view the detailed information about the policies applied to a specific client, use this command:

```
Device# show wireless client mac-address mac-address detail
```

To view client-level mobility statistics, use this command:

```
Device# show wireless client mac-address mac-address mobility statistics
```

To view client-level roaming history for an active client in a sub-domain, use this command:

```
Device# show wireless client mac-address mac-address mobility history
```

To view detailed parameters of a given profile policy, use this command:

```
Device# show wireless profile policy detailed policy-name
```

140 Local Extensible Authentication Protocol

Topics:

- Local EAP

Local EAP

Explains how Local Extensible Authentication Protocol (EAP) enables the controller to act as an authentication server.

Local EAP is a controller feature that

- makes the controller act as both authenticator and authentication server
- allows 802.1x authentication on WPA Enterprise wireless clients without the use of any RADIUS server, and
- refers to the EAP authentication server activity and not necessarily tied to the user credentials validation that can be delegated to an external LDAP database.

Feature scenarios

Local EAP is designed to allow administrators to use Enterprise-grade 802.1x authentication for a limited number of users in situations and branches where an external dedicated RADIUS server may not be available. It can also work as an emergency backup in case the RADIUS server is not available.

You can implement Local EAP either with users local to the controller or use an external LDAP database to store the user credentials.

Restrictions for local EAP

Outlines the limitations and requirements when using local EAP profile on a wireless controller.

Consider these restrictions when implementing local EAP on wireless controllers.

- It is not possible to configure AAA attributes, such as per-user ACL or per-user session timeout using local EAP.
- Local EAP only allows user database either locally on the controller or on an external LDAP database.
- Local EAP supports TLS 1.2 as of 17.1 and later software release.
- Local EAP uses the trustpoint of your choice on the controller. You will either need to install a publicly trusted certificate on the controller or import it on the clients for the EAP session to be trusted by the client.
- Local EAP supports *EAP-FAST*, *EAP-TLS*, and *PEAP* as EAP authentication methods.

> **Note**
>
> *PEAP-mschapv2* does not work when using certain external LDAP databases that only support clear text passwords.

- When FIPS mode is enabled on the controller, the password for local EAP users must be configured with an encryption higher than type six, otherwise the users fail to authenticate. To ensure successful authentication, configure the password using the command **algorithm-type sha256 secret <password>**.

Configure local EAP profile (CLI)

Configure local EAP profile using commands.

Create and configure a local EAP profile to support authentication methods for wireless clients.

EAP profiles define authentication methods and security parameters for wireless network access. This configuration establishes a local EAP profile with PEAP method and PKI trustpoint for secure client authentication.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create an EAP profile.

   Example

   ```
   Device(config)# eap profile profile-name
   ```

   Example

   ```
   Device(config)# eap profile mylocapeap
   ```

3. Configure the PEAP method on the profile.

   Example

   ```
   Device(config-eap-profile)# method peap
   ```

4. Configure the PKI trustpoint on the profile.

   Example

   ```
   Device(config-eap-profile)# pki-trustpoint trustpoint-name
   ```

   Example

   ```
   Device(config-eap-profile)# pki-trustpoint admincert
   ```

The local EAP profile is configured with PEAP authentication method and associated PKI trustpoint, ready for use in wireless network authentication.

Configure local EAP profile (GUI)

Configure local EAP profile using GUI.

Configure a local EAP profile to enable secure authentication for wireless clients using various EAP methods.

Use this procedure to create a local EAP profile through the graphical user interface when you need to provide secure wireless authentication capabilities.

Procedure

1. Choose **Configuration** > **Security** > **Local EAP**.
2. Click **Add**.
3. In the **Create Local EAP Profiles** page, enter a profile name.

> **Note**
>
> It is not advised to use LEAP EAP method due to its weak security. You can use any of these EAP methods to configure a trustpoint:
>
> - EAP-FAST
> - EAP-TLS
> - PEAP
>
> Clients do not trust the default controller certificate, so you need to deactivate the server certificate validation on the client side or install a certificate trustpoint on the controller.

4. Click **Apply to Device**.

The local EAP profile is successfully created and applied to the device, enabling secure authentication for wireless clients.

Configure AAA authentication (GUI)

Configure AAA authentication using the GUI interface.

This task enables you to set up authentication, authorization, and accounting (AAA) authentication methods using the graphical user interface to control network access and security.

AAA authentication is a security framework that controls access to network resources. You can configure different authentication methods such as local authentication or external authentication servers to validate user credentials before granting network access.

Procedure

---

1. Choose **Configuration** > **Security** > **AAA**, and navigate to the **AAA Method List** > **Authentication** tabs.
2. Click **Add**.
3. Choose **dot1x** as the **Type** and **local** as the **Group Type**.
4. Click **Apply to Device**.

The AAA authentication method is configured and applied to the device. The system will now use the specified authentication method (dot1x with local group type) to validate user access requests.

Configure AAA authorization method (GUI)

Configure AAA authorization method using GUI.

Configure AAA authorization methods to control access permissions for network and credential-download operations.

AAA authorization methods define how the system determines what actions authenticated users are permitted to perform. This configuration is essential for maintaining network security and access control.

Procedure

---

1. Navigate to **Authorization** sub-tab.

2. Create a new method for **credential-download** type and point it to local.

> **Note**
>
> Perform the same for **network** authorization type.

AAA authorization methods are configured for both credential-download and network authorization types, pointing to local authentication.

Configure AAA authorization method (CLI)

Configure AAA authorization method using commands.

Configure AAA authorization method to establish authentication and authorization policies for network access control using CLI commands.

Use this procedure when you need to set up AAA authorization method through the command line interface to control network access and credential management.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a AAA authentication model.

   Example

   ```
   Device(config)# aaa new-model
   ```

3. Configure the default local RADIUS server.

   Example

   ```
   Device(config)# aaa authentication dot1x default local
   ```

4. Configure default database to download credentials from local server.

   Example

   ```
   Device(config)# aaa authorization credential-download default local
   ```

5. Configure the local authentication method list.

Example

```
Device(config)# aaa local authentication default authorization default
```

6. Configure authorization for network services.

Example

```
Device(config)# aaa authorization network default local
```

AAA authorization method is configured with local authentication and authorization for dot1x, credential download, and network services.

Configure local advanced methods (GUI)

Configure local advanced methods using GUI.

Configure local advanced authentication and authorization methods to control access to network resources using the graphical user interface.

Local advanced methods provide enhanced security controls for authentication and authorization in AAA configurations. Use this procedure when you need to set default local authentication and authorization methods through the GUI.

Procedure

---

1. In the **Configuration** > **Security** > **AAA** window, perform the following:
   a. Navigate to **AAA Advanced** tab.
   b. From the **Local Authentication** drop-down list, choose a default local authentication.
   c. From the **Local Authorization** drop-down list, choose a default local authorization.
2. Click **Apply**.

The local advanced authentication and authorization methods are configured and applied to the AAA settings.

Configure WLAN (GUI)

Configure WLAN using GUI.

Configure a WLAN profile to enable wireless network access for clients.

WLAN configuration allows you to define wireless network parameters including SSID, radio policy, and authentication settings.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. In the **WLANs** window, click the name of the **WLAN** or click **Add** to create a new one.
3. In the **Add/Edit WLAN** window that is displayed, click the **General** tab to configure the following parameters.
   - In the **Profile Name** field, enter or edit the name of the profile.

- In the **SSID** field, enter or edit the SSID name.

  The SSID name can be alphanumeric, and up to 32 characters in length.
- In the **WLAN ID** field, enter or edit the ID number. The valid range is between 1 and 512.
- From the **Radio Policy** drop-down list, choose the **802.11** radio band.
- Using the **Broadcast SSID** toggle button, change the status to either **Enabled** or **Disabled**.
- Using the **Status** toggle button, change the status to either **Enabled** or **Disabled**.

4. In the **AAA** tab, you can configure the following:
   a. Choose an authentication list from the drop-down.
   b. Check the **Local EAP Authentication** check box to enable local EAP authentication on the WLAN. Also, choose the required **EAP Profile Name** from the drop-down list.
5. Click **Save & Apply to Device**.

The WLAN is configured and applied to the device. The wireless network is now available for client connections with the specified settings.

Configure WLAN (CLI)

Configure WLAN using commands.

Configure a WLAN using CLI commands to establish wireless network connectivity with dot1x security authentication.

Use this procedure when you need to set up a wireless LAN with security authentication through the command line interface.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id ssid-name
   ```

   ```
   Device(config)# wlan localpeapssid 1 localpeapssid
   ```

   wlan-name: Is the name of the configured WLAN.

   wlan-id: Is the wireless LAN identifier. The range is 1 to 512.

   ssid-name: Is the SSID name which can contain 32 alphanumeric characters.

> **Note**
>
> If you have already configured this command, enter **wlan** *wlan-name* command.

3. Enable security authentication list for dot1x security. The configuration is similar for all dot1x security WLANs.

   Example

   ```
   Device(config-wlan)# security dot1x authentication-list auth-list-name
   ```

   ```
   Device(config-wlan)# security dot1x authentication-list default
   ```

4. Set EAP Profile on an WLAN.

   Example

   ```
   Device(config-wlan)# local-auth profile-name
   ```

   ```
   Device(config-wlan)# local-auth mylocaleap
   ```

   profile name: Is the EAP profile on an WLAN.

The WLAN is configured with dot1x security authentication and the specified EAP profile.

Create a user account (CLI)

Configure a user account using commands.

This task creates a new user account with specified parameters including username, password, creation time, description, and user type.

Use this procedure when you need to create a user account through the command-line interface. This method allows you to specify detailed account parameters during the creation process.

Before you begin

Follow these steps to create a user account using CLI commands:

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a user account.

Example

```
Device(config)# user-name user-name
```

```
Device(config)# user-name 1xuser
```

3. Set the creation time of the user account.

   Example

   ```
   Device(config)# creation-time time
   ```

   ```
   Device(config)# creation-time 1572730075
   ```

4. Add a user-defined description to the new user account.

   Example

   ```
   Device(config)# description user-name
   ```

   ```
   Device(config)# description 1xuser
   ```

5. Create a password for the user account.

   Example

   ```
   Device(config)# password 0 password
   ```

   ```
   Device(config)# password 0 Cisco123
   ```

6. Specify the type of user account.

   Example

   ```
   Device(config)# type network-user description user-name
   ```

   ```
   Device(config)# type network-user description 1xuser
   ```

The user account is created with the specified username, password, creation time, description, and user type. The account is now available for use on the device.

Attach a policy profile to a WLAN interface (GUI)

Configure policy profile attachment to a WLAN interface using GUI.

This task allows you to attach a policy profile to a WLAN interface to control network access and security policies for wireless connections.

Policy profiles define access control rules and security settings for WLAN interfaces. Attaching a policy profile to a WLAN interface ensures that wireless clients connecting through that interface are subject to the defined policies.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. On the **Manage Tags** page, click **Policy** tab.
3. Click **Add** to view the **Add Policy Tag** window.
4. Enter a name and description for the policy tag.
5. Click **Add** to map the WLAN and policy.
6. Choose the WLAN profile to map with the appropriate policy profile, and click the tick icon.
7. Click **Save & Apply to Device**.

The policy profile is attached to the WLAN interface. The configuration is saved and applied to the device, ensuring that the policy rules are enforced for clients connecting through the specified WLAN interface.

Deploy policy tag to access points (GUI)

Configure policy tag deployment to access points using the GUI interface.

Deploy policy tags to access points to apply specific wireless policies and configurations to designated access points in your network.

Use this procedure when you need to assign policy tags to access points through the graphical user interface. Policy tags allow you to apply consistent configurations across multiple access points.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. On the **All Access Points** page, click the access point you want to configure.

   Make sure that the tags assigned are the ones you configured.
3. Click **Apply**.

The policy tag is successfully deployed to the selected access point, and the wireless policies associated with the tag are applied to that access point.

141 Local EAP Ciphersuite

Topics:

- Local EAP Ciphersuite

Outlines the Local Extensible Authentication Protocol ciphersuite feature, explaining how to configure and manage supported ciphersuites on the controller for secure local authentication and enhanced compatibility.

Local EAP Ciphersuite

This topic explains the Local EAP ciphersuite feature, which controls the list of ciphersuites that the controller, acting as an SSL server, supports for local authentication. It describes the default behavior and the ability to configure supported ciphersuites.

A local EAP ciphersuite is a security protocol configuration that

- determines which encryption algorithms the controller supports during EAP transactions
- allows customization of enabled or disabled ciphersuites for improved security and compatibility, and
- is manageable by administrators from Cisco IOS XE Cupertino 17.7.1 Release onwards.

Feature history

Feature Name	Release	Description
Ciphersuite Selection for Local EAP Authentication	Cisco IOS XE 17.1.1s	This feature provides the controller with a knob to control the list of ciphersuites used in local EAP authentication Prior to Cisco IOS XE Cupertino 17.7.1 Release, the controller acts as an SSL server supporting a hardcoded list of ciphersuites for each EAP application.

Supported and configurable ciphersuites for Local EAP

The controller, acting as an SSL server, negotiates ciphersuites with the client during the SSL handshake. The client sends a prioritized list of supported ciphersuites, and the server selects one that is mutually acceptable.

The table lists the hardcoded ciphersuites and their descriptions:

Table 120: Hardcoded List of Ciphersuites

Ciphersuites	Description
aes128-sha	Encryption Type tls_rsa_with_aes_128_cbc_sha.
aes256-sha	Encryption Type tls_rsa_with_aes_256_cbc_sha.
dhe-rsa-aes-gcm-sha2	Encryption Type tls_dhe_rsa_with_aes_128_gcm_sha256 and tls_dhe_rsa_with_aes_256_gcm_sha384(TLS1.2 and above).
dhe-rsa-aes-sha2	Encryption Type tls_dhe_rsa_with_aes_128_cbc_sha256 and tls_dhe_rsa_with_aes_256_cbc_sha256 (TLS 1.2 and above).
dhe-rsa-aes128-sha	Encryption Type tls_dhe_rsa_with_aes_128_cbc_sha.
dhe-rsa-aes256-sha	Encryption Type tls_dhe_rsa_with_aes_256_cbc_sha.
ecdhe-ecdsa-aes-gcm-sha2	Encryption Type tls_ecdhe_ecdsa_with_aes_128_gcm_sha256 and tls_ecdhe_ecdsa_with_aes_256_gcm_sha384(TLS1.2 and above).

Ciphersuites	Description
ecdhe-ecdsa-aes-sha	Encryption Type tls_ecdhe_ecdsa_with_aes_128_cbc_sha and tls_ecdhe_ecdsa_with_aes_256_cbc_sha.
ecdhe-ecdsa-aes-sha2	Encryption Type tls_ecdhe_ecdsa_with_aes_128_cbc_sha256 and tls_ecdhe_ecdsa_with_aes_256_cbc_sha384(TLS1.2 and above).
ecdhe-rsa-aes-gcm-sha2	Encryption Type tls_ecdhe_rsa_with_aes_128_gcm_sha256 and tls_ecdhe_rsa_with_aes_256_gcm_sha384(TLS1.2 and above).
ecdhe-rsa-aes-sha	Encryption Type tls_ecdhe_rsa_with_aes_128_cbc_sha and tls_ecdhe_rsa_with_aes_256_cbc_sha.
ecdhe-rsa-aes-sha2	Encryption Type tls_ecdhe_rsa_with_aes_128_cbc_sha256 and tls_ecdhe_rsa_with_aes_256_cbc_sha384(TLS1.2 and above).

Note

By default, all the ciphersuites are supported. Using the Local EAP ciphersuite feature, you can enable or disable the ciphersuites based on your requirement.

Restrictions for Local EAP Ciphersuite

Adhere to these restrictions when configuring local EAP Ciphersuite on a wireless controller.

- SNMP is not supported.
- Ciphersuites are specific to 802.1X.

Configure Local EAP Ciphersuite (CLI)

Use this procedure to configure local EAP ciphersuite using commands.

Configure Local Extensible Authentication Protocol (EAP) ciphersuite for secure authentication.

This configuration is used in environments where EAP authentication is required for network access.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Enter global configuration mode.

Example

```
Device# configure terminal
```

3. Create an EAP profile.

Example

```
Device(config)# eap profile name
```

Example

```
Device(config)# eap profile local_EAP_TLSv1
```

4. Select a ciphersuite.

Example

```
Device(config-eap-profile)# ciphersuite cipher-suite
```

Using this command, you will be able to configure only one ciphersuite. To configure more than one ciphersuite, you need to issue this command with various ciphersuites.

By default, all ciphersuites are supported if you issue the **no ciphersuite** command.

> **Note**
>
> To remove the ciphersuites, remove the ciphersuites one by one or all at once.

5. Return to privileged EXEC mode.

Example

```
Device(config-eap-profile)# end
```

Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The local EAP ciphersuite is now configured for secure authentication.

142 Authentication and Authorization Between Multiple RADIUS Servers

Topics:

- Authentication and authorization between multiple RADIUS servers
- Configure web authentication for WLAN with split authentication and authorization servers
- RADIUS Message-Authenticator support for non-EAP requests

This chapter explains how to configure and verify split RADIUS authentication and authorization on Cisco Catalyst 9800 controllers, including GUI and CLI steps for server groups, AAA lists, 802.1X, and web authentication.

Authentication and authorization between multiple RADIUS servers

Explains how authentication and authorization roles are managed across single and multiple RADIUS servers in wireless controller configurations.

A RADIUS authentication and authorization role is a function within a RADIUS-based wireless controller configuration that:

- allows authentication and authorization duties to be carried out by a single RADIUS server or split between different servers
- enables the controller to relay authentication attributes to a separate authorization server, and
- in a split configuration, requires both servers to return ACCESS-ACCEPT for a client session to be established.

Cisco Catalyst 9800 Series Wireless Controller use a request and response transaction with a single RADIUS server that combines both authentication and authorization.

A RADIUS server can assume the role of either an authentication server, authorization server, or both. In cases where there are separate RADIUS servers for authentication and authorization, the Session Aware Networking (SANet) component on the controller allows authentication on one server and authorization on another when a client joins controller .

Authentication can be performed using Cisco ISE, Cisco Catalyst Center, Free RADIUS, or any third-party RADIUS Server. After the authentication server successfully authenticates a user, the controller relays attributes received from the authentication server to another RADIUS server designated as the authorization server.

The authorization server then performs these:

- Processes received attributes with the other policies or rules defined on the server.
- Derives attributes as part of the authorization response and returns them to the controller .

> **Note**
>
> The Authentication/Authorization list created through Cisco Catalyst Center provisioning supports a maximum of 100 entries. Entries beyond 100 can be created, but they do not function.

Configure explicit authentication and authorization server list (GUI)

Use this procedure to configure explicit authentication and authorization server list using GUI.

Set up and customize authentication and authorization server lists to control network access and user verification methods using the GUI.

Designate specific authentication and authorization servers, such as RADIUS, TACACS+, or LDAP, to handle AAA (Authentication, Authorization, and Accounting) operations on your device.

Procedure

1. Choose **Configuration** > **Security** > **AAA**. On the **Authentication Authorization and Accounting** page, click the **Servers/Groups** tab. Click the type of AAA server you want to configure from these options:
   - RADIUS
   - TACACS+
   - LDAP

   In this procedure, the RADIUS server configuration is described.

2. With the **RADIUS** option selected, click **Add** and enter a name for the RADIUS server and the IPv4 or IPV6 address of the server.
3. Enter the authentication and encryption key to be used between the device and the key string RADIUS daemon running on the RADIUS server. You can choose to use either a PAC key or a non-PAC key.
4. Enter the server timeout value; the valid range is 1 to 1000 seconds. Enter a retry count; the valid range is 0 to 100.
5. Leave the **Support for CoA** field in **Enabled** state and click **Save & Apply to Device**.
6. On the **Authentication Authorization and Accounting** page, with **RADIUS** option selected, click the **Server Groups** tab. In the **Create AAA RADIUS Server Group** window that is displayed, enter a name for the RADIUS server group.
7. From the **MAC-Delimiter** drop-down list, select the delimiter to be used in the MAC addresses that are sent to the RADIUS servers. From the **MAC Filtering** drop-down list, select a value based on which to filter MAC addresses.
8. To configure dead time for the server group and direct AAA traffic to alternative groups of servers that have different operational characteristics, in the **Dead-Time** field, enter the amount of time, in minutes, after which a server is assumed to be dead.
9. Choose the servers that you want to include in the server group from the **Available Servers** list and move them to the **Assigned Servers** list. Click **Save & Apply to Device**.

Configure explicit authentication server list (GUI)

Use this procedure to configure explicit authentication server list using the GUI.

Set up and manage RADIUS authentication servers and server groups to control authentication and accounting within your network environment using the GUI.

Use this task when you need to explicitly designate which RADIUS servers and groups are available for authentication in your AAA configuration, using the device's graphical interface.

Procedure

1. Choose **Configuration** > **Security** > **AAA** > **Servers/Groups**.
2. Choose **RADIUS** > **Servers** tab.
3. Click **Add** to add a new server or click an existing server.
4. Enter the **Name**, the **Server Address**, **Key**, **Confirm Key**, **Auth Port** and **Acct Port**. Check the **PAC Key** checkbox and enter the **PAC key** and **Confirm PAC Key**.
5. Click **Apply to Device**.
6. Choose **RADIUS** > **Server Groups** and click **Add** to add a new server group or click an existing server group.
7. Enter the **Name** of the server group and choose the servers that you want to include in the server group, from the **Available Servers** list and move them to the **Assigned Servers** list.
8. Click **Apply to Device**.

Configure explicit authentication server list (CLI)

Use this procedure to configure explicit authentication server list using commands.

Set up an authentication server list by explicitly defining RADIUS server parameters using commands.

Use this procedure on Cisco devices when you require explicit configuration of authentication server parameters for AAA.

Procedure

1. Enable the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Specify the RADIUS server name.

   Example

   ```
   Device(config)# radius server server name
   ```

4. Specify the RADIUS server parameters.

   Example

   ```
   Device(config-radius-server)# address ipv4 192.0.2.56 auth-port 1812 acct-port
    1813
   ```

5. Specify the authentication and encryption key used between the device and the key string RADIUS daemon running on the RADIUS server.

   Example

   ```
   Device(config-radius-server)# pac key key
   ```

6. Return to the configuration mode.

   Example

   ```
   Device(config-radius-server)# exit
   ```

7. Create a radius server-group identification.

   Example

   ```
   Device(config)# aaa group server radius server-group
   ```

   server-group refers to the server group name. The valid range is from 1 to 32 alphanumeric characters.

   If the IP address of the RADIUS server is not added to the routes defined for the controller, the default route is used. We recommend that you define a specific route to source the traffic from the defined SVI in the AAA server group.

8. Configure the server name.

Example

```
Device(config)# server name server-name
```

9. Return to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Example

```
Device(config)# end
```

For more information, see *Configuring AAA for External Authentication*.

Configure explicit authorization server list (GUI)

Use this procedure to configure explicit authorization server list using the GUI.

Set up and organize RADIUS authorization servers and groups for secure network authentication and accounting using the GUI.

Configuring an explicit authorization server list ensures that devices can authenticate users and account activity through specified RADIUS servers. This centralized management helps enforce access policies and simplifies administration.

Procedure

1. Choose **Configuration** > **Security** > **AAA** > **Servers/Groups**.
2. Choose **RADIUS** > **Servers** tab.
3. Click **Add** to add a new server or click an existing server.
4. Enter the **Name**, the **Server Address**, **Key**, **Confirm Key**, **Auth Port** and **Acct Port.** Check the **PAC Key** checkbox and enter the **PAC key** and **Confirm PAC Key**.
5. Click **Apply to Device**.
6. Choose **RADIUS** > **Server Groups** and click **Add** to add a new server group or click an existing server group.
7. Enter the **Name** of the server group and choose the servers that you want to include in the server group, from the **Available Servers** list and move them to the **Assigned Servers** list.
8. Click **Apply to Device**.

Configure explicit authorization server list (CLI)

Use this procedure to configure explicit authorization server list using commands.

You must direct AAA validation access requests to a specific server. Otherwise, the wireless controller selects any available AAA server from the configured pool to validate access requests. Because the controller cannot target a specific server for validation, you may face inefficiencies and potential security vulnerabilities.

To address this, an authorization method list is created to allow administrators to explicitly direct the controller to use a specific AAA server or the local database for validation. This configuration ensures a streamlined and secure validation process by eliminating randomness in server selection.

Benefits of this approach are:

- **Improved Control**: You can explicitly define which AAA server handles validation requests and ensure precise control over server selection.
- **Flexibility**: You can configure either an external AAA server or a local database based on the network's requirements.

- **Enhanced Security**: By specifying the AAA server, you can ensure that only trusted servers are used for validation, reducing the risk of unauthorized access.

Follow these steps to configure an explicit authorization server list using commands:

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Set up the security framework by enabling AAA features for authentication, authorization, and accounting.

   Example

   ```
   Device# aaa new-model
   ```

4. Configure a RADIUS server to enable centralized authentication, authorization, and accounting for secure and scalable network access control.
   a) Specify the RADIUS server name.
   b) Specify the RADIUS server parameters.
   c) Specify the authorization and encryption key used between the device and the RADIUS daemon running on the RADIUS server by key string encryption.

   Example

   ```
   Device(config)# radius server server-name
   ```

   ```
   Device(config)# address ipv4 address auth-port auth_port_number acct-port
   acct_port_number
   ```

   ```
   Device(config)# pac key key
   ```

   ```
   Device(config)# end
   ```

5. Configure the RADIUS group.
   a) Create a radius server-group identification.

> **Note**
>
> **server-group** refers to the server group name. The valid range is from one to 32 alphanumeric characters.

Example

```
Device(config)# aaa group server radius server-name
Device(config-sg-radius)# server name server-name
Device(config-sg-radius)# exit
```

6. Define a network-level authorization policy for access control using the previously specified server group.

   Example

   ```
   Device(config)# aaa authorization network rogued group group-name
   ```

7. Create an attribute list to apply custom authentication or authorization attributes.
   a) Create an attribute list to specify policies for rogue AP detection and control and enter the attribute list configuration mode.
   b) Identify the device with a specific rogue state for network policies.
   c) Identify the device with a specific rogue class for network policies.

   Example

   ```
   Device(config)# aaa attribute list aaa-attribute-list-name
   Device(config-attr-list)# attribute type rogue-ap-state {alert | contain |
   threat }
   Device(config-attr-list)# attribute type rogue-ap-class {unclassified |
   malicious | friendly }
   Device(config-attr-list)# exit
   ```

8. Define a client with a specific MAC address as a *username* and associate it with the previously defined AAA attribute list. This allows for controlled access or applies policy decisions for that specific client.

   Important

   For rogue validation, the *username* must be a MAC address in the semicolon format.

   Example

   Define a client with a specific MAC address (00:00:00:00:00:00) as a *username* and associate it with an AAA attribute list named **rogue-attributes**. This allows for the use of specific AAA (authentication, authorization, and accounting) attributes to be applied to this MAC address, typically to control access or apply policy decisions for that client.

   ```
   Device(config)# username 00:00:00:00:00:00 mac aaa attribute list
   aaa-attribute-list-name
   ```

9. Display the AAA running configuration for review.

Example

```
Device(config)# show running-config | include aaa
```

Setting up these configurations ensure that only authorized clients can access the network, and any suspicious activity is flagged and handled appropriately.

Configure authentication and authorization list for 802.1X security (GUI)

Use this procedure to configure authentication and authorization list for 802.1X security using the GUI.

Set up 802.1X security by specifying authentication and authorization lists through the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name** , the **SSID** , and the **WLAN ID**.
4. In the **Security** > **AAA** tab, select an authentication list from the **Authentication List** drop-down list.
5. Click **Apply to Device**.

Configure authentication and authorization list for 802.1X security (CLI)

Use this procedure to configure authentication and authorization list for 802.1X security using commands.

Configure 802.1X authentication and authorization on a WLAN using commands.

Use this procedure when you need to enforce 802.1X security for a specific wireless network on your device.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id ssid name
   ```

- *wlan-name* : is the name of the configured WLAN.
- *wlan-id* : is the wireless LAN identifier. The valid range is from 1 to 512.
- *SSID-name* : is the SSID name which can contain 32 alphanumeric characters.

> **Note**
>
> If you have already configured this command, enter **wlan** *wlan-name* command.

4. Enable authentication list for 802.1x security.

   Example

   ```
   Device(config-wlan)# security dot1x authentication-list authenticate-list-name
   ```

5. Specify authorization list for 802.1x security.

   Example

   ```
   Device(config-wlan)# security dot1x authorization-list authorize-list-name
   ```

   For more information on the Cisco Catalyst Center, see the Cisco Catalyst Center documentation.

6. Return to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

   Example

   ```
   Device(config-wlan)# end
   ```

Configure web authentication for WLAN with split authentication and authorization servers

This section explains how to configure web authentication for WLAN with split authentication and authorization servers on a wireless controller.

Configure authentication and authorization list for web authentication (GUI)

Use this procedure to configure authentication and authorization list for web authentication using GUI.

Configure authentication and authorization lists to secure web authentication on WLANs using the GUI.

This task ensures only authorized users can access the WLAN, and that access policies are applied during web authentication by assigning the correct authentication and authorization lists.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.

2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**, the **SSID**, and the **WLAN ID**.
4. In the **Security** > **Layer2** tab, uncheck the **WPAPolicy**, **AES** and **802.1x** check boxes.
5. Check the **MAC Filtering** check box to enable the feature. With MAC Filtering enabled, select the authorization list from the **Authorization List** drop-down list.
6. In the **Security** > **AAA** tab, select the authentication list from the **Authentication List** drop-down list.
7. Click **Apply to Device**.

Configure authentication and authorization list for web authentication (CLI)

Use this procedure to configure authentication and authorization list for web authentication using commands.

Enable web-based authentication and authorization for users connecting to a specified WLAN using commands.

Web authentication allows users to be authenticated through a web browser before gaining access to network resources.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id ssid-name
   ```

   - *wlan-name* : is the name of the configured WLAN.
   - *wlan-id* : is the wireless LAN identifier.
   - *SSID-name* : is the SSID name which can contain 32 alphanumeric characters.

   Note

   If you have already configured this command, enter **wlan** *wlan-name* command.

4. Disable WPA security.

Example

```
Device(config-wlan)# no security wpa
```

5. Disable security AKM for 802.1x.

Example

```
Device(config-wlan)# no security wpa akm dot1x
```

6. Disable WPA2 security.

Example

```
Device(config-wlan)# no security wpa wpa2
```

7. Enable authentication or authorization list for 802.1x security.

Example

```
Device(config-wlan)# security web-auth { authentication-list
authenticate-list-name | authorization-list authorize-list-name }
```

> **Note**
>
> When the WPA security, AKM for 802.1x, and WPA2 security settings are not disabled, the system displays the error:
>
> *% switch-1:dbm:wireless:web-auth cannot be enabled. Invalid WPA/WPA2 settings.*

8. Return to the privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Example

```
Device(config-wlan)# end
```

Verify split authentication and authorization configuration

This section explains how to verify split authentication and authorization configuration on a wireless controller.

To view the WLAN details, use this command:

```
Device# show run wlan

wlan wlan-foo 2 foo-ssid
security dot1x authentication-list authc-server-group
security dot1x authorization-list authz-server-group
wlan wlan-bar 3 bar-ssid
security web-auth authentication-list authc-server-group
security web-auth authorization-list authz-server-group
```

To view the AAA authentication and server details, use this command:

```
Device# show run aaa

!
aaa authentication dot1x default group radius
username cisco privilege 15 password 0 cisco
!
!
radius server free-radius-authc-server
 address ipv4 9.2.62.56 auth-port 1812 acct-port 1813
 key cisco
!
radius server cisco-catalyst-center-authz-server
 address ipv4 9.4.62.32 auth-port 1812 acct-port 1813
 pac key cisco
!
!
aaa new-model
aaa session-id common
!
```

To view the authentication and authorization list for 802.1X security, use this command:

```
Device# show wlan name wlan-foo | sec 802.1x

802.1x authentication list name               : authc-server-group
802.1x authorization list name                : authz-server-group
            802.1x                            : Enabled
```

To view the authentication and authorization list for web authentication, use this command:

```
Device# show wlan name wlan-bar | sec Webauth

Webauth On-mac-filter Failure             : Disabled
Webauth Authentication List Name          : authc-server-group
Webauth Authorization List Name           : authz-server-group
Webauth Parameter Map                     : Disabled
```

Configuration examples

This section provides sample configurations for authentication with a third-party RADIUS server.

Configure Cisco Catalyst 9800 Series Wireless Controller for authentication with a third-party RADIUS server: example

This example shows how to configure Cisco Catalyst 9800 Series Wireless Controller for authentication with a third-party RADIUS server:

```
Device(config)# radius server free-radius-authc-server

Device(config-radius-server)# address ipv4 9.2.62.56 auth-port 1812 acct-port
 1813

Device(config-radius-server)# key cisco

Device(config-radius-server)# exit

Device(config)# aaa group server radius authc-server-group

Device(config)# server name free-radius-authc-server

Device(config)# end
```

Configure Cisco Catalyst 9800 Series Wireless Controller for authorization with Cisco ISE or Cisco Catalyst Center: example

This example shows how to configure Cisco Catalyst 9800 Series Wireless Controller for authorization with Cisco ISE or Cisco Catalyst Center:

```
Device(config)# radius server cisco-catalyst-center-authz-server

Device (config-radius-server)# address ipv4 9.4.62.32 auth-port 1812 acct-port
 1813

Device (config-radius-server)# pac key cisco

Device (config-radius-server)# exit

Device(config)# aaa group server radius authz-server-group

Device(config)# server name cisco-catalyst-center-authz-server

Device(config)# end
```

RADIUS Message-Authenticator support for non-EAP requests

Introduces RADIUS Message-Authenticator support for non-EAP requests, explains sending and validating Message-Authenticator in request and response packets, outlines related configuration commands, and highlights protection against the Blast-RADIUS vulnerability (CVE-2024-3596).

A RADIUS Message-Authenticator configuration is a RADIUS server-group setting that:

- sends Message-Authenticator in every RADIUS request packet

- validates Message-Authenticator in every RADIUS response packet, and
- supports non-EAP RADIUS requests, including web authentication and management authentication.

Cisco IOS XE Release 17.18 provides the `send-ma` and `validate-ma` commands in RADIUS server-group configuration mode. Use these commands to configure Message-Authenticator handling for a RADIUS server group. The configuration helps protect against the Blast-RADIUS vulnerability (CVE-2024-3596).

Configure RADIUS Message-Authenticator support for non-EAP requests (CLI)

Provides instructions for configuring a RADIUS server group to send and validate Message-Authenticator attributes in every non-EAP request and response using CLI, supporting authentication methods such as web and management authentication.

Configure a RADIUS server group to send Message-Authenticator in every request and validate Message-Authenticator in every response.

Use this task when the RADIUS server supports Message-Authenticator responses for non-EAP authentication methods, such as web authentication and management authentication.

Before you begin

- Ensure that a RADIUS server group is available.
- Confirm that the RADIUS server supports Message-Authenticator responses for non-EAP authentication methods.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the RADIUS server group name and server group configuration.

   Example

   ```
   Device(config)# aaa group server radius radius-group-name
   ```

3. Send Message-Authenticator in every request packet.

   Example

   ```
   Device(config-sg-radius)# send-ma
   ```

4. Validate Message-Authenticator in every response packet.

   Example

   ```
   Device(config-sg-radius)# validate-ma
   ```

The device sends Message-Authenticator in every RADIUS request, including non-EAP requests, and validates Message-Authenticator in every RADIUS response for the configured RADIUS server group.

143 CUI Information in RADIUS Accounting

Topics:

- CUI information in RADIUS accounting request

Outlines the use and configuration of Chargeable User Identity information in RADIUS accounting requests, including authentication prerequisites, supported modes, and verification procedures for wireless controllers.

CUI information in RADIUS accounting request

This topic explains how Chargeable User Identity (CUI) is used as a unique identifier for clients in RADIUS accounting requests, and how it is processed and transmitted between the controller and AAA server.

Chargeable User Identity (CUI) is a unique identifier for a client visiting a network that

- is distinct from the outer identity or device used for login
- serves as an obscured version of a username, and
- can be used as an alternative for a client's username during authentication.

Feature History

Feature Name	Release	Description
Chargeable UserIdentity in RADIUS Accounting	Cisco IOS XE 17.9.1	Chargeable User Identity (CUI) is a unique identifier for a client visiting a network. This attribute can be used as an alternative for the client's username as part of the authentication process. The **access-session wireless cui-enable** command is introduced.

Processing of CUI in RADIUS accounting requests

To handle RADIUS attribute 89 processing, attach a null value of CUI to an access request sent to a AAA server.

Use the **access-session wireless cui-enable** command. A CUI-capable AAA server sends the CUI string to the controller as part of an access-accept message.

The controller then sends this received CUI attribute in accounting packets and other access-request packets, if any.

Prerequisites and restrictions

Use these guidelines when configuring AAA override. Understand the supported authentication protocols and controller modes to ensure proper operation.

Ensure that AAA override is enabled.

Restrictions

Adhere to these restrictions when configuring AAA override:

- Only 802.1x network authentication protocol is supported.
- Inter-Release Controller Mobility (IRCM) is not supported.
- FlexConnect local authentication is not supported. Only local mode and FlexConnect central authentication mode is supported.

Add CUI information in a RADIUS accounting request

Use this procedure to add CUI information in a RADIUS accounting request to enhance the data sent to the AAA server.

Enhance RADIUS accounting requests by including CUI information for better tracking and management.

This configuration is useful in environments where detailed user session information is required for auditing and reporting purposes.

Before you begin

Ensure that the RADIUS server is properly configured to handle CUI attributes.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Add CUI attribute in authentication and accounting messages sent to the AAA server.

   Example

   ```
   Device(config)# access-session wireless cui-enable
   ```

The CUI information is now included in RADIUS accounting requests, allowing for enhanced tracking and management of user sessions.

Verify CUI information in a RADIUS accounting request

Verify CUI information in a RADIUS accounting request on a wireless controller.

To view the CUI attribute in an accounting request on aAAA server, use this command:

```
Device# show wireless client mac-address aaa.bbb.ccc.ddd detail
.
.
.
Session Manager:
  Point of Attachment : capwap_90000005
  IIF ID              : 0x90000005
  Authorized          : TRUE
  Session timeout     : 1800
  Common Session ID: 8A45400A0000000CE0527C5F
  Acct Session ID  : 0x00000003
  Last Tried Aaa Server Details:
        Server IP : 10.64.69.141
  Auth Method Status List
        Method : Dot1x
                 SM State         : AUTHENTICATED
                 SM Bend State    : IDLE
  Local Policies:
        Service Template : wlan_svc_default-policy-profile_local (priority
254)
                 VLAN             : 59
                 Absolute-Timer   : 1800
  Server Policies:
                 CUI              : 13e158006855c2ff718cc84487653f5a6ea55def
  Resultant Policies:
                 CUI              : 13e158006855c2ff718cc84487653f5a6ea55def
                 VLAN Name        : VLAN0059
```

```
                VLAN              : 59
                Absolute-Timer    : 1800
```

144 Secure LDAP

Topics:

- Lightweight Directory Access Protocol (LDAP)

Lightweight Directory Access Protocol (LDAP)

Explains how Lightweight Directory Access Protocols (LDAPs) function as client-server protocols for directory information management and their use in secure authentication and authorization.

A Lightweight Directory Access Protocol (LDAP) is a client-server protocol that

- allows clients to access and manage directory information stored on servers
- supports user authentication and authorization using operations such as bind and search, and
- operates over TCP/IP on port 389, optionally using Transport Layer Security (TLS) for secure communications.

Transport Layer Security (TLS)

Transport Layer Security (TLS) is an application-level protocol that enables secure data transactions by providing privacy, authentication, and data integrity. TLS relies on certificates, public keys, and private keys to prove the identity of clients.

Certificate Authorities (CAs) issue the certificates.

Each certificate includes these:

- The name of the authority that issued it.
- The name of the entity to which the certificate was issued.
- The public key of the entity.
- The timestamps indicating the expiration date of the certificate.

TLS support for LDAP is described in RFC 2830, which is an extension to the LDAP protocol.

LDAP Operations

Bind

The bind operation authenticates a user to the server and starts a connection with the LDAP server. LDAP is a connection-oriented protocol. The client specifies the protocol version and authentication information.

LDAP supports these binds:

- Authenticated bind: An authenticated bind is performed when a root Distinguished Name (DN) and password are available.
- Anonymous bind: An anonymous bind is performed when no root DN and password are provided.

In LDAP deployments, the search operation is commonly performed before the bind operation. If a password attribute is returned as part of the search operation, the LDAP client can verify the password locally, eliminating the need for an extra bind operation. If the password attribute is not returned, the bind operation can occur later. Another advantage of performing the search operation first is that the LDAP client can use the DN received in the search result as the user DN, instead of constructing it by prefixing the username (cn attribute) with the base DN. All entries stored in an LDAP server have a unique DN.

The DN consists of two parts:

- Relative Distinguished Name (RDN)
- Location in the LDAP server where the record resides.

Most of the entries that you store in an LDAP server will have a name, and the name is frequently stored in the Common Name (cn) attribute. Because every object has a name, most objects stored in an LDAP server use their cn value as the basis for their RDN.

Search

A search operation is used to search the LDAP server. The client specifies the starting point (base DN) of the search, the search scope (either the object, its children, or the subtree rooted at the object), and a search filter.

For authorization requests, the search operation is performed without a bind operation. The LDAP server must be configured with sufficient privileges for the search operation to succeed. These privileges are established during the bind operation.

An LDAP search operation can return multiple entries for a specific user. When this occurs, the LDAP client returns an error code to AAA. To prevent these errors, configure search filters to match only a single entry.

Compare

The compare operation replaces a bind request for authentication and helps maintain the initial bind parameters for the connection.

LDAP Dynamic Attribute Mapping

The Lightweight Directory Access Protocol (LDAP) is a powerful and flexible protocol for communication with AAA servers. LDAP attribute maps provide a method to cross-reference attributes retrieved from a server with Cisco attributes supported by security appliances.

When a user authenticates to a security appliance, the appliance authenticates the server and uses LDAP to retrieve the user's record. The record consists of LDAP attributes associated with fields displayed on the server's user interface. Each attribute retrieved includes a value entered by the administrator responsible for updating user records.

SLDAP prerequisite

This section explains the prerequisites for configuring SLDAP on a wireless controller.

This section lists the prerequisite for setting up SLDAP on a wireless controller and describes when to configure certificates.

- If you are using a Transport Layer Security (TLS) secure connection, you must configure X.509 certificates.

SLDAP restrictions

This section explains the restrictions for configuring SLDAP on a wireless controller.

- LDAP referrals are not supported.
- The LDAP server’s unsolicited messages or notifications are not handled.
- LDAP authentication is not supported for interactive terminal sessions.

Configure SLDAP (CLI)

Use this procedure to configure SLDAP on a wireless controller.

Set up Secure LDAP (SLDAP) for authentication and directory queries on the wireless controller using commands.

Before you begin

Ensure you have administrator access to the controller. Gather LDAP server details (IP address, base DN, credentials).

Procedure

1. Enable privileged EXEC mode.

Example

```
Device# enable
```

Enter your password if prompted.

2. Enter the global configuration mode.

Example

```
Device# configure terminal
```

3. Define a Lightweight Directory Access Protocol (LDAP) server and enter the LDAP server configuration mode.

Example

```
Device(config)# ldap server server-name
```

4. Specify the LDAP server IP address using IPv4.

Example

```
Device(config-ldap-server)# ipv4 ipv4-address
```

5. Specify the number of seconds the Cisco Catalyst 9800 Series Wireless Controller or the embedded wireless controller waits for a reply to a LDAP request before retransmitting the request.

Example

```
Device(config-ldap-server)# timeout retransmit seconds
```

6. Specify a shared secret text string used between the Cisco Catalyst 9800 Series Wireless Controller or the embedded wireless controller and a LDAP server.

Example

```
Device(config-ldap-server)# bind authenticate root-dn
CN=ldapipv6user,CN=Users,DC=ca,DC=ssh2,DC=com password Cisco12345
```

Use the **0** line option to configure an unencrypted shared secret.

Use the **7** line option to configure an encrypted shared secret.

7. Specify the base Distinguished Name (DN) of the search.

Example

```
Device(config-ldap-server)# base-dn string CN=Users,DC=ca,DC=ssh2,DC=com
```

8. Configure LDAP to initiate the TLS connection and specify the secure mode.

Example

```
Device(config-ldap-server)# mode secure no- negotiation
```

9. Return to the privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Example

```
Device(config-ldap-server)# end
```

Configure a AAA server group (GUI)

Use this procedure to configure a AAA server group using a GUI.

Configure a AAA server group using the GUI for RADIUS, TACACS+, or LDAP server hosts using the GUI.

Configuring a device to use AAA server groups helps you group existing server hosts, select a subset of those hosts, and assign them for a particular service. A server group is used with a global server-host list. The server group lists the IP addresses of the selected server hosts.

You can create these server groups:

Before you begin

Procedure

1. RADIUS
   a) Choose **Services** > **Security** > **AAA** > **Server Groups** > **RADIUS**.
   b) Click the **Add** button. The **Create AAA Radius Server Group** dialog box appears.
   c) Enter a name for the RADIUS server group in the **Name** field.
   d) Select a desired delimiter from the **MAC-Delimiter** drop-down list. The available options are colon, hyphen, and single-hyphen.
   e) Select a desired filter from the **MAC-Filtering** drop-down list. The available options are mac and Key.
   f) Enter a value in the **Dead-Time (mins)** field to make a server non-operational. You must specify a value between 1 and 1440.
   g) Choose any of the available servers from the **Available Servers** list and move them to the **Assigned Servers** list by clicking the **>** button.
   h) Click the **Save & Apply to Device** button.

2. TACACS+
   a) Choose **Services** > **Security** > **AAA** > **Server Groups** > **TACACS+**.
   b) Click the **Add** button. The **Create AAA Tacacs Server Group** dialog box appears.
   c) Enter a name for the TACACS server group in the **Name** field.
   d) Choose any of the available servers from the **Available Servers** list and move them to the **Assigned Servers** list by clicking the **>** button.
   e) Click the **Save & Apply to Device** button.

3. LDAP
   a) Choose **Services** > **Security** > **AAA** > **Server Groups** > **LDAP**.
   b) Click the **Add** button. The **Create AAA Ldap Server Group** dialog box appears.
   c) Enter a name for the LDAP server group in the **Name** field.
   d) Choose any of the available servers from the **Available Servers** list and move them to the **Assigned Servers** list by clicking the **>** button.
   e) Click the **Save & Apply to Device** button.

Configure a AAA server group (CLI)

Use this procedure to configure a AAA server group using commands.

Establish a AAA server group for centralized authentication, authorization, and accounting using commands.

AAA server groups allow you to centrally manage user access and policy enforcement with protocols such as LDAP, RADIUS, or TACACS+.

Procedure

1. Enable the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Enable AAA.

   Example

   ```
   Device(config)# aaa new-model
   ```

4. Define the AAA server group with a group name and enter the LDAP server group configuration mode.

   Example

   ```
   Device(config)# aaa group server ldap group-name
   ```

   All members of a group are of the same type: RADIUS, LDAP, or TACACS+.

5. Associate a particular LDAP server with the defined server group.

   Example

   ```
   Device(config-ldap-sg)# server server-name
   ```

   Each security server is identified by its IP address and UDP port number.

6. Exit the LDAP server group configuration mode.

   Example

   ```
   Device(config-ldap-sg)# exit
   ```

Configure search and bind operations for authentication requests (CLI)

Use this procedure to configure search and bind operations for an authentication request on a wireless controller.

Enable LDAP-based authentication by defining how the controller searches and binds user credentials during authentication requests using commands.

Configuring search and bind operations determines whether the LDAP server first binds the user and then performs a search, or compares credentials without binding. This affects how authentication requests are processed on the controller.

Procedure

1. Enable the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Enable AAA.

   Example

   ```
   Device(config)# aaa new-model
   ```

4. Define a Lightweight Directory Access Protocol (LDAP) server and enter the LDAP server configuration mode.

   Example

   ```
   Device(config)# ldap server server-name
   ```

5. Configure the sequence of search and bind operations for an authentication request.

   Example

   ```
   Device(config-ldap-server)# authentication bind-first
   ```

6. Replace the bind request with the compare request for authentication.

   Example

   ```
   Device(config-ldap-server)# authentication compare
   ```

7. Exit the LDAP server group configuration mode.

Example

```
Device(config-ldap-server)# exit
```

Configure a dynamic attribute map on a SLDAP server (CLI)

Use this procedure to configure a dynamic attribute map on a SLDAP server on a wireless controller.

Configure dynamic LDAP attribute maps to ensure seamless interoperability between user-defined attribute names and Cisco attribute requirements on a wireless controller using commands.

You must create LDAP attribute maps that map your existing user-defined attribute names and values to Cisco attribute names and values that are compatible with the security appliance. You can then bind these attribute maps to LDAP servers or remove them as required.

> **Note**
>
> To use the attribute mapping features correctly, make sure you understand Cisco LDAP and user-defined attribute names and values.

Before you begin

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Configure a dynamic LDAP attribute map and enter the attribute-map configuration mode.

   Example

   ```
   Device(config)# ldap attribute-map map-name
   ```

4. Define an attribute map.

   Example

   ```
   Device(config-attr-map)# map type ldap-attr-type aaa-attr-type
   ```

5. Exit the attribute-map configuration mode.

Example

```
Device(config-attr-map)# exit
```

Verify the SLDAP configuration

This section provides the show commands to verify the secure LDAP configuration.

To view details about the default LDAP attribute mapping, use this command:

```
Device# show ldap attributes
```

To view the LDAP server state information and various other counters for the server, use this command:

```
Device# show ldap server
```

145 Network Access Server Identifier

Topics:

- Network Access Server Identifier (NAS-ID)

Network Access Server Identifier (NAS-ID)

Explains the Network Access Server Identifier (NAS-ID) and how it is used to inform the RADIUS server of a RADIUS access request, enabling it to select a policy for that request.

A network access server identifier (NAS-ID) is a RADIUS attribute that

- identifies the source device, WLAN profile, VLAN interface, or AP group initiating a RADIUS access request
- enables the RADIUS server to apply appropriate authentication and policy selection, and
- supports customized policy enforcement based on user groups or device types.

The NAS-ID is sent to the RADIUS server by the controller through an authentication request to classify users to different groups. This enables the RADIUS server to send a customized authentication response.

Starting with Cisco IOS XE Cupertino 17.7.1, a new string named custom-string (custom string) is available.

> **Note**
>
> The acct-session-id is sent with the RADIUS access request only if accounting is enabled on the policy profile.

If you configure a NAS-ID for an AP group, it overrides the NAS-ID that is configured for a WLAN profile or the VLAN interface. Similarly, if you configure a NAS-ID for a WLAN profile, it overrides the NAS-ID that is configured for the VLAN interface.

These options can be configured for a NAS ID:

- sys-name (System Name)
- sys-ip (System IP Address) and sys-mac (System MAC Address)
- ap-ip (AP's IP address) and ap-name (AP's Name)
- ap-mac (AP's MAC Address)
- ap-eth-mac (AP's Ethernet MAC Address)
- ap-policy-tag (AP's policy tag name)
- ap-site-tag (AP's site tag name)
- ssid (SSID Name) and ap-location (AP's Location)
- custom-string (custom string)

Create a NAS ID policy (GUI)

Use this procedure to create a NAS ID policy using GUI.

Configure a NAS ID policy that defines how network access server identifiers are applied within your wireless AAA policy using the GUI.

Creating a NAS ID policy ensures correct identification and handling of wireless authentication requests. This is typically performed when setting up or modifying wireless AAA policies.

Procedure

1. Choose **Configuration** > **Security** > **Wireless AAA Policy**.
2. On the **Wireless AAA Policy** page, click the name of the **Policy** or click **Add** to create a new one.
3. In the **Add/Edit Wireless AAA Policy** window that is displayed, enter the name of the policy in the **Policy Name** field.
4. Select from one of the NAS ID options from the **Option 1** drop-down list.
5. Select from one of the NAS ID options from the **Option 2** drop-down list.
6. Select from one of the NAS ID options from the **Option 3** drop-down list.
7. Save the configuration.

Create a NAS ID policy (CLI)

Use this procedure to create a NAS ID policy on a wireless controller.

Create and configure a NAS ID policy for wireless RADIUS authentication and accounting using commands.

Follow the procedure given below to create NAS ID policy:

Before you begin

- A NAS ID can combine up to a maximum of three options.
- The maximum length of the NAS ID attribute is 253. Before adding a new attribute, the system checks the attribute buffer. If there is not enough space, the system ignores the new attribute.
- By default, a wireless aaa policy (default-aaa-policy) is created with the default configuration (sys-name). You can update this policy with various NAS ID options. However, the default-aaa-policy cannot be deleted.
- If you do not configure a NAS ID, the system uses the default sys-name as the NAS ID for all wireless-specific RADIUS packets (authentication and accounting) from the controller .
- Starting with Cisco IOS XE Cupertino 17.7.1, you can configure a custom NAS ID string using combinations of option1, option2, and option3 (**nas-id option3custom-string***custom-string*) for RADIUS packets.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a new AAA policy.

   Example

   ```
   Device(config)# wireless aaa policy policy-name
   ```

3. Configure NAS ID for option1.

Example

```
Device(config-aaa-policy)# nas-id option1 sys-name
```

4. Configure NAS ID for option2.

Example

```
Device(config-aaa-policy)# nas-id option2 sys-ip
```

5. Configure NAS ID for option3.

Example

```
Device(config-aaa-policy)# nas-id option3 sys-mac
```

Attach a policy to a tag (GUI)

Use this procedure to attach a policy to a tag using GUI.

Map a policy to a tag for consistent network device configuration using the GUI.

Attach a predefined policy to a tag to apply specific settings or profiles across selected devices.

Before you begin

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click **Add** and enter a name for the new policy, for example, test1, in the **General** tab.
3. Click the **Advanced** tab, and under **AAA Policy**, from the **Policy Name** drop-down list, select the policy name that you had created in the **General** tab.
4. Click **Apply to Device**.
5. Choose **Configuration** > **Tags & Profiles** > **Tags** > **Policy**.
6. Click **Add** to view the **Add Policy Tag** window.
7. Enter a name and description for the policy tag.
8. Click **Add** to map WLAN profile and policy profile.
9. Choose the **WLAN Profile** to map with the appropriate **Policy Profile**, and click the tick icon. Click **Save & Apply to Device**.

Attach a policy to a tag (CLI)

Use this procedure to attach a policy to a tag using commands.

Configure and attach a NAS ID policy to a wireless policy tag using commands.

Follow the procedure given below to attach a NAS ID policy to a tag:

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a WLAN policy profile.

   Example

   ```
   Device(config)# wireless profile policy policy-name
   ```

3. Configure a AAA policy profile.

   Example

   ```
   Device(config-wireless-policy)# aaa-policy aaa-policy-name
   ```

4. Return to the global configuration mode.

   Example

   ```
   Device(config-wireless-policy)# exit
   ```

5. Configure a wireless policy tag.

   Example

   ```
   Device(config)# wireless tag policy policy-tag
   ```

6. Map a WLAN profile to a policy profile.

   Example

   ```
   Device(config)# wlan wlan1 policy policy-name
   ```

   Note

   You can also use the **ap-tag** option to configure a NAS ID for an AP group. This NAS ID overrides the NAS ID configured for a WLAN profile or VLAN interface.

Verify the NAS ID configuration

This section provides the show commands to verify NAS ID configuration.

Use this **show** command to verify the NAS ID configuration:

```
Device# show wireless profile policy detailed test1

Policy Profile Name           : test1
Description                   :
Status                        : ENABLED
VLAN                          : 1
Client count                  : 0

:
:
AAA Policy Params
  AAA Override                : DISABLED
  NAC                         : DISABLED
  AAA Policy name             : test
```

146 Locally Significant Certificates

Topics:

- Locally significant certificates
- Provisioning locally significant certificates
- AP certificate renewal (LSC)
- MIC and LSC APs joining the controller
- LSC fallback APs
- Configuring controller self-signed certificate for wireless AP join

Outlines the concepts, provisioning workflows, renewal processes, controller integration, fallback mechanisms, and configuration procedures for Locally Significant Certificates (LSC) in wireless deployments, guiding users through AP certificate management, controller settings, and related operational tasks.

Locally significant certificates

Explains how to configure a wireless controller and LAP to use the locally significant certificate (LSC).

A locally significant certificate (LSC) is a PKI certificate that

- provides mutual authentication between Cisco Catalyst 9800 Series Wireless Controller and Lightweight Access Points (LAPs)
- generates certificates directly on APs and controllers, and
- enables organizations to use their own PKI for better security control and Certificate Authority (CA) management.

LSC provisioning and communication process

You need to provision the new LSC certificate on the controller and then the Lightweight Access Point (LAP) from the CA Server.

The LAP communicates with the controller using the CAPWAP protocol. Any request to sign the certificate and issue the CA certificates for LAP and controller itself must be initiated from the controller. The LAP does not communicate directly with the CA server. The CA server details must be configured on the controller and must be accessible.

The controller makes use of the Simple Certificate Enrollment Protocol (SCEP) to forward certReqs generated on the devices to the CA and makes use of SCEP again to get the signed certificates from the CA.

The SCEP is a certificate management protocol that the PKI clients and CA servers use to support certificate enrollment and revocation. It is widely used in Cisco and supported by many CA servers. In SCEP, HTTP is used as the transport protocol for the PKI messages. The primary goal of SCEP is the secure issuance of certificates to network devices. SCEP is capable of many operations, but for our release, SCEP is utilized for the following operations:

- CA and Router Advertisement (RA) Public Key Distribution
- Certificate Enrollment

Certificate provisioning in controllers

Explains the provisioning of new LSC certificates (CA and device) on the controller.

Certificate provisioning in controllers is a security process that you use to:

- Install new Locally Significant Certificate (LSC) certificates (both Certificate Authority (CA) certificates and device certificates) on the controller
- Use Simple Certificate Enrollment Protocol (SCEP) to receive Certificate Authority (CA) certificates from the CA server during initial setup when the controller has no certificates, and
- Push the same Certificate Authority (CA) certificates to Access Points (APs) when you provision the Access Points (APs) with Locally Significant Certificates (LSCs).

RSA keypair recommendations

These recommendations apply when configuring PKI certificates with RSA keypairs:

Note

We recommend that you use a new RSA keypair name for the newly configured PKI certificate. If you want to reuse an existing RSA keypair name (that is associated with an old certificate) for a new PKI certificate, do either of the following:

- Do not regenerate a new RSA keypair with an existing RSA keypair name. Instead, reuse the existing RSA keypair name. All certificates associated with the existing RSA keypair become invalid when you regenerate a new RSA keypair with an existing name.
- Remove the old PKI certificate configurations before reusing the existing RSA keypair name for the new PKI certificate.

How device certificate enrollment works

Learn how to enroll device certificates through CA-signed certificate requests using PKCS#10 and SCEP protocols.

When your Lightweight Access Point (LAP) and controller request Certificate Authority (CA)-signed certificates, they send the certRequest as a PKCS#10 message. The Certificate Authority (CA) receives a PKCS#10 certRequest. The Certificate Authority (CA) requires additional information to authenticate the identity of the requester and verify that the request is unaltered.

Summary

The key components involved in the device certificate enrollment process are:

- Lightweight Access Point (LAP) and controller: request Certificate Authority (CA)-signed certificates by sending PKCS#10 messages. These messages contain Subject Name, Public Key, and other attributes.
- Certificate Authority (CA): transforms the certRequest into an X.509 certificate after authentication and verification
- PKCS#10 certRequest: contains digitally signed certificate request information including Subject Name and Public Key
- Simple Certificate Enrollment Protocol (SCEP) client: provides functionality to wrap PKCS#10 in PKCS#7 Signed Data message type

Workflow

Your device certificate enrollment process works through these stages:

1. The Lightweight Access Point (LAP) and controller create a PKCS#10 certificate request containing the Subject Name, Public Key, and other attributes to be included in the X.509 certificate, and digitally sign it with the Private Key of the requester.
2. The PKCS#10 is wrapped in a PKCS#7 Signed Data message type as part of the Simple Certificate Enrollment Protocol (SCEP) client functionality, and the PKCSReq message is sent to the controller.
3. The Certificate Authority (CA) receives the PKCS#10 certRequest and authenticates the identity of the requester and verifies that the request is unaltered.
4. The Certificate Authority (CA) transforms the certRequest into an X.509 certificate.

Result

Upon successful enrollment operation, both the Certificate Authority (CA) and device certificates are available on the controller.

How certificate provisioning works on lightweight APs

Describes how lightweight APs provision new certificates through CAPWAP mode using controller-assisted certificate requests and automated installation processes.

Certificate provisioning enables lightweight APs (LAPs) to obtain new signed X.509 certificates while operating in CAPWAP mode. This process ensures secure communications between APs and controllers using updated security credentials.

Summary

The key components involved in certificate provisioning on lightweight APs are:

- Lightweight Access Point (LAP): Initiates certificate requests and receives new certificates while in CAPWAP mode
- Controller: Acts as a CA proxy to help obtain signed certificates from the Certificate Authority for the LAP
- Certificate Authority (CA): Signs and validates certificate requests to issue new X.509 certificates
- LWAPP payloads: Transport mechanism for certificate requests and responses between LAP and controller

Workflow

These stages describe how certificate provisioning works on lightweight APs:

1. The LAP sends a certificate request to the controller to obtain a new signed X.509 certificate.

   The controller acts as a CA proxy and helps obtain the certificate request signed by the CA for the LAP.
2. The certificate requests and certificate responses are transmitted between the LAP and controller using LWAPP payloads.
3. Both the LSC CA and LAP device certificates are installed in the LAP, and the system reboots automatically.
4. When the system comes up after reboot, the AP sends the LSC device certificate to the controller as part of the JOIN Request because it is configured to use LSCs.

   As part of the JOIN Response, the controller sends the new device certificate and validates the inbound LAP certificate with the new CA root certificate.

What's next

To configure, authorize, and manage certificate enrollment with the existing PKI infrastructure for controller and AP, you need to use the LSC provisioning functionality.

Restrictions for locally significant certificates

Lists restrictions for locally significant certificates.

- The locally significant certificate (LSC) workflow differs in FIPS and WLANCC mode. Your Certificate Authority (CA) server must support Enrollment over Secure Transport (EST) protocol. Your CA server must also issue EC certificates in FIPS and WLANCC mode.
- The Elliptic Curve Digital Signature Algorithm (ECDSA) cipher works only if both your AP and controller have EC certificates provisioned with LSC.
- EC certificates (LSC-EC) can be provisioned only if CA server supports EST (and not SCEP).
- You must configure FIPS and CC security modes to provision EC certificates.
- Correct all AP misconfigurations before you enable LSC. You can view the count for misconfigured APs in the output of this **show** command:

```
Device# show wireless summary
Priming controller         : DISABLED
```

```
Max APs supported          : 3000
Max clients supported      : 32000
Access Point Summary
                       Total    Up    Down
-----------------------------------------
802.11 2.4GHz              2     2       0
802.11 5GHz                5     2       3
802.11 6GHz                1     1       0
802.11 dual-band           2     0       2
802.11 dual-band(5/6GHz)   0     0       0
802.11 rx-dual-band        0     0       0
Client Serving(2.4GHz)     3     1       2
Client Serving(5GHz)       4     1       3
Client Serving(6GHz)       1     1       0
Monitor(Dual band)         0     0       0
Monitor(2.4GHz)            1     1       0
Monitor(5GHz)              1     1       0
Monitor(6GHz)              0     0       0
Sniffer(Dual band)         0     0       0
Sniffer(2.4GHz)            0     0       0
Sniffer(5GHz)              0     0       0
Sniffer(6GHz)              0     0       0
Misconfigured APs          1 (For more info use 'show ap tag summary')
```

For more information about misconfigured APs, run the **wireless config validate** command. To view reported errors, run the **show wireless config validation status** command.

Provisioning locally significant certificates

Details the configuration, provisioning, management, and verification of locally significant certificates (LSCs), including RSA key and PKI trustpoint setup, certificate enrollment, AP LSC provisioning, trustpool management, and secure wireless network authentication using both GUI and CLI procedures.

Configure RSA key for PKI trustpoint

Configure RSA key for PKI trustpoint on a wireless controller.

Generate RSA cryptographic keys for PKI trustpoint authentication and secure communications.

RSA keys are essential components of PKI infrastructure, providing the cryptographic foundation for certificate-based authentication and encryption on wireless controllers.

Before you begin

Ensure the device is in a stable operational state and has sufficient processing resources for key generation.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Generate RSA key for PKI trustpoint.

Example

```
Device(config)# crypto key generate rsa exportable general-keys modulus
key-size label RSA-key-label
```

Example

```
Device(config)# crypto key generate rsa exportable
general-keys modulus 2048 label lsc-tp
```

exportable is an optional keyword. You may or may not want to configure an exportable-key. If selected, you can export the key out of the box, if required.

- *key-size*: Size of the key modulus. The valid range is from 2048 to 4096.
- *RSA-key-label*: RSA key pair label.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

The RSA key is successfully generated and labeled for use with PKI trustpoint configurations.

Configure PKI trustpoint parameters

Configure RSA key for PKI trustpoint.

Configure a PKI trustpoint with RSA key parameters for certificate enrollment and authentication.

PKI trustpoints define the relationship between the device and a Certificate Authority (CA), enabling secure certificate-based authentication and encryption.

Before you begin

Ensure the CA server is accessible and configured to accept certificate requests.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Create a new trustpoint for an external CA server.

Example

```
Device(config)# crypto pki trustpoint trustpoint-name
```

Example

```
Device(config)# crypto pki trustpoint microsoft-ca
```

The *trustpoint-name* refers to the trustpoint name.

3. Specify the URL of the CA on which your router should send certificate requests.

Example

```
Device(ca-trustpoint)# enrollment url HTTP-URL
```

Example

```
Device(ca-trustpoint)# enrollment url http://CA_server/certsrv/mscep/mscep.dll
```

HTTP-URL: URL of the file system where your router should send certificate requests. An IPv6 address can be added in the URL enclosed in brackets. For example: http://[2001:DB8:1:1::1]:80. For more enrollment method options, see the enrollment url (ca-trustpoint) command page.

4. Create subject name parameters for the trustpoint.

Example

```
Device(ca-trustpoint)# subject-name subject-name
```

Example

```
Device(ca-trustpoint)# subject-name C=IN, ST=KA, L=Bengaluru, O=Cisco,
CN=eagle-eye/emailAddress=support@abc.com
```

5. Map RSA key with that of the trustpoint.

Example

```
Device(ca-trustpoint)# rsakeypair RSA-key key-size
```

Example

```
Device(ca-trustpoint)# rsakeypair ewlc-tp1
```

- *RSA-key*: RSA key pair label.
- *key-size*: Signature key length. Range is from 360 to 4096.

6. Configure revocation checking method.

Example

```
Device(ca-trustpoint)# revocation {crl | none | ocsp}
```

Example

```
Device(ca-trustpoint)# revocation none
```

7. Return to privileged EXEC mode.

Example

```
Device(ca-trustpoint)# end
```

The PKI trustpoint is now configured with the specified parameters and ready for certificate enrollment operations.

Authenticate and enroll a Public Key Infrastructure (PKI) trustpoint (GUI)

Use this procedure to authenticate and enroll a Public Key Infrastructure (PKI) trustpoint through the device's graphical user interface.

Use this procedure to add and authenticate a Public Key Infrastructure (PKI) trustpoint to enable certificate-based security for your device.

Before you begin

Obtain the enrollment URL and certificate authority information from your network administrator.

Procedure

---

1. Choose **Configuration** > **Security** > **Public Key Infrastructure (PKI) Management**.
2. In the **Public Key Infrastructure (PKI) Management** window, click the **Trustpoints** tab.
3. In the **Add Trustpoint** dialog box, provide this information:
   a) In the **Label** field, enter the RSA key label.
   b) In the **Enrollment URL** field, enter the enrollment URL.
   c) Check the **Authenticate** check box to authenticate the Public Certificate from the enrollment URL.
   d) In the **Subject Name** section, enter the **Country Code**, **State**, **Location**, **Organization**, **Domain Name**, and **Email Address**.
   e) Check the **Key Generated** check box to view the available RSA keypairs. Choose an option from the **Available RSA Keypairs** drop-down list.
   f) Check the **Enroll Trustpoint** check box.
   g) In the **Password** field, enter the password.
   h) In the **Re-Enter Password** field, confirm the password.
   i) Click **Apply to Device**.

   The new trustpoint is added to the trustpoint name list.

The new trustpoint is now listed in the trustpoint name list and is ready for use.

What to do next

After enrolling the trustpoint, configure it for use with your security policies or certificate-based authentication features.

Authenticate and enroll the PKI trustpoint with CA server (CLI)

Configure authentication and enrollment of a PKI trustpoint with CA server using commands.

Establish secure communication between the device and Certificate Authority by authenticating and enrolling a PKI trustpoint.

PKI trustpoints enable secure certificate-based authentication. The authentication process fetches the CA certificate, while enrollment generates and submits a certificate request to obtain a client certificate.

Before you begin

Ensure the PKI trustpoint is configured and the CA server is accessible.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Fetch the CA certificate.

   Example

   ```
   Device(config)# crypto pki authenticate trustpoint-name
   ```

   Example

   ```
   Device(config)# crypto pki authenticate microsoft-ca
   ```

3. Accept the CA certificate.

   Example

   ```
   Device(config)# % Do you accept this certificate? [yes/no]:
   yes Trustpoint CA certificate accepted.
   ```

4. Enroll the client certificate.

   Example

   ```
   Device(config)# crypto pki enroll trustpoint-name
   ```

   Example

   ```
   Device(config)# crypto pki enroll microsoft-ca
   %
   % Start certificate enrollment ..
   % Create a challenge password. You will need to verbally
   provide this password to the CA Administrator in order to
   revoke your certificate. For security reasons your password
   will not be saved in the configuration.
   Please make a note of it.
   ```

5. Enter a challenge password to the CA server.

Example

```
Device(config)# password
```

Example

```
Device(config)# abcd123
```

6. Re-enter the challenge password to the CA server.

Example

```
Device(config)# password
```

Example

```
Device(config)# abcd123
```

7. Include the router serial number in the subject name.

Example

```
Device(config)# % Include the router serial number
in the subject name? [yes/no]: yes
```

8. Exclude IP address from the subject name.

Example

```
Device(config)# % Include an IP address
in the subject name? [no]: no
```

9. Request certificate from CA.

Example

```
Device(config)#
Request certificate from CA? [yes/no]: yes
% Certificate request sent to Certificate Authority
% The 'show crypto pki certificate verbose
client' command will show the fingerprint.
```

10. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

The PKI trustpoint is authenticated and enrolled with the CA server, enabling secure certificate-based communication.

Configure AP join attempts with LSC certificate

Learn how to configure AP join attempts with LSC certificate using the GUI.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. In the **All Access Points** window, click the LSC Provision name.
3. From the **Status** drop-down list, choose a status to enable LSC.
4. From the **Trustpoint Name** drop-down list, choose the trustpoint.
5. In the **Number of Join Attempts** field, enter the number of retry attempts that will be permitted.
6. Click **Apply**.

Configure AP join attempts with LSC certificate (CLI)

Configure AP join attempts with LSC certificate using commands.

Configure the maximum number of AP join failure attempts with a newly provisioned LSC certificate to ensure proper fallback to MIC when needed.

When APs attempt to join with LSC certificates and exceed the configured failure limit, they automatically fall back to using their Manufacturer Installed Certificate (MIC) for authentication.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the maximum number of AP join failure attempts with the newly provisioned LSC certificate.

   Example

   ```
   Device(config)# ap lsc-provision join-attempt number-of-attempts
   ```

   Example

   ```
   Device(config)# ap lsc-provision join-attempt 10
   ```

   When the number of AP joins exceed the specified limit, AP joins back with the Manufacturer Installed Certificate (MIC).

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The AP join attempt limit with LSC certificate is now configured. APs will fall back to MIC authentication after exceeding the specified number of failed join attempts.

Configure subject-name parameters in LSC certificate

Configure subject-name parameters in LSC certificate.

Configure the subject-name parameters that will be included in LSC certificates generated by access points.

Subject-name parameters define the identity information included in Local Significant Certificate (LSC) requests generated by access points. These parameters include geographical and organizational details that identify the certificate holder.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the attributes to be included in the subject-name parameter of the certificate request generated by an AP.

   Example

   ```
   Device(config)# ap lsc-provision subject-name-parameter
   country country-str state state-str city city-str domain domain-str org
   org-str email-address email-addr-str
   ```

   Example

   ```
   Device(config)# ap lsc-provision subject-name-parameter
   country India state Karnataka city Bangalore domain domain1 org
   Right email-address adc@gfe.com
   ```

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The subject-name parameters are now configured for LSC certificate generation on access points.

Configure key size for LSC certificate

Configure key size for LSC certificate.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Specify the size of keys to be generated for the LSC on AP.

Example

```
Device(config)# ap lsc-provision key-size key-size
```

Example

```
Device(config)# ap lsc-provision key-size 2048
```

Valid key size options are 2048, 3072, or 4096 bits.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Configure trustpoint for LSC provisioning on an AP

Configure trustpoint for LSC provisioning on an access point.

Establish a trustpoint for Local Significant Certificate (LSC) provisioning to enable secure communication between access points and the controller.

LSC provisioning allows access points to obtain certificates from a designated trustpoint, enhancing network security through certificate-based authentication.

Before you begin

Ensure the trustpoint is properly configured and accessible on the network.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Specify the trustpoint with which the LSC is provisioned to an AP.

Example

```
Device(config)# ap lsc-provision trustpoint tp-name
```

Example

```
Device(config)# ap lsc-provision trustpoint microsoft-ca
```

The *tp-name* is the trustpoint name.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

The trustpoint for LSC provisioning is now configured and ready to provision certificates to access points.

Configure an AP LSC provision list (GUI)

Configure an AP LSC provision list using GUI.

Procedure

---

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. In the **All Access Points** window, click the corresponding LSC Provision name.
3. From the **Status** drop-down list, choose a status to enable LSC.
4. From the **Trustpoint Name** drop-down list, choose a trustpoint.
5. In the **Number of Join Attempts** field, enter the number of retry attempts that are allowed.
6. From the **Key Size** drop-down list, choose a key.
7. In the **Edit AP Join Profile** window, click the **CAPWAP** tab.
8. In the **Add APs to LSC Provision List** section, click **Select File** to upload the CSV file that contains AP details.
9. Click **Upload File**.
10. In the **AP MAC Address** field, enter the AP MAC address. and add them. (The APs added to the provision list are displayed in the **APs in provision List** .)
11. In the **Subject Name Parameters** section, enter the following details:
    - **Country**
    - **State**
    - **City**
    - **Organization**
    - **Department**
    - **Email Address**
12. Click **Apply**.

Configure an AP LSC provision list (CLI)

Configure an AP LSC provision list using commands.

Add access points to the LSC provision list for certificate-based authentication.

The LSC provision list allows you to specify which access points can use locally significant certificates for authentication with the wireless controller.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Add the AP to the LSC provision list.

   Example

   ```
   Device(config)# ap lsc-provision mac-address mac-address
   ```

   Example

   ```
   Device(config)# ap lsc-provision
   mac-address 001b.3400.02f0
   ```

   Note

   You can provision a list of APs using the **ap lsc-provision provision-list** command.

   (Or)

   You can provision all the APs using the **ap lsc-provision** command.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The AP is now added to the LSC provision list and can use locally significant certificates for authentication.

Configure LSC provisioning for all the APs (GUI)

Configure LSC provisioning for all the APs using GUI.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. In the **Access Points** window, expand the **LSC Provision** section.
3. Set **Status** to **Enabled** state.

> **Note**
>
> If you set **Status** to **Provision List**, LSC provisioning will be configured only for APs that are a part of the provision list.

4. From the **Trustpoint Name** drop-down list, choose the appropriate trustpoint for all APs.
5. In the **Number of Join Attempts** field, enter the number of retry attempts that the APs can make to join the controller.
6. From the **Key Size** drop-down list, choose the appropriate key size of the certificate:
   - **2048**
   - **3072**
   - **4096**
7. In the **Add APs to LSC Provision List** section, click **Select File** to upload the CSV file that contains the AP details.
8. Click **Upload File**.
9. In the **AP MAC Address** field, enter the AP MAC address. (The APs that are added to the provision list are displayed in the **APs in Provision List** section.)
10. In the **Subject Name Parameters** section, enter the following details:
    a. **Country**
    b. **State**
    c. **City**
    d. **Organization**
    e. **Department**
    f. **Email Address**
11. Click **Apply**.

Configure LSC provisioning for all APs (CLI)

Configure LSC provisioning for all APs using commands.

Enable LSC (Local Significant Certificate) provisioning across all access points to enhance security through automatic certificate management.

LSC provisioning allows access points to automatically obtain and manage security certificates, reducing manual certificate deployment overhead while maintaining secure wireless communications.

Before you begin

Ensure you have administrative access to the wireless controller and that all APs are properly registered with the controller.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enable LSC provisioning for all APs.

   Example

   ```
   Device(config)# ap lsc-provision
   ```

   By default, LSC provisioning is disabled for all APs.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

LSC provisioning is now enabled for all access points, allowing them to automatically obtain and manage security certificates.

Configure LSC provisioning for the APs in the provision list

Configure LSC provisioning for access points in provision list.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable LSC provisioning for a set of APs configured in the provision list.

   Example

   ```
   Device(config)# ap lsc-provision provision-list
   ```

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Import a CA certificate to the trustpool (GUI)

Configure CA certificate import to the trustpool using GUI.

PKI Trustpool Management is used to store a list of trusted certificates (either downloaded or built in) used by the different services on the controller. This is also used to authenticate a multilevel CA certificate. The built in CA certificate bundle in the PKI trustpool receives automatic updates from Cisco if they are not current, are corrupt, or if certain certificates need to be updated.

Perform this task to manually update the CA certificates in the PKI trustpool.

> **Note**
>
> If your LSC has been issued by an intermediate CA, you must import the complete chain of CA certificates into the trustpool. Otherwise, you will not be able to provision the APs without the complete chain being present on the controller. The import step is not required if the certificate has been issued by a root CA.

Follow these steps to import a CA certificate to the trustpool using GUI:

Procedure

1. Choose **Configuration** > **Security** > **PKI Management**.
2. In the **PKI Management** window, click the **Trustpool** tab.
3. Click **Import**.
4. In the **CA Certificate** field, copy and paste the CA certificate. Link together the multiple CA certificates in **.pem** format.
5. Click **Apply to Device**.

Import a CA certificate to the trustpool

Configure a CA certificate import to the trustpool using commands.

Import a CA certificate to establish trust relationships for secure communications.

The trustpool stores CA certificates that the device uses to validate certificates presented by other entities during secure communications.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Import the root certificate to the trustpool.

   Example

   ```
   Device(config)# crypto pki trust pool import terminal
   % Enter PEM-formatted CA certificate.
   % End with a blank line or "quit" on a line by itself.
   -----BEGIN CERTIFICATE-----
   -----END CERTIFICATE-----
   ```

```
-----BEGIN CERTIFICATE-----
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
-----END CERTIFICATE-----
Aug 23 02:47:33.450: %PKI-6-TRUSTPOOL_DOWNLOAD_SUCCESS:
Trustpool Download is successful
```

You need to paste the CA certificate from the **digicert.com**.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

The CA certificate is successfully imported to the trustpool and can be used for certificate validation.

Clean the CA certificates imported in trustpool (GUI)

Configure the system to remove CA certificates imported in trustpool using GUI.

Procedure

---

1. Choose **Configuration** > **Security** > **PKI Management**.
2. In the **PKI Management** window, click the **Trustpool** tab.
3. Click **Clean**.

> **Note**
>
> This erases the downloaded CA certificate bundles. However, it does not erase the built-in CA certificate bundles.

4. Click **Yes**.

Clean CA certificates imported in trustpool (CLI)

Clean the CA certificates imported in Trustpool using commands.

You cannot delete a specific CA certificate from the trustpool. However, you can clear all the CA certificates that are imported to the Trustpool.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Erase the downloaded CA certificate bundles.

   Example

   ```
   Device(config)# crypto pki trustpool clean
   ```

   This command does not erase the built-in CA certificate bundles.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   Alternatively, you can also press Ctrl-Z to exit global configuration mode.

Create a new trustpoint dedicated to a single CA certificate

Configure a new trustpoint dedicated to a single CA certificate.

Establish a trustpoint that is specifically configured for a single Certificate Authority (CA) certificate to enable secure PKI operations.

A trustpoint represents a Certificate Authority (CA) or identity certificate that is used for PKI operations. Creating a dedicated trustpoint for a single CA certificate helps organize and manage different certificate authorities in the network.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a trustpoint.

   Example

   ```
   Device(config)# crypto pki trustpoint tp-name
   ```

   Example

   ```
   Device(config)# crypto pki trustpoint tp_name
   ```

3. Create an enrollment terminal for the trustpoint.

   Example

   ```
   Device(ca-trustpoint)# enrollment terminal
   ```

4. Exit from the trustpoint configuration.

Example

```
Device(ca-trustpoint)# exit
```

5. Authenticate the trustpoint.

Example

```
Device(config)# crypto pki authenticate tp-name
```

Example

```
Device(config)# crypto pki authenticate tp_name
<<< PASTE CA-CERT in PEM format followed by quit >>>
```

The trustpoint is now created and authenticated with the CA certificate, ready for PKI operations.

Verify LSC configuration

Provides commands to verify LSC configuration details.=""?>

Use show commands to verify Local Security Certificate (LSC) configuration. View wireless management trustpoint details and Local Security Certificate (LSC) provision-related configuration for your access points.

To view the wireless management trustpoint details, use this command:

```
Device# show wireless management trustpoint

Trustpoint  Name : microsoft-ca
Certificate Info : Available
Certificate Type : LSC
Certificate Hash : 9e5623adba5307facf778e6ea2f5082877ea4beb
 Private key Info : Available
```

To view the Local Security Certificate (LSC) provision-related configuration details for an AP, use this command:

```
Device# show ap lsc-provision summary

AP LSC-provisioning : Disabled
Trustpoint used for LSC-provisioning : lsc-root-tp
Certificate chain status : Available
Number of certs on chain : 2
Certificate hash : 7f9d05183deecac4e5a79db65d538245685e8e30
LSC Revert Count in AP reboots : 1

AP LSC Parameters :
Country : IN
State : KA
City : BLR
Orgn : ABC
Dept : ABC
Email : support@abc.com
Key Size : 2048
EC Key Size : 384 bit

AP LSC-provision List :
```

```
Total number of APs in provision list: 2

Mac Addresses :
--------------
1880.90f5.1540
2c5a.0f70.84dc
```

Configure management trustpoint to LSC

Configure management trustpoint to LSC using GUI.

Procedure

1. Choose **Administration** > **Management** > **HTTP/HTTPS**.
2. In the **HTTP Trust Point Configuration** section, set **Enable Trust Point** to the **Enabled** state.
3. From the **Trust Points** drop-down list, choose the appropriate trustpoint.
4. Save the configuration.

Configure management trustpoint to LSC (CLI)

Configure management trustpoint to LSC using commands.

Configure management trustpoint to enable Local Significant Certificate (LSC) authentication for access points.

After LSC provisioning, the APs will automatically reboot and join at the LSC mode after bootup. Similarly, if you remove the AP LSC provisioning, the APs reboot and join at non-LSC mode.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the management trustpoint to LSC.

   Example

   ```
   Device(config)# wireless management trustpoint trustpoint-name
   ```

   Example

   ```
   Device(config)# wireless management trustpoint microsoft-ca
   ```

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The management trustpoint is now configured for LSC authentication, and access points will join using LSC mode after rebooting.

AP certificate renewal (LSC)

Explains AP certificate renewal with LSC, describing benefits, prerequisites, restrictions, and procedures for configuring automatic and forced renewal through GUI and CLI, including verification steps.

AP certificate renewal (LSC)

Describes the conceptual information about AP certificate renewal (LSC).

AP certificate renewal (LSC) is a feature that

- allows the renewal of Locally Significant Certificates (LSCs) for Access Points before their expiration
- ensures that the APs can continue to provide seamless services by renewing their certificates in a timely manner
- enables the controller to orchestrate this process to minimize service disruption. The process focuses on automatic renewal mechanisms that do not require manual intervention.

LSC provisioning and renewal methods

From Cisco IOS XE 26.2.1, supported access points can provision and renew LSC through either controller-based proxy enrollment or Cisco Catalyst Center secure AP onboarding. These certificates are crucial for your Access Points because they authenticate network communications. This renewal process is essential. Without a valid certificate, Access Points may lose network connectivity and affect network services.

The controller uses these methods to renew certificates:

- **Gradual Succession**: A method where Access Points are renewed one at a time, which is suitable when the number of Access Points that are due for renewal is under a certain threshold.
- **Staggered Rollover**: Access Points are renewed in a staggered fashion, minimizing the impact on radio coverage when the number of Access Points is larger.
- **One-Shot Renewal**: All affected Access Points are renewed simultaneously, which can be disruptive and should be used as a last resort.

Benefits of AP certificate renewal (LSC)

Lists the benefits of Access Point certificate renewal using Locally Significant Certificate (LSC).

- **Automated Renewal Process**: Reduces the need for manual certificate updates, ensuring continuous service.
- **Seamless Network Operations**: Minimizes network downtime by renewing certificates before they expire.
- **Enhanced Security**: Maintains a high level of security by ensuring that all Access Points have valid certificates.
- **Scalable Management**: Supports staggered AP renewal to reduce impact on network operations.

Requirements for AP certificate renewal (LSC)

Lists the requirements for access point certificate renewal using locally significant certificates.

Configure the controller and access points to use locally significant certificates before you proceed with certificate renewal.

- Provision or renew the locally significant certificates through either the controller-based proxy-enrollment workflow or the Cisco Catalyst Center secure AP onboarding workflow.
- Define the maintenance windows if scheduling is required for renewal.

Restrictions for AP certificate renewal (LSC)

This document describes the restrictions for access point certificate renewal (Local Significant Certificate).

These restrictions for access point certificate renewal (Local Significant Certificate) are:

- Only one Local Significant Certificate is supported per access point.
- Enable either staggered or one-shot renewal, not both.
- Local Significant Certificate renewal applies to Cisco Wave 2 access points and newer models; older access points do not support this feature.
- The staggered algorithm does not apply to access points in bridge mode. All Local Significant Certificate renewals for bridge mode access points occur at the same time.

Configure AP certificate auto-renewal (LSC) (GUI)

Learn how to configure AP certificate auto-renewal (LSC) to automatically renew access point locally significant certificates according to a defined schedule.

Follow these steps to configure AP certificate auto-renewal (LSC):

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points** > **LSC Provision**.
2. In the **LSC Renewal** section, check the **LSC Auto-Renewal** check box and follow these steps:
   a. Enter a value for due time. The valid range is 5 to 60.
   b. From the **Renewal Schedule** drop-down list, select an existing calendar profile schedule or click the icon to create a new calendar profile. For more information about creating a calendar profile, see the **Configuring a Calendar Profile (GUI)** and **Configuring a Calendar Profile (CLI)**.
   c. Use the slider to enable **Syslog** messages for the LSC renewal updates.
   d. From the **Renewal Mode** drop-down list, choose either *One Shot* or *Staggered*.
      - If you choose One Shot, all impacted AP LSCs renew in a single iteration.
      - If you choose Staggered, the system staggers the LSC renewal configuration for the APs and follow these steps:
        - Enter a value for the iteration expiry to set the expiry time for each iteration.
        - From the **AP LSC Renewal per Iteration** drop-down list, choose one of the following options to configure the percentage of APs to be renewed per iteration: 5%, 15%, or 25%.

          You can also see 1% and serial.
        - Enable the **Client Steering** slider to move clients attached to APs undergoing renewal to other APs.
        - Enable the **Client Deauthentication** slider to deauthenticate clients associated with the AP before it starts upgrading.

- From the **Accounting Percentage** drop-down list, select the percentage of APs that should join the controller after each iteration to consider the renewal successful.
- Use the slider to configure **Accounting Action**. If you select **Terminate**, the renewal stops if the configured percentage of APs does not join the controller. If you select **Ignore**, the renewal continues regardless.

3. Click **Apply**.

Configure AP certificate renewal (LSC) (CLI)

Use this procedure to configure AP certificate renewal (LSC) via CLI.

Follow these steps to configure AP certificate renewal (LSC) in the controller:

Procedure

1. Enter configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure LSC provisioning on an AP.

   Example

   ```
   Device(config)# ap lsc-provision
   ```

3. Configure AP certificate renewal.

   Example

   ```
   Device(config)# ap lsc-provision renew
   ```

4. Renew AP LSCs automatically within the specified number of days.

   Example

   ```
   Device(config)# ap lsc-provision renew due-time number-of-days
   ```

   The valid range is from 5 to 60 days, with a default value of 30 days.

5. Renew all AP LSCs in a single iteration.

   Example

   ```
   Device(config)# ap lsc-provision renew one-shot
   ```

> **Note**
>
> The one-shot AP LSC renewal feature is disabled by default.

6. Renew AP LSCs automatically in a staggered manner.

   Example

   ```
   Device(config)# ap lsc-provision renew staggered timeout
   iteration-timeout-in-mins
   ```

   Staggered AP LSC renewal is enabled by default.

   > **Note**
   >
   > You cannot use one-shot and staggered renewal methods at the same time.
   >
   > *iteration-timeout-in-mins*: The valid range is from 5 to 60 minutes.

7. Configure time schedule for auto-renewal within a maintenance window.

   Example

   ```
   Device(config)# ap lsc-provision renew scheduler calendar-profile
   ```

   No scheduler is enabled by default.

   When no scheduler is applied, the device auto-renews AP LSCs at 24-hour intervals and is triggered daily at 12:00 AM local time.

   > **Note**
   >
   > You must configure the calendar-profile definition beforehand. For more information, see **Configuring a Calendar Profile (CLI)**.

You have successfully configured AP certificate renewal (LSC).

Configure forced renewal of AP certificate (LSC) for individual or multiple APs (GUI)

Configure forced renewal of AP certificate (LSC) for individual or multiple APs using the GUI interface.

Follow these steps to configure forced renewal of AP certificate (LSC) for individual or multiple APs via GUI:

Procedure

1. Choose **Configuration** > **Wireless** > **Bulk AP Provisioning**.
2. On the **Select Parameters** page, check the **Renew LSC** check box to renew the LSC for all selected APs.
3. Click **Next**.

Configure forced renewal of AP Certificate (LSC) based on site tag (GUI)

Configure the forced renewal of Access Point (AP) Certificate (LSC) based on site tag through the graphical user interface.

Before you begin

Follow these steps to configure force renewal of AP certificate (LSC) based on site tag:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. In the **Site** tab, check the check box for the required site tag.
3. Click **Renew LSC** and follow these steps:
   a. Enter the due time in days. The value must be between 1 and 60.
   b. Click **Apply to Device**.

Configure forced renewal of AP Certificate (LSC) based on site tag (CLI)

Use this procedure to configure forced renewal of AP certificate (LSC) based on site tag using the CLI.

Follow these steps to configure the forced renewal of an AP certificate (LSC) based on a site tag using the CLI:

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Renew a specific site-tag AP that is set to expire within the specified time.

   Example

   ```
   Device# ap lsc renew due-time number-of-days site-tag site-tag-name
   ```

   number-of-days: The valid range is from 1 to 60 days.

The configuration of the forced renewal of an AP Certificate (LSC) based on a site tag is complete.

Configure forced renewal of all APs (GUI)

Configure forced renewal of Local Significant Certificates (LSC) for all access points using the graphical user interface.

Follow these steps to configure forced renewal of all APs (GUI):

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. From the **Select an Action** drop-down list, choose **Renew LSC**.

   The **Renew LSC** pop-up window is displayed.
3. Enter a due time value in minutes between 5 and 60.
4. Click **Apply to Device**.

Verify AP certificate renewal (LSC)

This section shows you how to verify AP Certificate Renewal (Local Significant Certificate) using show commands.

Lists the various show commands and their outputs during the verification of the AP Certificate Renewal (LSC).

Verify LSC configuration

To verify LSC provision-related configuration details for an AP, use this command:

```
Device# show ap lsc-provision info
AP name     Ethernet MAC     Radio MAC      LSC authentication  LSC workflow
   Cert expiry   Last renew attempt  Last renew failure
=============================================================================================================
AP-abc      1111.2222.3333  aaaa.bbbb.ccc0  Port-802.1x         PNP Onboarding
  <time-stamp>  <time-stamp>/NA     reason-code
AP-123      1234.5678.9012  dddd.eeee.fff0  DTLS                SCEP proxy
   <time-stamp>  <time-stamp>/NA     reason-code
AP-xyz      1010.1010.1010  abcd.efab.cde0  DTLS, Port-802.1x   SCEP proxy
   <time-stamp>  <time-stamp>/NA     reason-code
                                                                     0.0.0.0
                       372           64                AGL          Enabled
```

Verify staggered LSC renewal upgrade configuration

To verify progress and status of staggered LSC renewal upgrade, use this command:

```
Device# show ap upgrade
Status: In progress
From version:
To version: Unknown

Started at: 10/23/2025 08:16:01 IST
Configured percentage: 15
Percentage complete: 0
Expected time of completion: 10/23/2025 08:24:01 IST

Client steering: Enabled
Client de-authentication: Enabled
Accounting percentage: 90%
```

```
Iteration expiry time: 15 minutes

Progress Report
---------------
Iterations
----------
Iteration              Start time                          End time
          AP count
-------------------------------------------------------------------------------------------
0                     10/23/2025 08:16:01 IST             10/23/2025 08:16:01
 IST     0
1                     10/23/2025 08:16:01 IST             10/23/2025 08:18:05
 IST     1
2                      10/23/2025 08:18:05 IST             ONGOING
           0

Upgraded
--------
Number of APs: 0
AP Name                          Radio MAC               Iteration       Status
               Site
-------------------------------------------------------------------------------------------

In Progress
-----------
Number of APs: 1
AP Name                           Radio MAC
-----------------------------------------------
1815W                             00be.7517.d980

Remaining
---------
Number of APs: 6
AP Name                           Radio MAC
-----------------------------------------------
9166-1                            10f9.20fd.5400
non-9120-1                        488b.0a47.d8c0
9136_1                            687d.b45f.1610
9120-Dev                          a00f.3704.f3a0
9162-antenna                      ecf4.0c0e.9920
9176+alpine                       ecf4.0c92.bac0

APs not handled by Rolling AP Upgrade
-------------------------------------
AP Name                          Radio MAC               Status          Reason
 for not handling by Rolling AP Upgrade
-------------------------------------------------------------------------------------------
```

MIC and LSC APs joining the controller

Explains the authorization policy, configuration workflows, and verification steps for enabling MIC and LSC APs to securely join the controller, highlights behavioral changes, and outlines recommendations and limitations for policy deployment.

Support for MIC and LSC APs joining the controller

Explains the authorization policy that allows MIC APs to join the LSC-deployed controller, enabling LSC and MIC APs to coexist.

Support for MIC and LSC access points joining the controller is an authorization policy feature that

- allows MIC APs to join LSC-deployed controllers without requiring LSC provisioning
- enables LSC and MIC APs to coexist on the same controller simultaneously, and
- eliminates the need to provision LSC on MIC APs before moving them to LSC-deployed controllers.

Authorization policy behavior changes

The authorization policy behavior differs between software releases:

- **Cisco IOS XE Bengaluru 17.4.1 and earlier**: If your APs have default certificates (Manufacturing Installed Certificates [MIC] or Secure Unique Device Identifier [SUDI]), they cannot join an LSC-deployed controller, where the management certificate of the controller is an LSC. To resolve this issue, provision LSC on your APs. Use the provisioning controller before you move them to the LSC-deployed controller.
- **Cisco IOS XE Bengaluru 17.5.1 onwards**: The new authorization policy configuration allows MIC APs to join the LSC-deployed controller, so that LSC APs and MIC APs can coexist on the controller.

Recommendations and limitations

Outlines recommendations and limitations for MIC and LSC Access Points joining the controller.

When you configure the CA server with manual enrollment to accept certificate signing request (CSR), the controller monitors for the CA server response. If the CA server does not respond for 10 minutes, the fallback mode activates.

- Cisco Wave 2 APs regenerate CSR, and a fresh CSR is sent to the CA server.
- Cisco IOS APs restart, and then Cisco IOS APs send a fresh CSR, which is in turn sent to the CA server.

The locally significant certificate (LSC) feature on the controller does not work with password challenge authentication. Therefore, for LSC to work, you must disable password challenge on the CA server.

If you are using Microsoft CA, use Windows Server 2012 or later as the CA server.

MIC and LSC access point authentication

Describes the configuration workflow for MIC and LSC access points joining the controller.

Summary

The Manufacturer Installed Certificate (MIC) and Local Significant Certificate (LSC) access point authentication configuration involves these stages:

- Controller: Manages authentication policies and trustpoint configuration for secure access point connections
- Access points: Require certificate-based authentication to join the controller
- LSC trustpoint: Provides local significant certificate authentication mechanism
- AP certificate policy: Defines authentication requirements and allowed access point lists

Workflow

The key components involved in the MIC and LSC access point authentication configuration are:

1. The network administrator configures the LSC trustpoint on the LSC deployed controller using CLI.

   Configure LSC on the controller (CLI) on page 1767

2. The network administrator enables the AP certificate policy on access points using CLI.

   Enable the AP certificate policy on the APs (CLI) on page 1767

3. The network administrator configures the AP policy certificate using the GUI interface.

Configure the AP policy certificate (GUI) on page 1769

4. The network administrator configures the allowed list of APs to join the controller using CLI.

Configure the allowed list of APs to join the controller (CLI) on page 1769

Result

Access points are properly authenticated and authorized to join the controller with secure certificate-based authentication enabled.

Configure LSC on the controller (CLI)

Configure LSC on the controller using commands.

The server certificate used by the controller for CAPWAP-DTLS (Control and Provisioning of Wireless Access Points - Datagram Transport Layer Security) is based on the following configuration.

Before you begin

- Ensure that you enable LSC by setting the appropriate trustpoints for the following wireless management services:
  - AP join process: CAPWAP DTLS server certificate
  - Mobility connections: Mobility DTLS certificate
  - NMSP (Network Mobility Services Protocol) and CMX (Connected Mobile Experiences) connections: NMSP TLS certificate

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the LSC trustpoint in the LSC-deployed controller.

Example

```
Device(config)# wireless management trustpoint trustpoint-name
```

Enable the AP certificate policy on the APs (CLI)

Configure the AP certificate policy on the APs using commands.

Enable AP certificate policy during CAPWAP-DTLS handshake to allow MIC APs to join the controller.

- If the management trustpoint is a locally significant certificate (LSC), by default, manufacturing installed certificate (MIC) APs fail to join the controller. This configuration acts as an enable or disable configuration knob that allows MIC APs to join the controller.
- This configuration is a controller authorization to allow APs to join MIC at the time of Datagram Transport Layer Security (DTLS) handshake.

To prevent manufacturing installed certificate (MIC) expiry failures, ensure that you configure a policy, as shown here:

- Create a certificate map and add the rules:

```
configure terminal
crypto pki certificate map map1 1
issuer-name co Cisco Manufacturing CA
```

> **Note**
>
> You can add multiple rules and filters under the same map. The rule mentioned in the example above specifies that any certificate whose issuer-name contains *Cisco Manufacturing CA* (case insensitive) is selected under this map.

- Use the certificate map under the trustpool policy:

```
configure terminal
crypto pki trustpool policy
match certificate map1 allow expired-certificate
```

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the trustpoint name for the controller certificate chain.

   Example

   ```
   Device(config)# ap auth-list ap-cert-policy allow-mic-ap trustpoint
   trustpoint-name
   ```

   > **Note**
   >
   > The allow-mic-ap trustpoint command is required only for the virtual controller (Cisco Catalyst 9800-CL Wireless Controller for Cloud). In all the other appliance controller platforms, the default certificate is selected. This default certificate is manufacturer-installed SUDI.

3. Enable the AP certificate policy during CAPWAP-DTLS handshake.

   Example

   ```
   Device(config)# ap auth-list ap-cert-policy allow-mic-ap
   ```

4. Enable the AP certificate policy as MIC.

Example

```
Device(config)# ap auth-list ap-cert-policy mac-address H.H.H policy-type mic
```

Example

```
Device(config)# ap auth-list ap-cert-policy mac-address 1111.1111.1111
policy-type mic
```

The AP certificate policy is now enabled and configured for MIC APs to join the controller during CAPWAP-DTLS handshake.

Configure the AP policy certificate (GUI)

Configure the AP policy certificate using GUI.

Procedure

1. Choose **Configuration > Wireless > Access Points**
2. In the **All Access Points** window, click **AP Certificate Policy**.
3. In the **AP Policy Certificate** window, complete the following actions:
   a) Click the **Authorize APs joining with MIC** toggle button to enable AP authorization.
   b) From the **Trustpoint Name** drop-down list, choose the required trustpoint.
   c) Click **Add MAC or Serial Number** to add a MAC address or a serial number manually or through a .csv file. The **Add MAC or Serial Number** window is displayed.
   d) Click the **AP Authlist Type** and enter the MAC address or the serial number. Upload the .csv file or enter the MAC address in the list box.
   The newly added MAC address and serial numbers are displayed under **List of MAC Address and Serial Numbers**.
   e) Click **Apply**.

   The AP certificate policy is added to the **AP Inventory** window.

> **Note**
>
> To add a new AP with MIC, perform Step 1 to Step 3 described in Configuring the AP Policy Certificate (GUI) section. To add a new AP with LSC, perform the procedure described in the Configuring AP LSC Provision List (GUI) and Step 1 to Step 3 in the Configuring the AP Policy Certificate (GUI) section.

Configure the allowed list of APs to join the controller (CLI)

Configure the allowed list of APs to join the controller using commands.

Create an allowed list of access points that can join the controller based on MAC address or serial number authentication.

The allowed list of APs can either be populated based on the Ethernet MAC address or based on the serial number of the APs.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the AP certificate policy based on the Ethernet MAC address or based on the assembly serial number of the AP.

   Example

   ```
   Device(config)# ap auth-list ap-cert-policy mac-address AP-Ethernet-MAC-address
    policy-type mic
   ```

   Example

   ```
   Device(config)# ap auth-list ap-cert-policy serial-number AP-serial-number
   policy-type mic
   ```

   Example

   ```
   Device(config)# ap auth-list ap-cert-policy mac-address 00b0.e192.0d98
   policy-type mic
   ```

The AP certificate policy is configured and the specified access points are added to the allowed list for joining the controller.

Verifying the configuration status

Provides commands to verify the configuration status of AP certificate policy authorization.

To verify if the APs have been authorized by the AP certificate policy, use this command:

```
Device# show ap auth-list ap-cert-policy
Authorize APs joining with MIC : ENABLED
MIC AP policy trustpoint
Name : CISCO_IDEVID_SUDI
Certificate status : Available
Certificate Type : MIC
Certificate Hash : xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx
```

To verify the AP certificate policy on the MAC address and the serial number of the AP, use these commands:

```
Device# show ap auth-list ap-cert-policy mac-address
MAC address      AP cert policy
---------------------------------
1111.2222.3333   MIC
```

```
Device# show ap auth-list ap-cert-policy serial-number
Serial number    AP cert policy
```

```
--------------------------------
F1234567890      MIC
```

Note

If you set an invalid trustpoint (not SSC), the **allow-mic-ap policy** does not enable. If you set an invalid trustpoint, this error displays on the console:

```
Device(config)# ap auth-list ap-cert-policy allow-mic-ap trustpoint lsc-root-tp
Dec 18 07:38:29.944: %CERT_MGR_ERRMSG-3-CERT_MGR_GENERAL_ERR: Chassis 1 R0/0:
 wncd: General error: MIC AP Policy trustpoint: 'lsc-root-tp' cert-chain type
 is LSC, It must be either MIC or vWLC-SSC
```

LSC fallback APs

Explains LSC fallback mechanisms for CAPWAP authentication, provides methods to identify APs in fallback state, guides troubleshooting and syslog analysis, and details recovery commands to restore normal LSC functionality.

LSC fallback APs

Explains the state when the AP falls back to its default certificate (MIC) for CAPWAP after maximum number of failures.

LSC fallback is a state that

- occurs when an AP configured with LSC for CAPWAP fails to establish DTLS connection after reaching the maximum number of retry attempts
- causes the AP to revert to its default certificate (MIC) for CAPWAP authentication, and
- provides a backup authentication method when LSC-based CAPWAP connection attempts fail.

LSC fallback process

When you configure an AP with LSC for CAPWAP but it fails to establish a DTLS connection, the AP reboots. The AP then retries for a certain number of times. For information on how to configure an AP with LSC, see Configure AP join attempts with LSC certificate (CLI) on page 1747.

The AP falls back to its default certificate (MIC) for CAPWAP after maximum number of failures. The AP fallback state is referred to as the LSC fallback.

Note

MIC is also known as SUDI certificate.

Troubleshoot LSC fallback state

Configure troubleshooting procedures to identify and resolve LSC fallback APs using show wireless summary.=""?>

This task enables you to identify access points in LSC fallback state and understand the recovery process when APs fail to establish proper LSC connections with the controller.

When an AP in LSC fallback state joins the controller, it generates specific syslog messages and is held in a misconfigured state. The AP does not broadcast WLAN or SSID configurations in this state, which permits the admin to examine the reason for previous failures and recover APs.

Before you begin

Procedure

1. Identify the syslog message generated when an AP in LSC fallback state joins the controller.

   Example

   ```
   Jun 15 23:24:14.836: %APMGR_TRACE_MESSAGE-3-WLC_GEN_ERR: Chassis 1 R0/0: wncd:
    Error
   in AP: 'AP2c5a.0f70.84dc' with address 70db.9888.cc20 is joined with MIC,
   while configuration
   requires LSC. No WLANs will be pushed.
   ```

   The controller allows such an AP to be joined with MIC (when AP certificate policy allows it) and AP is held in misconfigured state.

   > **Note**
   >
   > The AP does not broadcast WLAN or SSID configurations in such state. This permits the admin to examine the reason for previous failures and recover APs.

2. Use **show wireless summary** to identify LSC fallback APs.

   Example

   ```
   Device# show wireless summary
   …
   Access Point Summary
   …
   DTLS LSC fallback APs     20 (No WLANs will be pushed to these APs)
   …
   For more information on DTLS LSC fallback APs,
       execute 'wireless config validate' and look for reported errors in
       'show wireless config validation status' CLI output.

   Use 'show ap config general | inc AP Name | LSC fallback' to list DTLS LSC
   fallback APs.
   Examine LSC fallback reasons / DTLS handshake failures with LSC then
       issue 'ap lsc dtls-fallback clear-certificate / clear-flag' to recover
   APs
   ```

You have successfully identified LSC fallback APs and understand the commands available for further investigation and recovery.

Recover from LSC fallback

Configure recovery steps to restore LSC functionality after DTLS fallback occurs.

Recover APs from LSC fallback state and restore normal LSC functionality for CAPWAP DTLS connections.

When LSC fallback occurs, APs may require recovery actions to restore normal operation. Recovery options depend on whether you want to retain the LSC certificate or clear it completely.

Procedure

Choose one of the following recovery methods based on your requirements:

- To clear the LSC fallback flag and retain the LSC certificate, use the **ap lsc dtls-fallback clear-flag** command to instruct the AP to reload.

  Note

  The AP reuses the LSC for CAPWAP DTLS connection post the reload.

- To clear the LSC certificate completely, use the **ap lsc dtls-fallback clear-certificate** command to clear LSC and instruct the AP to reload.

  Note

  The AP uses MIC for CAPWAP-DTLS post the reload. If LSC is used for Dot1x port authentication then further recovery is needed on switch port for AP authentication.

Note

- The **ap lsc dtls-fallback clear-flag** command is sufficient to retain LSC on AP. Both **ap lsc dtls-fallback clear-flag** and **ap lsc dtls-fallback clear-certificate** commands are not required at the same time.
- APs must be in connected state when issuing the recovery command. You will need to reissue the command, if any **LSC fallback** AP joins afterwards.

The access point reloads and recovers from the LSC fallback state according to the selected recovery method.

Configuring controller self-signed certificate for wireless AP join

Outlines the process of configuring and managing controller self-signed certificates to enable wireless AP join, covering use cases, prerequisites, setup steps, PKI server and trustpoint configuration, verification, and server management.

Use cases for controller self-signed certificate for wireless AP join

Lists the scenarios in which controller self-signed certificates are required for wireless AP join, including platform limitations and compatibility issues with certain APs.

You need controller self-signed certificates in specific circumstances to enable wireless APs to join the controller successfully.

Controller self-signed certificate use cases

These situations require controller self-signed certificates:

- The Cisco Catalyst 9800-CL platform is not equipped with manufacturer-installed SUDI certificates. Configure self-signed certificates on these controllers.
- Some APs running earlier software and using a Manufacturer Installed Certificate (MIC) issued by a SHA1 Cisco Trusted CA cannot join controllers with SHA2 SUDI certificates. During the CAPWAP join process, these APs display a bad certificate error and fail the DTLS handshake.

  Workaround: To support AP upgrades, configure a self-signed certificate for the controller as a workaround for the second use case. After updating all APs, delete the self-signed certificates. Then, revert to the SUDI certificate.

  > **Note**
  >
  > This workaround does not apply to Embedded Wireless Controllers running on Catalyst 9000 series switches. It is only valid for hardware appliance controllers such as Cisco Catalyst 9800-40, 9800-80, and 9800-L.

- The command **wireless management certificate ssc auth-token** is required for ensuring resilient operation of Cisco Catalyst 9800-CL controllers. It is mandatory to configure an explicit authentication token on all 9800-CL devices within the network. Additionally, this command is required when migrating from an AireOS controller to a Cisco Catalyst 9800-CL controller to facilitate secure AP join, as the 9800-CL uses a self-signed certificate instead of a MIC used by AireOS controllers.

Prerequisites for controller self-signed certificate for wireless AP join

Follow these guidelines to prepare the controller for self-signed certificate configuration. This enables wireless AP join. Meet all prerequisites before setup.

- Ensure that the VLAN interface is up and that the IP address is reachable.
- Ensure that the **ip http server** is enabled. For more information, see Enabling HTTP Server .
- Set the **clock calendar-valid** command appropriately. For more information, see *Configure clock calendar.*
- Check whether the PKI CA server is configured. If it is, delete the existing CA server configuration.

> **Note**
>
> The **show crypto pki server** command output should not display anything.

Configure clock calendar (CLI)

Configure clock calendar using commands to ensure accurate timekeeping on the device.

Set the clock calendar to ensure accurate timekeeping on the device.

This configuration is necessary for time-sensitive operations and logging on the device.

Before you begin

Access to the device with appropriate privileges to enter global configuration mode.

Follow these steps to configure clock calendar:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable clock calendar.

   Example

   ```
   Device(config)# clock calendar-valid
   ```

3. Exit configuration mode.

   Example

   ```
   Device(config)# exit
   ```

The clock calendar is now configured and operational on the device.

Enable HTTP server using CLI

Configure the HTTP server to enable web-based management of the device using commands.

Enable the HTTP server to allow web-based management of the device.

This configuration is applicable for devices that require HTTP access for management purposes.

Before you begin

Ensure you have administrative access to the device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable the HTTP server on your IP or IPv6 system.

   Example

   ```
   Device(config)# ip http server
   ```

   Enables the HTTP server on your IP or IPv6 system, including a Cisco web browser user interface. By default, the HTTP server uses the standard port 80.

3. Enable the secure HTTP server on your IP or IPv6 system.

   Example

   ```
   Device(config)# ip http secure-server
   ```

   Enables the secure HTTP server on your IP or IPv6 system, providing encrypted access to the web interface.

4. Exit configuration mode.

   Example

   ```
   Device(config)# exit
   ```

The HTTP server is now enabled, allowing web-based management of the device.

Configure CA server (CLI)

Configure a CA server using commands to manage certificates for devices in the network.

Deploy a CA server to manage certificates for devices in the network.

This configuration is essential for environments that require secure communication through certificates.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Generate RSA keys for the CA server.

Example

```
Device(config)# crypto key generate rsa general-keys modulus size_of_key_module
 label keypair_name
```

Configures a certificate for the controller.

When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.

> **Note**
>
> The recommended key-pair name is *WLC_CA* and key modulus is *2048* bits.

3. Enable the IOS certificate server.

Example

```
Device(config)# crypto pki server certificate_server_name
```

Enables IOS certificate server.

> **Note**
>
> The *certificate_server_name* must be the same name as the *keypair_name*.

4. Configure the issuer name for the CA certificate.

Example

```
Device(cs-server)# issuer-name O=Cisco Virtual Wireless LAN Controller,
CN=CA-vWLC
```

Configures X.509 distinguished name for the issuer CA certificate.

> **Note**
>
> You need to configure the same *issuer-name* as suggested for AP join.

5. Grant certificate requests automatically.

Example

```
Device(cs-server)# grant auto
```

Grants certificate requests automatically.

6. Optional: Specify the hash function for the signature used in the granted certificates.

   Example

   ```
   Device(cs-server)# hash sha256
   ```

   Specifies the hash function for the signature used in the granted certificates.

7. Optional: Specify the lifetime in days of a CA certificate.

   Example

   ```
   Device(cs-server)# lifetime ca-certificate time-interval
   ```

   Specifies the lifetime in days of a CA certificate.

8. Optional: Specify the lifetime in days of a granted certificate.

   Example

   ```
   Device(cs-server)# lifetime certificate time-interval
   ```

   Specifies the lifetime in days of a granted certificate.

9. Set the CA key and CA certificate archive format and password.

   Example

   ```
   Device(cs-server)# database archive pkcs12 password 0 cisco123
   ```

   Sets the CA key and CA certificate archive format and password to encrypt the file.

10. Enable the certificate server.

    Example

    ```
    Device(cs-server)# no shutdown
    ```

    Enables the certificate server.

    > **Note**
    >
    > Issue this command only after you have completely configured your certificate server.

11. Return to privileged EXEC mode.

    Example

    ```
    Device(cs-server)# end
    ```

    Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The CA server is now configured and ready to manage certificates for devices in the network.

Configure trustpoint (CLI)

Configure a trustpoint to enable secure communication with a Certificate Authority (CA) using commands.

Configure a trustpoint to enable secure communication with a Certificate Authority (CA).

This configuration is necessary for devices that require secure communications and certificate management.

Before you begin

Ensure you have access to the device and the necessary permissions to configure trustpoints.

Follow these steps to configure trustpoint using CLI commands:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Generate RSA keys for the trustpoint.

   Example

   ```
   Device(config)# crypto key generate rsa exportable general-keys modulus
   size-of-the-key-modulus label label
   ```

   When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.

3. Create a new trustpoint for an external CA server.

   Example

   ```
   Device(config)# crypto pki trustpoint trustpoint_name
   ```

   Here, *trustpoint_name* refers to the trustpoint name.

   > **Note**
   >
   > Ensure that same names are used for key-pair (*label*) and *trustpoint_name*.

4. Map RSA key with that of the trustpoint.

   Example

   ```
   Device(ca-trustpoint)# rsakeypair RSA_key
   ```

   Maps RSA key with that of the trustpoint.

- RSA_key–Refers to the RSA key pair label.
- key_size–Refers to the signature key length. The value ranges from 360 to 4096.

5. Create subject name parameters for the trustpoint.

   Example

   ```
   Device(ca-trustpoint)# subject-name subject_name
   ```

   Specifies the subject name for the trustpoint.

6. Check revocation.

   Example

   ```
   Device(ca-trustpoint)# revocation-check none
   ```

   Specifies whether to check for certificate revocation.

7. Specify the hash algorithm.

   Example

   ```
   Device(ca-trustpoint)# hash sha256
   ```

   Specifies the hash algorithm to be used.

8. Specify the serial number.

   Example

   ```
   Device(ca-trustpoint)# serial-number
   ```

   Specifies the serial number for the trustpoint.

9. Optional: Set certificate key-usage purpose.

   Example

   ```
   Device(ca-trustpoint)# eku request server-auth client-auth
   ```

   Sets the extended key usage for the certificate.

10. Enable password for the trustpoint.

    Example

    ```
    Device(ca-trustpoint)# password 0 password
    ```

11. Enroll the URL for the CA server.

    Example

    ```
    Device(ca-trustpoint)# enrollment url url
    ```

Replace the dummy IP with management VLAN interface IP of the controller where CA server is configured.

12. Exit the configuration mode.

Example

```
Device(ca-trustpoint)# end
```

The trustpoint is now configured and ready for use with the CA server.

Authenticate and enroll the PKI trustpoint with CA server

Configure the PKI trustpoint to authenticate and enroll with the Certification Authority (CA) server using commands.

Authenticate and enroll the PKI trustpoint to establish a secure connection with the Certification Authority (CA) server.

This procedure is used in environments where a PKI trustpoint needs to be authenticated and enrolled with a Certification Authority (CA) server for certificate management.

Before you begin

Ensure that the Certification Authority (CA) server is reachable and the trustpoint is properly configured.

Follow these steps to authenticate and enroll the PKI trustpoint with CA server:

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Authenticate the PKI trustpoint with the CA server.

Example

```
Device(config)# crypto pki authenticate trustpoint-name
Certificate has the following attributes:
Fingerprint MD5: 64C5FC9A C581D827 C25FC3CF 1A7F42AC
Fingerprint SHA1: 6FAFF812 7C552783
6A8FB566 52D95849 CC2FC050
% Do you accept this certificate? [yes/no]: yes
Trustpoint CA certificate accepted.
```

Fetches the CA certificate.

3. Enroll the PKI trustpoint for a client certificate.

Example

```
Device(config)# crypto pki enroll trustpoint-name
Enter following answers for UI interaction:
% Include an IP address in the subject name? [no]: no
Request certificate from CA? [yes/no]: yes
```

4. Exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

   Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The PKI trustpoint is now authenticated and enrolled with the Certification Authority (CA) server, allowing for secure certificate management.

Tag wireless management trustpoint name

Configure a wireless management trustpoint name for identification and management purposes using commands.

Tag a wireless management trustpoint name for identification and management purposes.

This configuration is used in environments where wireless management trustpoints need to be clearly identified for management and security.

Before you begin

Ensure you have the necessary privileges to enter global configuration mode.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Tag the wireless management trustpoint name.

   Example

   ```
   Device(config)# wireless management trustpoint trustpoint-name
   ```

3. Exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

   Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The wireless management trustpoint name is successfully tagged and can be used for management purposes.

Disable PKI server

Configure the PKI server to be disabled and prevent it from issuing certificates using commands.

Disable the PKI server to prevent it from issuing certificates.

This procedure is used in scenarios where the PKI server is no longer needed or must be temporarily disabled.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a new server for an external CA server.

   Example

   ```
   Device(config)# crypto pki server certificate-server-name
   ```

   The *certificate-server-name* refers to the server name.

   > **Note**
   >
   > Here, server_name refers to the server name.

3. Disable the certificate server.

   Example

   ```
   Device(cs-server)# shutdown
   ```

   Update purpose string to \

4. Return to privileged EXEC mode.

   Example

   ```
   Device(cs-server)# end
   ```

   Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The PKI server is now disabled and will not issue certificates.

Verify controller certificates for wireless AP join

Lists commands to verify controller certificates for wireless AP join.

Use these commands to verify controller certificates for wireless access point (AP) join.

To view the CA server details, use the command:

```
Device# show crypto pki server
Certificate Server WLC_CA:
```

```
Status: enabled
State: enabled
Server's configuration is locked (enter "shut" to unlock it)
Issuer name: O=Cisco Virtual Wireless LAN Controller, CN=CA-vWLC
CA cert fingerprint: 79A3DBD5 59A7E384 73ABD152 C133F4E2
Granting mode is: auto
Last certificate issued serial number (hex): 1
CA certificate expiration timer: 12:04:00 UTC Mar 8 2029
CRL NextUpdate timer: 18:04:00 UTC Mar 11 2019
Current primary storage dir: nvram:
Database Level: Minimum - no cert data written to storage
```

To view the trustpoint details, use this command:

```
Device# show crypto pki trustpoint ewlc-tp1 status
Trustpoint ewlc-tp1:
...
State:
Keys generated ..... Yes (General Purpose, exportable)
Issuing CA authenticated ..... Yes
Certificate request(s) ..... Yes
```

To view the wireless management trustpoint details, use this command:

```
Device# do show wireless management trustpoint
Trustpoint Name : ewlc-tp1
Certificate Info : Available
Certificate Type : SSC
Certificate Hash : 4a5d777c5b2071c17faef376febc08398702184e
Private key Info : Available
FIPS suitability : Not Applicable
```

To view the Hypertext Transfer Protocol (HTTP) server status, use this command:

```
Device# show ip http server status | include server status
HTTP server status: Enabled
HTTP secure server status: Enabled
```

147 Public Key Infrastructure Management

Topics:

- About PKI management (GUI)
- Configuration Tasks

About PKI management (GUI)

Introduces Public Key Infrastructure concepts and GUI-based management, highlighting restrictions such as CSR generation hanging indefinitely and providing foundational information for secure certificate administration.

A public key infrastructure (PKI) management is a network application tool that

- enables graphical configuration of PKI components using GUI.
- provides tabs for configuring and monitoring trustpoints, certificate authorities, keys, and certificates,
- allows administrators to generate, enroll, and manage cryptographic credentials for secure communications, and

Feature history

Feature Name	Release	Description
Enhanced Certificate Management Through GUI	Cisco IOS XE 17.3.1	The PKI Management page now includes tabs for Trustpoints, CA Server, Key Pair Generation, and Certificate Management operations.

Tabs for PKI management in GUI

- **Trustpoints tab:** Add, create, or enroll a new trustpoint, view the current trustpoints and their usage.
- **CA Server tab**: Enable or disable Certificate Authority (CA) server functionality. This is required for generating a self-signed certificate (SSC).
- **Key Pair Generation tab**: Generate key pairs.
- **Certificate Management tab**: Generate and manage certificates, and perform all certificate-related operations.

Restriction: CSR generation on the GUI may hang indefinitely

The controller's GUI-based certificate management feature relies on Embedded Event Manager (EEM) scripts. If TACACS command authorization is enabled, the user is authenticated on the GUI but the controller does not pass the user details to the EEM environment. As a result, the user is not authorized, and the user cannot enter the configuration mode using the EEM scripts.

The restriction applies if you have this command-authorization configuration:

```
aaa authorization commands 1 MethodName group GroupName local if-authenticated
aaa authorization commands 15 MethodName group GroupName local if-authenticated
ip http authentication aaa command-authorization 1 MethodName
ip http authentication aaa command-authorization 15 MethodName
```

To resolve this issue, enter this command before you generate the Certificate Signing Request (CSR) using the GUI. For proper authorization, configure the desired username.

```
event manager session cli username <username>
```

After you issue the CSR using the GUI, remove this configuration.

The restriction does not affect CSR generation through the CLI.

Configuration Tasks

Details GUI-driven configuration tasks for PKI, including trustpoint authentication and enrollment, AP self-signed certificate generation, CA server addition, RSA or EC key setup, and certificate management, covering CSR generation and PKCS12 import procedures.

Authenticate and enroll a Public Key Infrastructure (PKI) trustpoint (GUI)

Use this procedure to authenticate and enroll a Public Key Infrastructure (PKI) trustpoint through the device's graphical user interface.

Use this procedure to add and authenticate a Public Key Infrastructure (PKI) trustpoint to enable certificate-based security for your device.

Before you begin

Obtain the enrollment URL and certificate authority information from your network administrator.

Procedure

1. Choose **Configuration** > **Security** > **Public Key Infrastructure (PKI) Management**.
2. In the **Public Key Infrastructure (PKI) Management** window, click the **Trustpoints** tab.
3. In the **Add Trustpoint** dialog box, provide this information:
   a) In the **Label** field, enter the RSA key label.
   b) In the **Enrollment URL** field, enter the enrollment URL.
   c) Check the **Authenticate** check box to authenticate the Public Certificate from the enrollment URL.
   d) In the **Subject Name** section, enter the **Country Code**, **State**, **Location**, **Organization**, **Domain Name**, and **Email Address**.
   e) Check the **Key Generated** check box to view the available RSA keypairs. Choose an option from the **Available RSA Keypairs** drop-down list.
   f) Check the **Enroll Trustpoint** check box.
   g) In the **Password** field, enter the password.
   h) In the **Re-Enter Password** field, confirm the password.
   i) Click **Apply to Device**.

   The new trustpoint is added to the trustpoint name list.

The new trustpoint is now listed in the trustpoint name list and is ready for use.

What to do next

After enrolling the trustpoint, configure it for use with your security policies or certificate-based authentication features.

Generate an AP self-signed certificate (GUI)

Generate an AP self-signed certificate using GUI.

Create a self-signed certificate (SSC) for APs on a virtual controller using the graphical interface.

Use this procedure only for virtual controllers.

- Cisco Catalyst 9800-CL Wireless Controller for Cloud
- Cisco Catalyst 9800-40 Wireless Controller
- Cisco Catalyst 9800-80 Wireless Controller

- Cisco Catalyst 9800-L Wireless Controller (Copper uplink)
- Cisco Catalyst 9800-L Wireless Controller (Fiber uplink)

Procedure

1. Choose **Configuration** > **Security** > **PKI Management**.
2. In the **AP SSC Trustpoint** area, click **Generate** to generate an AP SSC trustpoint.
3. From the **RSA Key-Size** drop-down list, choose a key size.
4. From the **Signature Algorithm** drop-down list, choose an option.
5. From the **Password Type** drop-down list, choose a password type.
6. In the **Password** field, enter a password. The valid range is between 8 and 32 characters.
7. Click **Apply to Device**.

The controller creates and applies a new self-signed certificate for the AP.

Add the CA server (GUI)

Add the certificate authority server using GUI.

Register a new certificate authority (CA) server in the system using the graphical user interface.

Perform this task when you need to establish a new certificate authority for PKI management.

Procedure

1. Choose **Configuration** > **Security** > **PKI Management**.
2. In the **PKI Management** window, click the **CA Server** tab.
3. In the **CA Server** section, click the **Shutdown Status** to enable the status. If you choose the shutdown status as **Enabled**, enter and confirm the password.
4. If you choose the shutdown status as **Disabled**, you must enter the **Country Code**, **State**, **Location**, **Organization**, **Domain Name**, and **Email Address**.
5. Click **Apply** to add the CA server.
6. Click **Remove CA Server** to delete the CA server.

The certificate authority server is added and configured as specified.

Add an RSA or EC Key for PKI trustpoint (GUI)

Add an RSA or EC key for PKI trustpoint using GUI.

To enable certificate-based authentication, add an RSA or EC key to a PKI trustpoint.

Procedure

1. Choose **Configuration > Security** > **PKI Management**.
2. In the **PKI Management** window, click the **Key Pair Generation** tab.
3. In the **Key Pair Generation** section, click **Add**.
4. In the dialog box that is displayed, provide this information:

> **Note**
>
> To allow Japanese characters in **Key Name** field, ensure that the line vty configuration includes "exec-character-bits 8".

a)

> **Note**
>
> To allow Japanese characters in **Key Name** field, ensure that the line vty configuration includes "exec-character-bits 8".

In the **Key Name** field, enter the key name.

b) In the **Key Type** options, select either **RSA Key** or **EC Key**.
c) In the **Modulus Size** field, enter the modulus value for the RSA key or the EC key. The default modulus size for the RSA key is 4096 and the default value for the EC key is 521.
d) Check the **Key Exportable** check box to export the key. By default, this is checked.
e) Click **Generate**.

A new RSA or EC key is created for the PKI trustpoint and is ready for use in certificate operations.

Add and manage certificates

Add and manage certificates.

To add and manage certificates, use one of these methods.

Generate and import a certificate signing request (CSR)

Add and manage certificates.

Secure the system by adding and managing device certificates.

Perform this task when onboarding a new device, renewing expiring certificates, or updating trust relationships.

Before you begin

- When configuring a password for the .pfx file, avoid using these ASCII characters: "*, ^, (), [], \, ", and +". Using these ASCII characters results in a configuration error and prevents the certificate from being imported to the controller.
- Ensure you have the required certificate files and CA information.

Before you begin

You can add and manage certificates using either of the following methods:

Procedure

---

1. Choose **Configuration** > **Security** > **PKI Management** > **Add Certificate**.
2. Click **Generate Certificate Signing Request**.
   a) In the **Certificate Name** field, enter the certificate name.

b) From the **Key Name** drop-down list, choose an RSA key pair. (Click the plus (+) icon under the **Key Pair Generation** tab to create new RSA key pairs.).
c) Enter values in the **Country Code**, **Location**, **Organization**, **State**, **Organizational Unit**, and the **Domain Name** fields.
d) Click **Generate**.
The generated Certificate Signing Request (CSR) is displayed on the right. Click **Copy** to copy and save a local copy. Click **Save to Device** to save the generated CSR to the /bootflash/csr directory.

> **Note**
>
> If an IP address is used on the Domain Name field the controller creates the CSR without a Subject Alternative Name (SAN), since the IP address is not supported as an attribute in the SAN field of the CSR when the CSR is generated from the controller.

3. Click **Authenticate Root CA**.
   a) From the **Trustpoint** drop-down list, choose the trustpoint label generated in Step 2, or any other trustpoint label that you want to authenticate.
   b) In the **Root CA Certificate (.pem)** field, copy and paste the certificate that you have received from the CA.

   > **Note**
   >
   > Ensure that you copy and paste the PEM Base64 certificate of the issuing CA of the device certificate.

   c) Click **Authenticate**.
4. Click **Import Device Certificate**.
   a) From the **Trustpoint** drop-down list, choose the trustpoint label that was generated in Step 2, or any other trustpoint label that you want to authenticate.
   b) In the **Signed Certificate (.pem)** field, copy and paste the signed certificate that you received, from your CA.
   c) Click **Import**.

This completes the device certificate import process and the certificate can now be assigned to features.

Import a PKCS12 certificate

Import a PKCS12 certificate into the system to enable secure authentication and encryption for network communications.

Use this task when you need to install a PKCS12 certificate file for system, server, or application authentication.

The certificate can be located on any of these sources: FTP, SFTP, TFTP, SCP, or Desktop (HTTPS).

Before you begin

Obtain the PKCS12 certificate file and its password.

Procedure

---

1. Click **Import PKCS12 Certificate**.
2. From the **Transport Type** drop-down list, choose either **FTP**, **SFTP**, **TFTP**, **SCP**, or **Desktop (HTTPS)**.

Transport Type	Action
For FTP, SFTP, and SCP	enter values in the **Server IP Address (IPv4/IPv6)**, **Username**, **Password**, **Certificate File Path**, **Certificate Destination File Name**, and **Certificate Password** fields.
For TFTP	enter values in the **Server IP Address (IPv4/IPv6)**, **Certificate File Path**, **Certificate Destination File Name**, and **Certificate Password** fields.
For Desktop (HTTPS)	enter values in the **Source File Path** and **Certificate Password** fields.

3. Click **Import**.

The system imports the PKCS12 certificate. You see a confirmation message when the process completes successfully.

148 Controller Self-Signed Certificate for Wireless AP Join

Topics:

- Controller self-signed certificates for wireless AP Join

Explains how to configure and manage controller self-signed certificates to enable secure wireless access point authentication and join processes.

Controller self-signed certificates for wireless AP Join

This topic explains controller self-signed certificates, which are digital certificates generated by a wireless controller to enable secure CAPWAP communication and facilitate access point (AP) onboarding when manufacturer-installed certificates are unavailable or incompatible.

A controller self-signed certificate is a digital certificate generated and issued by a wireless controller itself that

- enables secure CAPWAP communication for access point (AP) join processes
- is used when manufacturer installed SUDI certificates are unavailable or incompatible, and
- helps ensure connectivity and secure onboarding of APs under specific scenarios.

Self-signed certificates allow Cisco Catalyst controllers to fulfill certificate requirements for AP joining and management, especially when manufacturer-installed or third-party certificates are missing or cannot be validated.

Use cases for controller self-signed certificate for wireless AP join

Lists the scenarios in which controller self-signed certificates are required for wireless AP join, including platform limitations and compatibility issues with certain APs.

You need controller self-signed certificates in specific circumstances to enable wireless APs to join the controller successfully.

Controller self-signed certificate use cases

These situations require controller self-signed certificates:

- The Cisco Catalyst 9800-CL platform is not equipped with manufacturer-installed SUDI certificates. Configure self-signed certificates on these controllers.
- Some APs running earlier software and using a Manufacturer Installed Certificate (MIC) issued by a SHA1 Cisco Trusted CA cannot join controllers with SHA2 SUDI certificates. During the CAPWAP join process, these APs display a bad certificate error and fail the DTLS handshake.

  Workaround: To support AP upgrades, configure a self-signed certificate for the controller as a workaround for the second use case. After updating all APs, delete the self-signed certificates. Then, revert to the SUDI certificate.

  > **Note**
  >
  > This workaround does not apply to Embedded Wireless Controllers running on Catalyst 9000 series switches. It is only valid for hardware appliance controllers such as Cisco Catalyst 9800-40, 9800-80, and 9800-L.

- The command **wireless management certificate ssc auth-token** is required for ensuring resilient operation of Cisco Catalyst 9800-CL controllers. It is mandatory to configure an explicit authentication token on all 9800-CL devices within the network. Additionally, this command is required when migrating from an AireOS controller to a Cisco Catalyst 9800-CL controller to facilitate secure AP join, as the 9800-CL uses a self-signed certificate instead of a MIC used by AireOS controllers.

Requirement: Use minimum RSA key size for DTLS connections

Use these guidelines when configuring DTLS connections for AP and mobility: always use certificates with an RSA key size of at least 2048 bits to ensure successful connections after a reload.

Always ensure certificates used for DTLS connections (for AP and mobility) have an RSA key size of at least 2048 bits. Using a smaller key size will cause AP and mobility connections to fail after a reload.

This principle applies to all DTLS connections involving access points (AP) and mobility features that require device certificates.

Using a minimum RSA key size of 2048 bits ensures compliance with security best practices and prevents connection failures due to insufficient key strength.

DTLS connections for AP and mobility will remain operational after a reload when certificates meet the minimum RSA key size requirement.

To verify the device certificate key size, use this command:

```
show crypto pki certificate verbose tp-name
```

Replace *tp-name* with your trustpoint name.

Prerequisites for controller self-signed certificate for wireless AP join

Follow these guidelines to prepare the controller for self-signed certificate configuration. This enables wireless AP join. Meet all prerequisites before setup.

- Ensure that the VLAN interface is up and that the IP address is reachable.
- Ensure that the **ip http server** is enabled. For more information, see Enabling HTTP Server .
- Set the **clock calendar-valid** command appropriately. For more information, see *Configure clock calendar.*
- Check whether the PKI CA server is configured. If it is, delete the existing CA server configuration.

> **Note**
>
> The **show crypto pki server** command output should not display anything.

Configure clock calendar (CLI)

Configure clock calendar using commands to ensure accurate timekeeping on the device.

Set the clock calendar to ensure accurate timekeeping on the device.

This configuration is necessary for time-sensitive operations and logging on the device.

Before you begin

Access to the device with appropriate privileges to enter global configuration mode.

Follow these steps to configure clock calendar:

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enable clock calendar.

   Example

   ```
   Device(config)# clock calendar-valid
   ```

3. Exit configuration mode.

   Example

   ```
   Device(config)# exit
   ```

The clock calendar is now configured and operational on the device.

Enable HTTP server using CLI

Configure the HTTP server to enable web-based management of the device using commands.

Enable the HTTP server to allow web-based management of the device.

This configuration is applicable for devices that require HTTP access for management purposes.

Before you begin

Ensure you have administrative access to the device.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable the HTTP server on your IP or IPv6 system.

   Example

   ```
   Device(config)# ip http server
   ```

   Enables the HTTP server on your IP or IPv6 system, including a Cisco web browser user interface. By default, the HTTP server uses the standard port 80.

3. Enable the secure HTTP server on your IP or IPv6 system.

   Example

   ```
   Device(config)# ip http secure-server
   ```

   Enables the secure HTTP server on your IP or IPv6 system, providing encrypted access to the web interface.

4. Exit configuration mode.

Example

```
Device(config)# exit
```

The HTTP server is now enabled, allowing web-based management of the device.

Configure CA server (CLI)

Configure a CA server using commands to manage certificates for devices in the network.

Deploy a CA server to manage certificates for devices in the network.

This configuration is essential for environments that require secure communication through certificates.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Generate RSA keys for the CA server.

   Example

   ```
   Device(config)# crypto key generate rsa general-keys modulus size_of_key_module
    label keypair_name
   ```

   Configures a certificate for the controller.

   When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.

   > **Note**
   >
   > The recommended key-pair name is *WLC_CA* and key modulus is *2048* bits.

3. Enable the IOS certificate server.

   Example

   ```
   Device(config)# crypto pki server certificate_server_name
   ```

   Enables IOS certificate server.

> **Note**
>
> The *certificate_server_name* must be the same name as the *keypair_name*.

4. Configure the issuer name for the CA certificate.

   Example

   ```
   Device(cs-server)# issuer-name O=Cisco Virtual Wireless LAN Controller,
   CN=CA-vWLC
   ```

   Configures X.509 distinguished name for the issuer CA certificate.

   > **Note**
   >
   > You need to configure the same *issuer-name* as suggested for AP join.

5. Grant certificate requests automatically.

   Example

   ```
   Device(cs-server)# grant auto
   ```

   Grants certificate requests automatically.

6. Optional: Specify the hash function for the signature used in the granted certificates.

   Example

   ```
   Device(cs-server)# hash sha256
   ```

   Specifies the hash function for the signature used in the granted certificates.

7. Optional: Specify the lifetime in days of a CA certificate.

   Example

   ```
   Device(cs-server)# lifetime ca-certificate time-interval
   ```

   Specifies the lifetime in days of a CA certificate.

8. Optional: Specify the lifetime in days of a granted certificate.

   Example

   ```
   Device(cs-server)# lifetime certificate time-interval
   ```

   Specifies the lifetime in days of a granted certificate.

9. Set the CA key and CA certificate archive format and password.

Example

```
Device(cs-server)# database archive pkcs12 password 0 cisco123
```

Sets the CA key and CA certificate archive format and password to encrypt the file.

10. Enable the certificate server.

Example

```
Device(cs-server)# no shutdown
```

Enables the certificate server.

> **Note**
>
> Issue this command only after you have completely configured your certificate server.

11. Return to privileged EXEC mode.

Example

```
Device(cs-server)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The CA server is now configured and ready to manage certificates for devices in the network.

Configure trustpoint (CLI)

Configure a trustpoint to enable secure communication with a Certificate Authority (CA) using commands.

Configure a trustpoint to enable secure communication with a Certificate Authority (CA).

This configuration is necessary for devices that require secure communications and certificate management.

Before you begin

Ensure you have access to the device and the necessary permissions to configure trustpoints.

Follow these steps to configure trustpoint using CLI commands:

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Generate RSA keys for the trustpoint.

   Example

   ```
   Device(config)# crypto key generate rsa exportable general-keys modulus
   size-of-the-key-modulus label label
   ```

   When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.

3. Create a new trustpoint for an external CA server.

   Example

   ```
   Device(config)# crypto pki trustpoint trustpoint_name
   ```

   Here, *trustpoint_name* refers to the trustpoint name.

   > **Note**
   >
   > Ensure that same names are used for key-pair (*label*) and *trustpoint_name*.

4. Map RSA key with that of the trustpoint.

   Example

   ```
   Device(ca-trustpoint)# rsakeypair RSA_key
   ```

   Maps RSA key with that of the trustpoint.

   - RSA_key–Refers to the RSA key pair label.
   - key_size–Refers to the signature key length. The value ranges from 360 to 4096.

5. Create subject name parameters for the trustpoint.

   Example

   ```
   Device(ca-trustpoint)# subject-name subject_name
   ```

   Specifies the subject name for the trustpoint.

6. Check revocation.

   Example

   ```
   Device(ca-trustpoint)# revocation-check none
   ```

   Specifies whether to check for certificate revocation.

7. Specify the hash algorithm.

Example

```
Device(ca-trustpoint)# hash sha256
```

Specifies the hash algorithm to be used.

8. Specify the serial number.

Example

```
Device(ca-trustpoint)# serial-number
```

Specifies the serial number for the trustpoint.

9. Optional: Set certificate key-usage purpose.

Example

```
Device(ca-trustpoint)# eku request server-auth client-auth
```

Sets the extended key usage for the certificate.

10. Enable password for the trustpoint.

Example

```
Device(ca-trustpoint)# password 0 password
```

11. Enroll the URL for the CA server.

Example

```
Device(ca-trustpoint)# enrollment url url
```

Replace the dummy IP with management VLAN interface IP of the controller where CA server is configured.

12. Exit the configuration mode.

Example

```
Device(ca-trustpoint)# end
```

The trustpoint is now configured and ready for use with the CA server.

Authenticate and enroll the PKI trustpoint with CA server

Configure the PKI trustpoint to authenticate and enroll with the Certification Authority (CA) server using commands.

Authenticate and enroll the PKI trustpoint to establish a secure connection with the Certification Authority (CA) server.

This procedure is used in environments where a PKI trustpoint needs to be authenticated and enrolled with a Certification Authority (CA) server for certificate management.

Before you begin

Ensure that the Certification Authority (CA) server is reachable and the trustpoint is properly configured.

Follow these steps to authenticate and enroll the PKI trustpoint with CA server:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Authenticate the PKI trustpoint with the CA server.

   Example

   ```
   Device(config)# crypto pki authenticate trustpoint-name
   Certificate has the following attributes:
   Fingerprint MD5: 64C5FC9A C581D827 C25FC3CF 1A7F42AC
   Fingerprint SHA1: 6FAFF812 7C552783
   6A8FB566 52D95849 CC2FC050
   % Do you accept this certificate? [yes/no]: yes
   Trustpoint CA certificate accepted.
   ```

   Fetches the CA certificate.

3. Enroll the PKI trustpoint for a client certificate.

   Example

   ```
   Device(config)# crypto pki enroll trustpoint-name
   Enter following answers for UI interaction:
   % Include an IP address in the subject name? [no]: no
   Request certificate from CA? [yes/no]: yes
   ```

4. Exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

   Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The PKI trustpoint is now authenticated and enrolled with the Certification Authority (CA) server, allowing for secure certificate management.

Tag wireless management trustpoint name

Configure a wireless management trustpoint name for identification and management purposes using commands.

Tag a wireless management trustpoint name for identification and management purposes.

This configuration is used in environments where wireless management trustpoints need to be clearly identified for management and security.

Before you begin

Ensure you have the necessary privileges to enter global configuration mode.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Tag the wireless management trustpoint name.

   Example

   ```
   Device(config)# wireless management trustpoint trustpoint-name
   ```

3. Exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

   Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The wireless management trustpoint name is successfully tagged and can be used for management purposes.

Disable PKI server

Configure the PKI server to be disabled and prevent it from issuing certificates using commands.

Disable the PKI server to prevent it from issuing certificates.

This procedure is used in scenarios where the PKI server is no longer needed or must be temporarily disabled.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a new server for an external CA server.

   Example

   ```
   Device(config)# crypto pki server certificate-server-name
   ```

   The *certificate-server-name* refers to the server name.

> **Note**
>
> Here, server_name refers to the server name.

3. Disable the certificate server.

   Example

   ```
   Device(cs-server)# shutdown
   ```

   Update purpose string to \

4. Return to privileged EXEC mode.

   Example

   ```
   Device(cs-server)# end
   ```

   Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The PKI server is now disabled and will not issue certificates.

Verify controller certificates for wireless AP join

Lists commands to verify controller certificates for wireless AP join.

Use these commands to verify controller certificates for wireless access point (AP) join.

To view the CA server details, use the command:

```
Device# show crypto pki server
Certificate Server WLC_CA:
Status: enabled
State: enabled
Server's configuration is locked (enter "shut" to unlock it)
Issuer name: O=Cisco Virtual Wireless LAN Controller, CN=CA-vWLC
CA cert fingerprint: 79A3DBD5 59A7E384 73ABD152 C133F4E2
Granting mode is: auto
Last certificate issued serial number (hex): 1
CA certificate expiration timer: 12:04:00 UTC Mar 8 2029
CRL NextUpdate timer: 18:04:00 UTC Mar 11 2019
Current primary storage dir: nvram:
Database Level: Minimum - no cert data written to storage
```

To view the trustpoint details, use this command:

```
Device# show crypto pki trustpoint ewlc-tp1 status
Trustpoint ewlc-tp1:
...
State:
Keys generated ..... Yes (General Purpose, exportable)
```

```
Issuing CA authenticated ....... Yes
Certificate request(s) ..... Yes
```

To view the wireless management trustpoint details, use this command:

```
Device# do show wireless management trustpoint
Trustpoint Name : ewlc-tp1
Certificate Info : Available
Certificate Type : SSC
Certificate Hash : 4a5d777c5b2071c17faef376febc08398702184e
Private key Info : Available
FIPS suitability : Not Applicable
```

To view the Hypertext Transfer Protocol (HTTP) server status, use this command:

```
Device# show ip http server status | include server status
HTTP server status: Enabled
HTTP secure server status: Enabled
```

149 Managing Rogue Devices

Topics:

- Rogue devices and detection
- Wireless Service Assurance (WSA) rogue events
- Rogue AP scale modes per class
- Rogue PMF containment

This chapter explains configuring and operating rogue device detection and containment on Cisco wireless controllers—covering RLDP, management frame protection, AP authentication, off-channel PMF, WSA event reporting, and scale quotas/priorities.

Rogue devices and detection

Introduces rogue device concepts, covering detection techniques, containment strategies, integration with Cisco Prime Infrastructure, MFP-based and off-channel PMF containment, AP impersonation detection, security level configuration, management frame protection, policy configuration, and provides verification procedures for comprehensive rogue threat management.

A rogue device is any unauthorized network device.

A rogue device can:

- disrupt wireless LAN operations by hijacking legitimate clients,
- facilitate attacks such as plaintext, denial-of-service, and man-in-the-middle attacks on wireless networks, and
- pose serious security risks by allowing unauthorized access, interception, and breaches inside the corporate firewall.

Risks and impact on network security

Rogue access points are a common form of rogue devices. Hackers can use rogue access points to capture sensitive information such as usernames and passwords. By transmitting a series of Clear to Send (CTS) frames, a rogue access point can mimic a legitimate access point. This action instructs a specific client to transmit while forcing other clients to wait, which prevents legitimate clients from accessing network resources.

Ban rogue access points from the air space to protect users and maintain network integrity.

Because rogue access points are inexpensive and available, staff may connect unauthorized access points to existing LANs. This can create ad hoc wireless networks without approval from IT departments.

These rogue access points can create serious security risks because they may connect inside the corporate firewall. If security settings are disabled on these devices, unauthorized users can intercept network traffic and hijack client sessions. When wireless users connect to rogue access points within the enterprise network, the risk of a security breach increases.

Rogue client status change

The controller marks the rogue client as a threat if a wireless client in the RUN state has the same MAC address.

Restrictions on rogue detection

- Rogue containment is not supported on DFS channels.

Rogue AP containment and attack signatures

Use this concept to understand how rogue AP containment operates and the types of attack signatures relevant for wireless security, including mechanisms introduced in release 17.7.1.

A rogue AP containment and attack signature is a set of wireless network security mechanisms that

- detect and contain unauthorized rogue wireless APs,
- use automated or manual methods to select the best access point for containment actions, and
- define specific attack signatures, including behaviors of rogue AP impersonators and new threats based on beacon frames.

Containment operation

A rogue access point is moved to a contained state either automatically or manually. The controller selects the best available access point (AP) for containment and pushes the containment information to it. Each AP stores its list of containments per radio.

For auto-containment, you can configure the controller to use only APs in monitor mode. Containment operations occur through two primary methods:

- The container AP periodically reviews its containment list and sends unicast containment frames. For rogue AP containment, these are sent only if a rogue client is associated.
- When the system detects contained rogue activity, the AP immediately transmits containment frames.

Individual rogue containment involves sending a sequence of unicast disassociation and deauthentication frames.

Attack signature examples

These signatures help identify sophisticated attack methods targeting wireless networks and support more effective containment and remediation strategies.

- **Beacon DS Attack** –When managed and rogue APs use the same BSSID, the rogue APs are termed as impersonators. An attacker can add the Direct-Sequence parameter set information element with any channel number. If the added channel number is different from the channel number used by the managed AP, the attack is termed as Beacon DS Attack.
- **Beacon Wrong Channel** –When managed and rogue APs use the same BSSID, the rogue APs are termed as AP impersonators. If an AP impersonator uses a channel number that is different from the one used by the managed AP with the same BSSID, the attack is termed as Beacon Wrong Channel. In such a case, the Direct-Sequence Information Element might not even be present in the Beacon frame.

Cisco Prime Infrastructure interaction and rogue detection

Cisco Prime Infrastructure uses rule-based classification and interacts with the controller to detect and manage rogue access points based on configured classification rules and rogue states.

A rogue access point classification rule is a network security mechanism that

- applies predefined criteria to evaluate and categorize access points detected on the network,
- determines the current state of each detected access point (such as Friendly, Malicious, Internal, or External), and
- defines when and how the controller communicates rogue events to Cisco Prime Infrastructure.

How Cisco Prime Infrastructure interacts with the controller for rogue detection

Cisco Prime Infrastructure interacts with controllers to detect and manage rogue access points according to the classification rules set on the controllers. The rule-based classification enables Cisco Prime Infrastructure to receive detailed trap notifications when rogue access point events occur. Key interactions and behaviors include:

- The controller uses its configured classification rules to determine the state of each rogue access point.
- When certain rogue access point events occur (such as state changes or rogue entry removal), the controller sends traps (notifications) to Cisco Prime Infrastructure.
- Trap notifications depend on both the rogue state and the event type, including these cases:
  - If an unknown access point moves to the Friendly state for the first time and the rogue state is Alert, a trap is sent.
  - No trap is sent if the rogue state is Internal or External.

  - If a rogue entry is removed after timeout, the controller sends a trap for entries classified as Malicious (Alert, Threat) or Unclassified (Alert).
  - The controller does not remove rogue entries with the following states: Contained, Contained Pending, Internal, and External.

- When a new, unknown access point is detected and moves to the Friendly state under Alert, Cisco Prime Infrastructure receives a notification.
- If a rogue entry with a Malicious or Unclassified (Alert) state is removed after a timeout, a trap is generated and sent to Cisco Prime Infrastructure.
- Rogue entries in Contained, Contained Pending, Internal, and External states are retained by the controller and not removed automatically, so no removal trap is sent for them.

Rogue containment (Protected Management Frames (PMF) enabled)

Information about rogue containment.

Starting with Cisco IOS XE 17.3.1, the system does not contain rogue devices enabled with 802.11w Protected Management Frames (PMF). Instead, the system marks the rogue device as *Contained Pending*, and raises a Web Security Appliance (WSA) alarm for the *Contained Pending* event. Skipping device containment prevents unnecessary use of AP resources.

Run the **show wireless wps rogue ap detailed** command to verify device containment when PMF is enabled on a rogue device.

> **Note**
>
> This feature is supported only on Wave 2 APs.

Off-channel PMF rogue containment

This content describes the conceptual information about off-channel PMF rogue containment.

A off-channel PMF rogue containment is a wireless security technique that

- detects rogue access points using off-channel or non-serving radios
- enables access points to report rogue APs to a central controller, and
- allows continuous monitoring and containment of rogue APs without disrupting client service on the primary radio.

The containment process involves three steps:

1. An AP detects a rogue AP and reports its details to the controller.
2. The controller sends a containment request specifying the off-channel PMF containment type.
3. The AP validates the request and contains the rogue AP using designated off-channel radios.

The **pmf-offchannel** command is introduced in the AP profile of the controller to enable off-channel PMF rogue containment.

If a rogue AP is detected operating on a frequency not currently used by client devices, the Cisco AP can use its off-channel radio to monitor and contain that rogue AP, minimizing interruption to legitimate wireless clients.

Standard rogue containment only uses the serving radio, which can interfere with client service. Off-channel PMF rogue containment specifically avoids disruption by using non-serving or dedicated radios.

Limitations for off-channel PMF rogue containment

Describes the limitations of off-channel PMF rogue containment.

The limitations for off-channel PMF rogue containment are:

- You can only contain rogue clients that are not in power-saving mode.
- You cannot contain rogue clients on Dynamic Frequency Selection (DFS), or 5 gigahertz (GHz).
- You cannot contain rogue clients on the 6 gigahertz (GHz) band.
- After you contain rogue clients, they might re-associate with the rogue Service Set Identifier (SSID).
- Containment depends on client behavior and may not work if future changes are made to client wireless drivers.

AP impersonation detection

Access Point (AP) impersonation is a wireless security attack where an attacker mimics a legitimate access point to intercept and manipulate communication between devices and the network.

An AP impersonation is a wireless security attack that

- allows a rogue device to masquerade as a legitimate AP
- enables attackers to intercept and manipulate wireless communication between client devices and the network, and
- threatens the confidentiality, integrity, and security of wireless network traffic.

Detection methods

The various methods to detect AP impersonation are:

- You can detect AP impersonation if a managed AP reports itself as Rogue. This method is always enabled and does not require configuration.
- AP impersonation detection uses Management Frame Protection (MFP).
- AP impersonation detection uses AP authentication.

Management Frame Protection (MFP)-based detection

Infrastructure MFP protects 802.11 session management by adding Message Integrity Check (MIC) elements to management frames sent by APs (not clients). Other APs in the network then validate these management frames.

- If infrastructure MFP is enabled, managed APs check whether the MIC elements are present and valid.
- If either condition is not met, the managed AP sends rogue AP reports with an updated AP authentication failure counter field.

AP authentication-based detection

When you enable AP authentication, the controller creates an AP domain secret and shares it with all APs in the same network. This process enables APs to authenticate each other.

- An AP authentication information element is attached to beacon and probe response frames.
- If the AP authentication information element has an incorrect signature, an off timestamp, or the information element is missing, the AP that detects the condition increments the **AP authentication failure count** field.

- An impersonation alarm is raised after the **AP authentication failure count** field exceeds its threshold.
- The rogue AP is classified as **Malicious** with the state **Threat**.

Run the **show wireless wps rogue ap detail** command to see when AP impersonation is detected as a result of authentication errors.

Configuration notes

- Run the **CCX Aironet-IESupport** command in all WLAN procedures to prevent the BSSID from being detected as a rogue.
- For AP impersonation detection, Network Time Protocol (NTP) must be enabled under the AP profile. CAPWAP-based time is not sufficient.

Rogue detection security level

The rogue detection security level configuration allows you to set rogue detection parameters for different deployment needs.

A rogue detection security level is a configuration preset that

- determines the sensitivity and scope of rogue wireless device detection
- restricts or allows configuration of specific detection parameters, and
- provides predefined or customizable options for different deployment needs.

- Rogue detection: identifies unauthorized or unknown wireless devices in a network environment.
- Security level: specifies a preset combination of parameters for rogue detection.

The system provides four rogue detection security levels.

- Critical: Provides basic rogue detection for highly sensitive deployments. Fixed configuration parameters ensure maximum security and consistency.
- High: Provides basic rogue detection suitable for medium-scale environments. Several parameters are fixed to balance protection and operational simplicity.
- Low: Provides basic rogue detection suitable for small-scale deployments. Fixed parameters provide easy management.
- Custom: The default security level. You can fully configure all rogue detection parameters to suit any environment.

> **Note**
>
> To modify all parameters, select the Custom security level. The critical, high, and low levels have fixed settings.

Table 121: Rogue Detection: Predefined Levels

Parameter	Critical	High	Low
Cleanup Timer	3600 seconds (1 hour)	1200 seconds (20 minutes)	240 seconds (4 minutes)
AAA Validate Clients	Disabled	Disabled	Disabled
AAA Validate AP	Disabled	Disabled	Disabled

Parameter	Critical	High	Low
Adhoc Reporting	Enabled	Enabled	Enabled
Monitor Mode Report Interval	10 seconds (0:10)	30 seconds (0:30)	60 seconds (1:00)
Minimum RSSI	-128 dBm	-80 dBm	-80 dBm
Transient Interval	600 seconds (10 minutes)	300 seconds (5 minutes)	120 seconds (2 minutes)
Auto Contain **This feature works only on Monitor Mode APs.**	Disabled	Disabled	Disabled
Auto Contain Level	1	1	1
Auto Contain Same SSID	Disabled	Disabled	Disabled
Auto Contain Valid Clients on Rogue AP	Disabled	Disabled	Disabled
Auto Contain Adhoc	Disabled	Disabled	Disabled
Containment Auto Rate	Enabled	Enabled	Enabled
Validate Clients with Cisco Connected Mobile Experiences (CMX)	Enabled	Enabled	Enabled
Containment FlexConnect	Enabled	Enabled	Enabled

You can configure all these parameters in the Custom security level.

- A hospital implements the Critical security level to maintain rigorous control over rogue detection with fixed settings.
- A small business chooses the Low security level for straightforward rogue detection with minimal configuration.
- An enterprise IT team uses the Custom security level to tailor all rogue detection parameters to their unique requirements.

Set rogue detection security level (CLI)

Set the rogue detection security level.

Set the wireless rogue detection security level for your network deployment.

Before you begin

Use these steps to set the rogue detection security level.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the rogue detection security level to custom.

Example

```
Device(config)# wireless wps rogue security-level custom
```

3. Configure the rogue detection security level for small-scale deployments.

Example

```
Device(config)# wireless wps rogue security-level low
```

4. Configure the rogue detection security level for medium-scale deployments.

Example

```
Device(config)# wireless wps rogue security-level high
```

5. Configure the rogue detection security level for highly sensitive deployments.

Example

```
Device(config)# wireless wps rogue security-level critical
```

The device applies the selected rogue detection security level to enhance wireless intrusion protection for your deployment.

Configuring rogue detection (GUI)

Configure rogue detection using the GUI.

Enable rogue access point detection and set parameters in the GUI.

Use rogue detection to identify and manage unauthorized or suspicious APs in your network. Complete this task when you establish or update your wireless security policies.

Before you begin

Use these steps to configure rogue detection using the GUI:

Procedure

---

1. Choose Configuration, then Tags and Profiles, then AP Join.
2. Click the **AP Join Profile Name** to edit the access point (AP) join profile properties.
3. In the **Edit AP Join Profile** window, click the **Rogue AP** tab.
4. Check the **Rogue Detection** check box to enable rogue detection.
5. In the **Rogue Detection Minimum RSSI** field, enter the RSSI value.
6. In the **Rogue Detection Transient Interval** field, enter the interval in seconds (minutes).
7. In the **Rogue Detection Report Interval** field, enter the report interval value in seconds (minutes).

8. In the **Rogue Detection Client Number Threshold** field, enter the threshold for rogue client detection.
9. Check the **Auto Containment on FlexConnect Standalone** check box to enable auto containment.
10. Click **Update and Apply to Device**.

The rogue-detection feature is activated with your configured parameters. This helps you identify and contain unauthorized access points (APs) in the network.

Configure rogue detection (CLI)

Configure rogue detection using CLI commands.

You enable and customize rogue detection on Cisco wireless access points using specific CLI commands.

Use these commands to detect and contain unauthorized access points (APs) and improve wireless network security.

Before you begin

Use these steps to configure rogue detection.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Set the minimum RSSI value for APs to detect rogues and create entries in the system.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

   ```
   Device(config)# rogue detection rssi in dBm
   ```

   You can enter an RSSI value from -128 to -70 dBm. The default is -128 dBm.

   Note

   You can use this feature with any AP mode.

   Many rogues have weak RSSI values and do not provide useful information for rogue analysis. Specify a minimum RSSI value to filter these rogues.

3. Choose a rogue containment option.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

   ```
   Device(config)# rogue detection containment flex-rate
   ```

The **auto-rate** option contains rogues automatically. The **flex-rate** option contains standalone FlexConnect APs.

4. Turn on rogue detection for all APs.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

   ```
   Device(config)# rogue detection enable
   ```

5. Set the interval for rogue reports on monitor mode APs.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

   ```
   Device(config)# rogue detection report-interval time in seconds
   ```

   The valid range for the reporting interval is 10 to 300 seconds.

   If the controller detects thousands of rogue APs, the PUBD (Public Utility Bulletin Daemon) process may cause sustained high CPU usage. Increase the Rogue Detection Report Interval to a value higher than the default of 10 to resolve this issue.

Rogue detection is active using your specified parameters on the AP profile. This improves security by monitoring and containing unauthorized devices.

Configure RSSI deviation notification threshold for rogue APs (CLI)

Configure the RSSI deviation notification threshold for rogue APs using CLI commands.

Set the signal strength deviation threshold to trigger notifications for rogue access points on your network.

Before you begin

Use these steps to configure the RSSI deviation notification threshold for rogue APs.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the RSSI deviation notification threshold for rogue APs.

   Example

   ```
   Device(config)# wireless wps rogue ap notify-rssi-deviation
   ```

3. Return the system to privileged EXEC mode.

Example

```
Device(config)# end
```

You can press Ctrl and Z to exit global configuration mode.

The system notifies you when a rogue APs RSSI deviation exceeds the set threshold.

Configure management frame protection (GUI)

Configure management frame protection using the GUI.

Enable and configure management frame protection (MFP) to secure wireless network communications against attacks such as rogue AP impersonation.

Before you begin

Use these steps to configure management frame protection.

Procedure

---

1. Choose **Configuration**, then **Security**, then **Wireless Protection Policies**.
2. In the **Rogue Policy** tab, under the **MFP Configuration** section, check the **Global MFP State** check box to enable the global MFP state.
3. Check the **AP Impersonation Detection** check box to enable AP impersonation detection.
4. In the **MFP Key Refresh Interval** field, specify the refresh interval in hours.
5. Click **Apply**.

Management frame protection is enabled and configured, providing enhanced security for wireless management frames.

Configure Management Frame Protection (CLI)

Configure management frame protection using command-line interface (CLI) commands.

Configure Management Frame Protection (MFP) on a device using the command-line interface (CLI).

Before you begin

Use these steps to configure management frame protection.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure management frame protection.

   Example

   ```
   Device(config)# wireless wps mfp
   ```

3. Configure AP impersonation detection or set the MFP key refresh interval in hours.

   Example

   ```
   Device(config)# wireless wps mfp ap-impersonation
   ```

   ```
   Device(config)# wireless wps mfp key-refresh-interval
   ```

   key-refresh-interval: Set sthe MFP key refresh interval in hours.

   The valid range is one to 24 hours. The default value is 24 hours.

4. Save the configuration, exit configuration mode, and return to privileged **EXEC** mode.

   Example

   ```
   Device(config)# end
   ```

The device protects wireless management traffic according to the configured settings.

Enable AP authentication

Enable access point authentication on a wireless controller.

Set up AP authentication on a wireless controller. This enhances network security by authenticating APs and by configuring threshold values for authentication failures.

Before you begin

Use these steps to enable AP authentication.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the wireless Wi-Fi Protected Setup (WPS) AP authentication.

   Example

   ```
   Device(config)# wireless wps ap-authentication
   ```

3. Configure AP neighbor authentication and set the threshold for AP authentication failures.

   Example

   ```
   Device(config)# wireless wps ap-authentication threshold threshold
   ```

4. Configure a WLAN.

Example

```
Device(config)# wlan wlan-namewlan-idSSID-name
```

5. Enable support for Aironet information elements on this Wireless Local Area Network (WLAN).

Example

```
Device(config-wlan)# ccx aironet-iesupport
```

6. Return to privileged EXEC mode.

Example

```
Device# end
```

AP authentication is enabled on the wireless controller. Authentication thresholds and Aironet IE support are configured for the specified WLAN.

Configure off-channel PMF rogue containment (GUI)

This task explains how to enable off-channel PMF rogue containment using the GUI.

This task enables off-channel Protected Management Frame (PMF) rogue containment on an AP policy profile.

Before you begin

Perform the steps in this section to configure off-channel PMF rogue containment.

Procedure

---

1. Choose **Configuration**, then **Tags & Profiles**, then **AP Join**.
2. Click **Add**.

   The window labeled **Add AP Join Profile** appears.
3. In the **Add AP Join Profile** window, click the **General** tab and enter the profile name.
4. Click the **Security** tab.
5. Under the **Rogues** section, check the **Rogue Containment PMF-Offchannel** check box.
6. Click **Apply to Device**.

Off-channel PMF rogue containment is enabled on the AP policy profile.

Configure off-channel PMF rogue containment (CLI)

This topic explains how to configure off-channel PMF containment using the CLI.

Set up off-channel Protected Management Frame (PMF) rogue AP containment on APs using CLI commands.

Before you begin

Use these steps to configure off-channel PMF rogue containment on APs.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter a profile name to create an AP profile.

   Example

   ```
   Device(config)# ap profile ap-pmf-off-channel
   ```

3. Enable PMF-denial for rogue AP containment.

   Example

   ```
   Device(config-ap-profile)# rogue detection containment pmf-denial
   ```

   Note

   The **pmf-deauth** feature is enabled by default when **pmf-denial** is enabled. If you do not want **pmf-deauth** , then you must disable it explicitly.

4. Enable PMF-denial type off-channel PMF rogue AP containment.

   Example

   ```
   Device(config-pmf-denial)# pmf-offchannel
   ```

Off-channel PMF containment for rogue APs is enabled on the configured AP profile.

Verify management frame protection

Verify management frame protection on a wireless controller.

To verify if the Management Frame Protection (MFP) feature is enabled or not, use this command:

```
Device# show wireless wps summary
Client Exclusion Policy
  Excessive 802.11-association failures   : unknown
  Excessive 802.11-authentication failures: unknown
  Excessive 802.1x-authentication         : unknown
  IP-theft                                : unknown
  Excessive Web authentication failure    : unknown
  Failed Qos Policy                       : unknown

Management Frame Protection
  Global Infrastructure MFP state : Enabled
```

```
  AP Impersonation detection      : Disabled
  Key refresh interval            : 15
```

To view the MFP details, use this command:

```
Device# show wireless wps mfp summary
Management Frame Protection
  Global Infrastructure MFP state : Enabled
  AP Impersonation detection      : Disabled
  Key refresh interval            : 15
```

Verify rogue events

Show command to verify rogue events.

To verify the rogue event history, run the **show wireless wps rogue ap detailed** command:

```
Device# show wireless wps rogue ap detailed
Rogue Event history

Timestamp                   #Times   Class/State Event                Ctx
                  RC
--------------------------- -------- ----------- --------------------
------------------------- ----
05/10/2021 13:56:46.657434 2        Mal/Threat  FSM_GOTO
           Threat  0x0
05/10/2021 13:56:46.654905 1        Unk/Init    EXPIRE_TIMER_START
             240s  0x0
05/10/2021 13:56:46.654879 1        Unk/Init    AP_IMPERSONATION
DS:1,ch:1,band_id:0  0x0
05/10/2021 13:56:46.654673 1        Unk/Init    RECV_REPORT
 70db.98fc.2680/0  0x0
05/10/2021 13:56:46.654663 1        Unk/Init    INIT_TIMER_START
             180s  0x0
05/10/2021 13:56:46.654608 1        Unk/Init    CREATE
                   0x0


Rogue BSSID                          : 002c.c8c1.096d
Last heard Rogue SSID                : MarvellAP0d
802.11w PMF required                 : No
Is Rogue an impersonator             : Yes
Beacon Wrong Channel                 : Yes
Beacon DS Attack                     : Yes
Is Rogue on Wired Network            : No
Classification                       : Malicious
Manually Contained                   : No
State                                : Threat
First Time Rogue was Reported        : 05/10/2021 13:56:46
Last Time Rogue was Reported         : 05/10/2021 13:56:46

Number of clients                    : 0
```

Verify rogue detection

Command to verify rogue detection on the device.

This section describes the new command for rogue detection.

These commands can be used to verify rogue detection on the device.

Table 122: Verifying Adhoc rogues information

Command	Purpose
show wireless wps rogue adhoc detailed *mac_address*	Displays the detailed information for an Adhoc rogue.
show wireless wps rogue adhoc summary	Displays a list of all Adhoc rogues.

Table 123: Verifying rogue AP information

Command	Purpose
show wireless wps rogue ap clients *mac_address*	Displays the list of all rogue clients associated with a rogue.
show wireless wps rogue ap custom summary	Displays the custom rogue AP information.
show wireless wps rogue ap detailed *mac_address*	Displays the detailed information for a rogue AP.
show wireless wps rogue ap friendly summary	Displays the friendly rogue AP information.
show wireless wps rogue ap list *mac_address*	Displays the list of rogue APs detected by a given AP.
show wireless wps rogue ap malicious summary	Displays the malicious rogue AP information.
show wireless wps rogue ap summary	Displays a list of all Rogue APs.
show wireless wps rogue ap unclassified summary	Displays the unclassified rogue AP information.

Table 124: Verifying Rogue Auto-Containment Information

Command	Purpose
show wireless wps rogue auto-contain	Displays the rogue auto-containment information.

Table 125: Verifying Classification Rule Information

Command	Purpose
show wireless wps rogue rule detailed *rule_name*	Displays the detailed information for a classification rule.
show wireless wps rogue rule summary	Displays the list of all rogue rules.

Table 126: Verifying Rogue Statistics

Command	Purpose
show wireless wps rogue stats	Displays the rogue statistics.

Table 127: Verifying Rogue Client Information

Command	Purpose
show wireless wps rogue client detailed *mac_address*	Displays detailed information for a Rogue client.
show wireless wps rogue client summary	Displays a list of all the Rogue clients.

Table 128: Verifying Rogue Ignore List

Command	Purpose
show wireless wps rogue ignore-list	Displays the rogue ignore list.

Verify off-channel PMF rogue containment

Describes the various show commands and their outputs during the verification of the off-channel PMF containment.

Verify rogue containment summary

To verify rogue containment summary, use this command:

```
Device# show wireless wps rogue containment summary
Rogue Containment activities for each managed AP

AP: 6XXX.b4XX.2aXX  Slot: 1
  Active Containments   : 3
   Containment Mode     : OFFCHAN_PMF
   Rogue AP MAC         : 68XX.b4XX.2aXX
   Containment Channels : 40
                                                0.0.0.0
   372           64               AGL          Enabled
```

Verify rogue statistics

To display rogue statistics, use this command:

```
Device# show wireless wps rogue stats
 Stats
  Alert                         : 256
  Internal                      : 0
  External                      : 0
  Contained                     : 1
  Containment-pending           : 0
  Threat                        : 0
  Pending                       : 0
Rogue Clients
  Total/Max Scale               : 20/16000
  Contained                     : 0
  Containment-pending           : 0
.
.
.
                     0.0.0.0                          372           64
        AGL          Enabled
```

Examples: rogue detection onfiguration

Example for rogue detection configuration.

This example shows how to configure the minimum RSSI that a detected rogue AP needs to be at, to have an entry created in the device:

```
Device# configure terminal
Device(config)# ap profile profile1
```

```
Device(config)# rogue detection min-rssi -100
Device(config)# end
Device# show wireless wps rogue client summary/show wireless wps rogue ap
summary
```

This example shows how to configure the classification interval:

```
Device# configure terminal
Device(config)# ap profile profile1
Device(config)# rogue detection min-transient-time 500
Device(config)# end
Device# show wireless wps rogue client summary/show wireless wps rogue ap
summary
```

Configure rogue policies (GUI)

You can configure rogue policies using the GUI.

Use this task to define and customize rogue wireless protection policies. These policies help the system detect and respond to unauthorized wireless activity.

Before you begin

Perform the steps in this section to configure rogue policies.

Procedure

1. Choose **Configuration**, then **Security**, then **Wireless Protection Policies**.
2. In the **Rogue Policies** tab, select the security level from the **Rogue Detection Security Level** drop-down.
3. In the **Expiration timeout for Rogue APs** field, enter the timeout value in seconds.
4. Select the **Validate Rogue Clients against AAA** check box to validate rogue clients using the AAA server.
5. Select the **Validate Rogue APs against AAA** check box to validate rogue access points using the AAA server.
6. In the **Rogue Polling Interval** field, enter the interval in seconds at which the system polls the AAA server for rogue information.
7. Select the **Detect and Report Adhoc Networks** check box to enable detection of rogue ad hoc networks.
8. In the **Rogue Detection Client Number Threshold** field, enter the number of clients at which the system generates an SNMP trap.
9. In the **Auto Contain** section, enter these details.
10. Select the containment level from the **Auto Containment Level** drop-down.
11. Select the **Auto Containment only for Monitor Mode APs** check box to limit automatic containment to monitor-mode APs.
12. Select the **Using our SSID** check box to limit automatic containment to rogue APs that use an SSID that is configured on the controller.
13. Select the **Adhoc Rogue AP** check box to enable automatic containment for ad hoc rogue APs.
14. Click **Apply**.

The system updates rogue policies to enhance detection and containment of unauthorized wireless threats according to your configuration.

Configure rogue policies (CLI)

Use this procedure to configure rogue policies using commands.

Establish rogue policies to enhance network security by managing rogue access points and clients.

This configuration is essential in environments where rogue devices may pose a security threat to the network.

Before you begin

Ensure you have access to the device's global configuration mode.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. Configure the rogue detection security level.

   Example

   ```
   Device(config)# wireless wps rogue security-level security-level
   ```

   You can set the *security-level* to be **critical** for highly sensitive deployments, **custom** for customizable security level, **high** for medium-scale deployments, and **low** for small-scale deployments.

3. Configure the expiration time for rogue entries.

   Example

   ```
   Device(config)# wireless wps rogue ap timeout timout-in-seconds
   ```

   Valid range for the time in seconds is 240 seconds to 3600 seconds.

4. Configure the use of AAA or local database to detect valid MAC addresses.

   Example

   ```
   Device(config)# wireless wps rogue client aaa
   ```

   Configures the use of AAA or local database to detect valid MAC addresses.

5. Configure the use of MSE to detect valid MAC addresses.

   Example

   ```
   Device(config)# wireless wps rogue client mse
   ```

   Configures the use of MSE to detect valid MAC addresses.

6. Configure the minimum RSSI notification threshold for rogue clients.

Example

```
Device(config)# wireless wps rogue client notify-min-rssi RSSI-threshold
```

Valid range for the RSSI threshold in dB is -128 dB to -70 dB.

7. Configure the RSSI deviation notification threshold for rogue clients.

Example

```
Device(config)# wireless wps rogue client notify-min-deviation
RSSI-threshold-value
```

Valid range for the RSSI threshold in dB is 0 dB to 10 dB.

8. Configure the use of AAA or local database to classify rogue AP based on rogue AP MAC addresses.

Example

```
Device(config)# wireless wps rogue ap aaa
```

Configures the use of AAA or local database to classify rogue AP based on rogue AP MAC addresses.

9. Configure rogue AP AAA validation interval.

Example

```
Device(config)# wireless wps rogue ap aaa polling-interval
AP-AAA-interval-value
```

The valid range for the AP AAA interval in seconds is 60 seconds to 86400 seconds.

10. Enable detecting and reporting adhoc rogue (IBSS).

Example

```
Device(config)# wireless wps rogue adhoc
```

Enables detecting and reporting adhoc rogue (IBSS).

11. Configure the rogue client per a rogue AP SNMP trap threshold.

Example

```
Device(config)# wireless wps rogue client client-threshold threshold-value
```

The valid range for the threshold is 0 to 256.

12. Configure the init timer for rogue APs.

Example

```
Device(config)# wireless wps rogue ap init-timer timer-value
```

The default timer value is set to 180 seconds.

> **Note**
>
> When a rogue AP is detected, an init timer is started and the rules are applied when this timer expires. This allows for rogue AP information to stabilize before applying any rules. However, you can change the value of this timer using this command. For instance, the init timer can be set to 0, if the rules need to be applied as soon as a new rogue AP is detected.

The rogue policies are now configured, enhancing the network's security against rogue devices.

Wireless Service Assurance (WSA) rogue events

Explains wireless service assurance rogue events, including event monitoring and configuration steps, to enhance visibility and proactive management of rogue devices in the wireless network.

Wireless Service Assurance (WSA) rogue events are telemetry notifications that replicate the information of corresponding SNMP traps. Support is available in Release 16.12.x, where x denotes a release version.

- You receive details such as the MAC address of the rogue AP.
- You receive information about the managed AP and the radio that detected the rogue AP with the strongest RSSI.
- You receive event-specific data, such as SSID; channel for potential honeypot events; and MAC address of the impersonating AP for impersonation events.

WSA Rogue Events: Details and Support

For all exported events, these details are provided to the wireless service assurance (WSA) infrastructure:

- MAC address of the rogue AP
- Details of the managed AP and the radio that detected the rogue AP with the strongest RSSI
- Event-specific data such as SSID; channel for potential honeypot events; and MAC address of the impersonating AP for impersonation events.

You can scale the WSA rogue events feature up to four times the maximum number of supported access points (APs). You can also scale it to one-half of the maximum number of supported clients.

The WSA rogue events feature is supported on Cisco Catalyst Center and other third-party infrastructure.

Wireless service assurance rogue events

Configure wireless service assurance rogue events.

Configure your wireless device to send service assurance rogue event data to the event queue.

Before you begin

To configure wireless service assurance for rogue events, complete these steps.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enable wireless service assurance.

   Example

   ```
   Device# network-assurance enable
   ```

3. Enable wireless service assurance for rogue devices.

   Example

   ```
   Device# wireless wps rogue network-assurance enable
   ```

   This ensures that the wireless service assurance (WSA) rogue events are sent to the event queue.

Your device sends wireless service assurance rogue events to the event queue for enhanced monitoring.

Monitor wireless service assurance rogue events

View wireless service assurance (WSA) rogue event statistics and details.

Before you begin

Use these steps to monitor wireless service assurance rogue events.

Procedure

---

1. **show wireless wps rogue stats**

   Example

   ```
   Device# show wireless wps rogue stats
              WSA Events
              Total WSA Events Triggered          : 9
              ROGUE_POTENTIAL_HONEYPOT_DETECTED   : 2
              ROGUE_POTENTIAL_HONEYPOT_CLEARED    : 3
              ROGUE_AP_IMPERSONATION_DETECTED     : 4
              Total WSA Events Enqueued           : 6
              ROGUE_POTENTIAL_HONEYPOT_DETECTED   : 1
              ROGUE_POTENTIAL_HONEYPOT_CLEARED    : 2
              ROGUE_AP_IMPERSONATION_DETECTED     : 3
   ```

   In this example, nine events occurred, but only six events were enqueued. Three events occurred before you enabled the WSA rogue feature.

2. **show wireless wps rogue stats internal**

   show wireless wps rogue ap detailed *rogue-ap-mac-addr*

   These commands display information about WSA events in the event history.

You can track WSA rogue event activity, investigate event history, and verify system responsiveness to rogue threats.

Rogue AP scale modes per class

Describes rogue AP scale modes, feature history, advantages and disadvantages of various role-scale modes, configuration procedures, and verification steps to optimize rogue AP management and scaling across different deployment classes.

A rogue AP scale mode is a database management mechanism that

- decides if the system adds or discards a newly detected rogue AP when the database reaches its capacity
- lets you set quotas or priorities by rogue AP classification to use available space efficiently, and
- allows you to customize storage management based on classification needs and resources.

Types of rogue AP scale modes

These modes determine whether the system adds a rogue AP to the database when it reaches maximum scale:

- **Quota**: Quotas apply to each classification as a percentage of the maximum scale. If a classification has a quota of X percent, that portion of the rogue AP database is reserved for the classification. If all memory for that classification is used, the system drops any new rogue APs in that classification.
- **Priority**: Priorities apply to different classifications. If you do not set quotas, the system uses priority mode by default. The system uses these priorities:
  - malicious: highest
  - custom: high
  - unclassified: medium
  - friendly: low

  The system enforces priorities only when the maximum database scale is reached. If you classify a new rogue AP and the database is full, the system adds it only if there are lower-priority rogue APs that are present. If this happens, the system deletes the newest rogue AP entry of the lowest priority. If there are no lower-priority rogue APs, the system drops the new AP.
- **Hybrid**: Hybrid mode combines quotas and priorities. When space is available, rogue APs of lower priority use any unused quota reserved for higher-priority rogue APs.

Rogue AP scale mode logic after reaching maximum scale

After the rogue AP database reaches maximum scale, the system applies this logic when it classifies a new rogue AP:

- If the number of stored rogue APs in the new rogue AP's class is below its quota, store the new rogue AP and delete the newest rogue AP of the lowest-priority classification that exceeds its quota.
- If a lower-priority classification exceeds its quota, the system deletes the newest rogue AP in that classification and stores the new rogue AP.
- If neither condition applies, the system drops the new rogue AP.

Note

- Configure quotas and priorities carefully to balance security requirements with database limitations.
- If you do not configure quotas, the system uses priority mode.

Suppose the database has reached maximum scale and a new rogue AP classified as malicious is detected. If the malicious quota allows storage, the system stores the new rogue AP and removes the newest rogue AP from the lowest-priority classification that exceeds its quota. If no such rogue AP exists, the system drops the new rogue AP.

Feature history for rogue full scale quotas and priorities

Feature history for rogue full scale quotas and priorities.

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 129: Feature History for Rogue Full Scale Quotas and Priorities

Release	Feature	Feature information
Cisco IOS XE 17.9.1	Rogue full scale quotas and priorities	The rogue full scale quotas and priorities feature helps you to improve the scalability, performance, manageability, and serviceability of rogue Access Points (APs).

Advantages and disadvantages of role-scale modes

This section lists information for choosing the appropriate role-scale mode, along with their main benefits and drawbacks.

Mode	Advantages	Disadvantages
Quota	Simple to use and understand.	• Memory is not used efficiently. • New rogue APs for a class that is already in its maximum quota are dropped. While the memory reserved for another class that does not have any rogue APs, stays empty. For example, this could lead to dropping malicious rogue APs, while there is still memory available.
Priority	• Simple to use and understand. • Utilizes the available memory. • Stores the important rogue APs.	Some of the lower-priority rogue AP classes might not be represented in the rogue database, if higher-priority rogue APs utilize all the available memory.

Mode	Advantages	Disadvantages
Hybrid	Utilizes the available memory, while providing quotas so that all the classes are represented in the database.	Difficult for users to understand the exact behavior.

Configure rogue AP scale (CLI)

Configure rogue AP scale using commands.

Set scale quotas, priorities, and hybrid modes for rogue AP types (malicious, custom, unclassified, friendly) using CLI commands.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure rogue scale quota for malicious, custom, unclassified, and friendly rogue APs.

   Example

   ```
   Device(config)# wireless wps rogue scale quota
   malicious percentage-malicious-rogue-AP custom percentage-custom-rogue-AP
   unclassified percentage-unclassified-rogue-AP friendly
   percentage-friendly-rogue-AP
   ```

   The default value for quota is 0. The sum of all the quotas must be less than or equal to 100 percent.

   If the sum of all the configured quotas is equal to 0, then priority mode is used. If the sum of all the quotas is not equal to 0, then quota mode is used. If hybrid mode is configured, hybrid mode is used no matter what the quota configuration is. Hybrid mode with all the quotas equal to 0, is identical to the priority mode.

   > **Note**
   >
   > Hybrid mode is enabled after the maximum scale is reached. All the rogue APs are stored before the maximum scale is reached.

3. Configure rogue scale priority for malicious, custom, unclassified, and friendly rogue APs.

   Example

   ```
   Device(config)# wireless wps rogue scale priority malicious highest custom
   high unclassified medium friendly low
   ```

Example

```
[no] wireless wps rogue scale priority malicious { high | highest | low | medium } custom { high | highest | low | medium } unclassified { high | highest | low | medium } friendly { high | highest | low | medium }
```

The default value for malicious is **highest** , the default value for custom is **high** , the default value for unclassified is **medium** , and the default value for friendly is **low** .

4. Configure rogue scale hybrid mode.

Example

```
Device(config)# wireless wps rogue scale mode hybrid
```

Unused quota reserved for higher-priority rogue APs are used by rogue APs of lower priority when space is available.

Rogue AP classification uses configured quotas, priorities, or hybrid mode to efficiently manage detected rogue APs according to defined organizational policies.

Verify rogue scale details

Command to verify rogue scale details.

To verify the rogue scale details, run this command:

```
Device# show wireless wps rogue stats
  Total Post Init/Max            : 0/4000
  Total/Max                      : 0/4200
  Init                           : 0
Classification
  Friendly                       : 0/0/0 (Total/Init/Quota[%])
  Malicious                      : 0/0/0 (Total/Init/Quota[%])
  Custom                         : 0/0/0 (Total/Init/Quota[%])
  Unclassified                   : 0/0/0 (Total/Init/Quota[%])
  Unknown                        : 0/0 (Total/Init)
Configured Quotas by Classification
  Custom         : <% of max scale>
  Friendly       : <% of max scale>
  Malicious      : <% of max scale>
  Unclassified   : <% of max scale>
Configured Priorities by Classification
  Custom          :  2 (High)
  Friendly        :  4 (Low)
  Malicious       :  1 (Highest)
  Unclassified    :  3 (Medium)

Configured Rogue Scale Mode: [Priority|Quota|Hybrid]
```

To view the rogue ad hoc summary, run this command:

```
Device# show wireless wps rogue adhoc summary
Detect and report Ad-Hoc Networks : Enabled
Auto-Contain Ad-Hoc Networks      : Disabled
Total Number of Rogue Ad-Hoc      : 0
Friendly Ad-Hoc Rogues            : 0
Malicious Ad-Hoc Rogues           : 0
Custom Ad-Hoc Rogues              : 0
```

```
Unclassified Ad-Hoc Rogues        : 0
Unknown Ad-Hoc Rogues             : 0
Client MAC Address   Adhoc BSSID        Classification  State        # APs
Last Heard
-------------------------------------------------------------------------------------------
```

Rogue PMF containment

Provides an overview of rogue Protected Management Frame (PMF) concepts, configuration tasks, and verification references for wireless security management.

Rogue PMF containment is a wireless security feature that

- uses 802.11w Protected Management Frames (PMF) to contain rogue APs and clients
- operates on centrally switched WLANs when the radio channel of the detecting AP matches the rogue AP's channel, and
- activates only when certain mode and channel conditions are met to secure the network against unauthorized devices.

Feature history

Table 130: Feature history table for rogue PMF containment

Feature name	Release information	Feature description
Rogue PMF containment	Cisco IOS XE 17.12.1	Starting with Cisco IOS XE Dublin 17.12.1, the controller contains a rogue AP with 802.11w Protected Management Frame (PMF) on centrally switched wireless LANs. Containment occurs if the client-serving radio channel of a rogue-detecting AP matches the channel of the corresponding rogue AP.

Operational scenarios

PMF containment occurs in these scenarios:

- You can use PMF containment only in the local mode.
- You can perform PMF containment only for rogue clients that have not joined a rogue AP.
- You can use PMF containment only if a rogue-detecting AP shares the same primary channel with a rogue client.
- You cannot use PMF containment on DFS channels, even if a DFS channel serves as the client-serving channel.
- PMF containment works only if at least one WLAN operates on the serving radio.

For information about APs that support the Rogue PMF Containment feature, see Cisco AP Feature Matrix.

Enable rogue PMF containment

Enable Rogue PMF Containment using commands.

Enable PMF containment to protect your wireless network from rogue APs.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile
   ```

3. Enable PMF-denial rogue AP containment.

   Example

   ```
   Device(config-ap-profile)# rogue detection containment pmf-denial
   ```

4. Enable PMF-denial type deauthentication rogue AP containment.

   Example

   ```
   Device(config-pmf-denial)# pmf-deauth
   ```

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-profile)# end
   ```

Rogue AP PMF containment is enabled for the specified AP profile.

```
Device# configure terminal
Device(config)# ap profile pmf-ap-profile
Device(config-ap-profile)# rogue detection containment pmf-denial
Device(config-pmf-denial)# pmf-deauth
Device(config-ap-profile)# end
```

Verify PMF containment

Verify PMF Containment using commands.

To verify PMF containment and the relevant statistics, use these commands.

To view the summary of containment details for all AP radios, use this command

```
Device# show wireless wps rogue containment summary

Rogue Containment activities for each managed AP

AP: 687d.b45f.2ae0  Slot: 1
```

```
  Active Containments   : 3
   Containment Mode     : DEAUTH_PMF
   Rogue AP MAC         : 687d.b45f.2a2d
   Containment Channels : 40
```

To verify the rogue statistics, use this command:

```
Device# show wireless wps rogue stats
.
.
.
 States
  Alert                          : 256
  Internal                       : 0
  External                       : 0
  Contained                      : 1
  Containment-pending            : 0
  Threat                         : 0
  Pending                        : 0
Rogue Clients
  Total/Max Scale                : 20/16000
  Contained                      : 0
  Containment-pending            : 0
.
.
.
```

150 Classifying Rogue Access Points

Topics:

- Rogue AP classifications

Explains how unauthorized wireless access points are detected, classified, and contained using controller rules or manual actions.

Rogue AP classifications

Outlines how the controller classifies, organizes, and manages rogue APs using configurable rules, describes classification types and rogue states, explains rule and manual transitions, and provides examples of classification mapping and allowable state changes.

A rogue AP classification is a wireless security mechanism that

- enables administrators to group rogue APs as Friendly, Malicious, Custom, , or Unclassified,
- uses configurable rules to automate the categorization and state assignment of unauthorized APs, and
- allows both automatic and manual reclassification of rogues based on network security requirements.

Additional reference information

- By default, no classification rules are active, so all unknown access points are assigned the Unclassified state. Administrators must enable classification rules to begin organizing rogue access points.
- When rules are enabled, all rogue access points in the Alert state are reclassified automatically based on the latest conditions and rule configurations.
- Changing or adding a rule triggers reclassification of all applicable rogue access points that are in the Alert state.
- You can manually move any rogue or ad hoc rogue access point to Unclassified and state, which represents the default state. Manually moved rogues are subject to rule-based reclassification.
- Rule-based classification does not apply to ad hoc rogues and rogue clients.
- You can configure up to 64 rogue classification rules per controller .
-

How the controller classifies rogue access points

When the controller software receives a rogue report from one of its managed access points, it follows this process:

- If the unknown access point is listed in the friendly MAC address list, the controller classifies it as Friendly.
- If it is not on the friendly MAC list, the controller applies the configured rogue classification rules.
-
- Rule-based classification is not applied to rogue access points that were manually classified.
- If a rogue matches the criteria in a classification rule, the controller assigns the classification type specified in that rule.
- If a rogue does not match any configured rules, it remains Unclassified.
- If a rogue access point is detected on the same wired network, the controller marks its state as Threat and classifies it as Malicious automatically (regardless of rules). You can manually contain such rogues, changing their state to Contained. If the rogue becomes unavailable, the controller moves it to the Alert state until you manually contain it.
- Administrators can always manually change the classification type and state of an access point as needed.
- Before classification, rogue access points are marked Pending.

Examples: Classification mapping

Rule-Based Classification Type	Rogue State
Custom	• Alert–No action is taken other than notifying the management station. The management station in the controller manages the controller and wired networks. • Contained–The unknown AP is contained. If none of the managed APs are available for containment, the rogue is in Contained Pending state.
Delete	Deletes the rogue AP.
Friendly	• Internal–If the unknown AP poses no threat to WLAN security, you can manually configure it as Friendly, Internal. An example of this would be the APs in your lab network. • External–If the unknown AP is outside the network and poses no threat to WLAN security, you can manually configure it as Friendly, External. An example of this would be the AP in your neighboring coffee shop. • Alert– No action is taken other than notifying the management station. The management station manages the controller and wired networks.
Malicious	• Alert– No action is taken other than notifying the management station. The management station manages the controller and wired networks. • Threat–The unknown AP is found to be on the network and poses a threat to WLAN security. • Contained–The unknown AP is contained. If none of the managed APs are available for containment, the rogue is in Contained Pending state.
Unclassified	• Alert– No action is taken other than notifying the management station. The management station manages the controller and wired networks. • Contained–The unknown AP is contained. If none of the managed APs are available for containment, the rogue is in contained pending state.

As mentioned earlier, the controller can automatically change the classification type and rogue state of an unknown AP based on user-defined rules. Alternatively, you can manually move the unknown AP to a different classification type and rogue state.

Allowable classification type and rogue state transitions

Assigning rogue APs classifications is like sorting emails automatically into spam or inbox based on filters (rules), while still allowing the user to manually move an email to a different folder when needed.

Rogue containment methods

Use this concept to understand the different methods for containing rogue access points, including how auto-containment applies to monitor mode APs and the priorities of various containment options.

A rogue containment method is a wireless security approach that

- detects unauthorized access points in the network environment,
- applies different containment strategies including manual, rule-based, and auto-containment, and
- prioritizes and restricts the actions of access points based on containment configuration and AP operating mode.

Containment methods and priorities

- **Manual rogue containment (Priority 1):** Administrators manually designate which rogue APs to contain, offering the highest control and flexibility. Containment levels can range from 1 to 4 APs participating.
- **Rule-based rogue containment (Priority 2):** The controller automatically contains rogue APs based on predefined rules, always using 1 AP for containment.
- **Auto-containment (Priority 3):** The controller automatically contains rogue APs, but only when specific options (such as "Using our SSID", "A valid client on the Rogue AP", or "Ad-hoc Rogue AP") are enabled.

Auto-containment for monitor mode APs

- When you enable 'Auto Containment only for Monitor Mode APs', only monitor mode APs perform auto-containment. Local or FlexConnect mode APs do not.
- This option applies only to auto-containment. It does not affect your manual or rule-based methods.
- Other AP modes (local or FlexConnect) still enforce rogue containment actions outside the scope of this auto-containment setting.

Containment prioritization and exceptions

- Manual containment always takes highest priority, followed by rule-based, with auto-containment last.
- If a rogue AP matches both manual or rule-based configurations, auto-containment does not occur for that AP.

Example

An ad-hoc rogue AP detected with a valid client connected triggers auto-containment by monitor mode APs only if the corresponding option is enabled. If manual containment is also configured for this rogue AP, manual containment takes precedence and disables auto-containment for that case.

Guidelines and restrictions for classifying Rogue APs

This section lists guidelines and restrictions for classifying rogue access points.

Classifying Custom type rogues is tied to rogue rules. Therefore, it is not possible to manually classify a rogue as Custom. Custom class change can occur only when rogue rules are used.

Some SNMP traps are sent for containment by rule and every 30 minutes for rogue classification change.

Rogue rules are applied on every incoming new rogue report in the controller in the order of their priority.

After a rogue satisfies a rule and is classified, it does not move down the priority list for the same report.

Previously classified rogue gets re-classified on every new rogue report with the following restrictions:

- Rogues which are classified as friendly by rule and whose state is set to ALERT, go through re-classification on receiving the new rogue report.
- If a rogue is classified as friendly by the administrator manually, then the state is INTERNAL and it does not get re-classified on successive rogue reports.
- If rogue is classified as malicious, irrespective of the state it does not get re-classified on subsequent rogue reports.

The rogue classification rules are re-evaluated at every report received by the managed access points. Hence, a rogue access point can move from one state to another, if a different rule matches the last report.

If a rogue AP is classified as friendly or ignored, all rogue clients associated with it are not tracked.

Until the controller discovers all the APs through neighbor reports from APs, the rogue APs are kept in unconfigured state for three minutes after they are detected. After 3 minutes, the rogue policy is applied on the rogue APs and the APs are moved to unclassified, friendly, malicious, or custom class. Rogue APs kept in unconfigured state means that no rogue policy has yet been applied on them.

When a rogue BSSID is submitted for a containment on Cisco Catalyst 9800 Series Wireless Controller, if the controller has enough resources, it will contain. The APs that detect the particular contained rogue AP starts broadcasting the DEAUTH packets.

Wireless client connected to the contained rogue BSSID will disconnect once DEAUTH packets are received. However, when the client assumes being in a connected state, repeatedly tries to reconnect and the wireless client's user browsing experience would be badly affected.

Also, in a high RF environment like that of a stadium, though DEAUTH packets are broadcasted, client does not receive all of them because of RF disturbance. In this scenario, the client may not be fully disconnected but will be affected badly.

The rouge AP manual classification limit has been enhanced from 625 to 10,000 configurations at a time. The rouge client manual classification limit has been enhanced from 625 to 10,000 configurations at a time.

How to classify rogue APs

-

Classify a rogue AP manually (GUI)

Use this procedure to classify a detected rogue access point (AP) as either authorized or unauthorized using the GUI.

Set the correct classification for rogue access points to maintain wireless network security and compliance.

Use this task after the system detects unclassified APs and you need to determine their status.

Procedure

1. Choose **Monitoring > Wireless** > **Rogues**.
2. In the **Unclassified** tab, select an AP to view the detail in the lower pane.
3. Use the **Class Type** drop-down to set the status.
4. Click **Apply**.

The AP is reclassified and removed from the unclassified list; its status is updated accordingly.

Classify rogue APs and clients manually (CLI)

Use this procedure to classify rogue access points (AP) manually using commands.

Classify rogue access points and clients to enhance network security.

This procedure is used in environments where rogue access points and clients need to be identified and managed to prevent unauthorized access to the network.

Before you begin

Ensure you have the necessary permissions to configure the device.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Detect and report the ad hoc rogue.

   Example

   ```
   Device(config)# wireless wps rogue adhoc alert mac-addr
   ```

   Enter one of these options after you enter the **adhoc** keyword:

   - **alert**–Sets the ad hoc rogue access point to alert mode. If you choose this option, enter the MAC address for the *mac-addr* parameter.
   - **auto-contain**–Sets the automatically containing ad hoc rogue to auto-contain mode.
   - **contain**–Sets the containing ad hoc rogue access point to contain mode. If you choose this option, enter the MAC address for the *mac-addr* parameter and containment level for the *containment-level* parameter. The valid range for *containment-level* is from 1 to 4.
   - **external**–Sets the ad hoc rogue access point as **external**. If you choose this option, enter the MAC address for the *mac-addr* parameter.
   - **internal**–Sets the ad hoc rogue access point as **internal**. If you choose this option, enter the MAC address for the *mac-addr* parameter.

3. Configure the rogue APs.

   Example

   ```
   Device(config)# wireless wps rogue ap malicious mac-addr state contain
   containment-level
   ```

   Example

   ```
   Device(config)# wireless wps rogue ap malicious 74a0.2f45.c520 state contain
    3
   ```

   Enter one of the following options after the **ap** keyword:

   - **friendly**–Configures the friendly rogue access points. If you choose this option, enter the MAC address for the *mac-addr* parameter. After that enter the **state** keyword followed by either of these options: **internal** or **external**.
   - **malicious**–Configures the malicious rogue access points. If you choose this option, enter the MAC address for the *mac-addr* parameter. After that enter the **state** keyword followed by either of these options: **alert** or **contain**.
   - **alert**–Sets the malicious rogue access point to **alert** mode.
   - **contain**–Sets the malicious rogue access point to **contain** mode. If you choose this option, enter the containment level for the *containment-level* parameter. The valid range is from 1 to 4.

4. Configure the rogue clients.

   Example

   ```
   Device(config)# wireless wps rogue client contain mac-addr containment-level
   ```

Example

```
Device(config)# wireless wps rogue client contain 74a0.2f45.c520 2
```

Enter the following option after you enter the **client** keyword:

contain–Contains the rogue client. After you choose this option, enter the MAC address for the *mac-addr* parameter and the containment level for *containment-level* parameter. The valid range for *containment-level* is from 1 to 4.

5. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure rogue classification rules (GUI)

Configure rogue classification rules using GUI.

Procedure

---

1. Choose **Configuration** > **Security** > **Wireless Protection Policies**.
2. In the **Wireless Protection Policies** page, choose **Rogue AP Rules** tab.
3. On the **Rogue AP Rules** page, click the name of the **Rule** or click **Add** to create a new one.
4. In the **Add/Edit Rogue AP Rule** window that is displayed, enter the name of the rule in the **Rule Name** field.
5. Choose the rule type from the following **Rule Type** drop-down list options:
   - Friendly
   - Malicious
   - Unclassified
   - Custom

Configure rogue classification rules (CLI)

Use this procedure to configure rogue classification rules using commands.

Establish rules for classifying rogue access points to enhance network security.

This configuration is used in environments where rogue access points need to be identified and classified to prevent unauthorized access.

Before you begin

Ensure you have the necessary permissions to configure rogue classification rules on the device.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Create or enable a rule and set its priority.

Example

```
Device(config)# wireless wps rogue rule rule-name priority priority
```

When creating a rule, you must enter the priority for the rule.

> **Note**
>
> After creating a rule, you can edit the rule and change the priority only for the rogue rules that are disabled. You cannot change the priority for the rogue rules that are enabled. While editing, changing the priority for a rogue rule is optional.

3. Classify a rule.

Example

```
Device(config)# wireless wps rogue rule rule_3 priority 3
```

```
Device(config-rule)# classify friendly state {alert | external | internal}
```

Specifies the classification that needs to be applied to the rogue access points matching this rule.

- **friendly**–Configures the friendly rogue access points. After that enter the **state** keyword followed by either of these options: **alert**, **internal**, or **external**. If you select an **internal** option, it indicates that you trust a foreign access point. If you select an **external** option, it indicates that you acknowledge the presence of a rogue access point.
- **malicious**–Configures the malicious rogue access points. After that enter the state keyword followed by either of these options: alert or contained.
- **alert**–Sets the malicious rogue access point to alert mode.
- **contained**–Sets the malicious rogue access point to contained mode.

4. Add conditions to a rule.

Example

```
Device(config)# wireless wps rogue rule rule_3 priority 3
```

```
Device(config-rule)# condition {client-count condition_value | duration
duration_value | encryption | infrastructure | rssi | ssid ssid_name |
wildcard-ssid}
```

Adds the following conditions to a rule, which the rogue access point must meet:

- **client-count**–Requires that a minimum number of clients be associated to the rogue access point. For example, if the number of clients associated to the rogue access point is greater than or equal to the configured value, the

access point could be classified as Malicious. If you choose this option, enter the minimum number of clients to be associated to the rogue access point for the *value* parameter. The valid range is from 1 to 10 (inclusive), and the default value is 0.

- **duration**—Requires that the rogue access point be detected for a minimum period of time. If you choose this option, enter a value for the minimum detection period for the *duration_value* parameter. The valid range is from 0 to 3600 seconds (inclusive), and the default value is 0 seconds.
- **encryption**—Requires that the advertised WLAN does not have encryption enabled. You can choose **any** for any type of encryption, **off** for no encryption, **wpa1** for WPA encryption, **wpa2** for WPA2 encryption, **wpa3-owe** for WPA3 OWE encryption, or **wpa3-sae** for WPA3 SAE encryption.
- **infrastructure**—Requires the SSID to be known to the controller.
- **rssi**—Requires the rogue access point to be detected with a minimum RSSI value. If the classification is Friendly, the condition requires the rogue access point to be detected with a maximum RSSI value. The valid range is from -95 to -50 dBm (inclusive).
- **ssid**—Requires the rogue access point to have a specific SSID. You could specify up to 25 different SSIDs. You should specify an SSID that is not managed by the controller. If you choose this option, enter the SSID for the *ssid_name* parameter. The SSID is added to the configured SSID list you just created.
- **wildcard-ssid**—Allows you to specify an expression that could match an SSID string. You can specify up to 25 of these SSIDs.

5. Specify matching conditions for the rule.

Example

```
Device(config)# wireless wps rogue rule rule_3 priority 3
```

```
Device(config-rule)# match {all | any}
```

Specifies whether a detected rogue access point must meet all or any of the conditions specified by the rule for the rule to be matched and the rogue access point to adopt the classification type of the rule.

6. Set a command to its default.

Example

```
Device(config-rule)# default
```

7. Exit the sub-mode.

Example

```
Device(config)# wireless wps rogue rule rule_3 priority 3
```

```
Device(config-rule)# exit
```

```
Device(config)#
```

8. Disable a particular rogue rule.

Example

```
Device(config)# wireless wps rogue rule rule_3 priority 3
```

```
Device(config-rule)# shutdown
```

Disables a particular rogue rule. In this example, the rule *rule_3* is disabled.

9. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

10. Enter global configuration mode.

Example

```
Device# configure terminal
```

11. Disable all the rogue rules.

Example

```
Device(config)# wireless wps rogue rule shutdown
```

12. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

Returns to privileged EXEC mode. Alternatively, you can also press Ctrl-Z to exit global configuration mode.

You have now configured the rogue classification rules, enabling effective management of rogue access points in the network..

Monitor rogue classification rules

This section provides CLI command examples for classifying rogue access points using different rules and conditions.

You can monitor the rogue classification rules using these commands.

Table 131: Commands for monitoring rogue classification rules

Command	Purpose
show wireless wps rogue rule detailed	Displays detailed information of a classification rule.
show wireless wps rogue rule summary	Displays a summary of the classification rules.

Examples: classifying rogue AP

This section provides CLI command examples for classifying rogue access points using different rules and conditions.

This example shows how to classify a rogue AP with MAC address 00:11:22:33:44:55 as malicious and mark it for being contained by 2 managed APs:

```
Device# configure terminal
Device(config)# wireless wps rogue ap malicious 0011.2233.4455 state contain
2
```

This example shows how to create a rule that can categorize a rogue AP that is using SSID **my-friendly-ssid**, and it is seen for at least for 1000 seconds as friendly internal:

```
Device# configure terminal
Device(config)# wireless wps rogue rule ap1 priority 1
Device(config-rule)# condition ssid my-friendly-ssid
Device(config-rule)# condition duration 1000
Device(config-rule)# match all
Device(config-rule)# classify friendly state internal
Device(config-rule)# no shutdown
```

This example shows how to apply a condition that a rogue access point must meet:

```
Device# configure terminal
Device(config)# wireless wps rogue rule ap1 priority 1
Device(config-rule)# condition client-count 5
Device(config-rule)# condition duration 1000
Device(config-rule)# no shutdown
Device(config-rule)# end
```

This example shows a condition to classify rogue devices with the controller SSIDs as malicious:

```
Device# configure terminal
Device(config)# wireless wps rogue rule ap1 priority 1
Device(config-rule)#  classify malicious state alert
Device(config-rule)#  condition duration 30
Device(config-rule)#  condition infrastructure ssid
Device(config-rule)#  match all
Device(config-rule)#  no shutdown
Device(config-rule)# end
```

151 Advanced WIPS

Topics:

- Advanced WIPS

This chapter explains how to enable, configure, and verify Cisco Advanced WIPS on supported controllers and APs, manage signatures and forensic capture, and view and analyze real-time and historical WIPS alarms via GUI and CLI.

Advanced WIPS

Outlines Advanced WIPS features, covering solution components, supported modes and platforms, signature definitions, guidelines and restrictions, enabling methods through GUI and CLI, syslog support and thresholds, alarm monitoring, and verification procedures for secure wireless environments.

A Cisco Advanced Wireless Intrusion Prevention System (aWIPS) is a wireless security system that

- uses advanced techniques for wireless threat detection and performance management
- enables the AP to detect threats and generate alarms, and
- combines network traffic analysis, topology information, signatures, and anomaly detection for comprehensive wireless threat prevention.

Feature history for advanced WIPS

This table provides release and related information for the features explained in this module. These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Table 132: Feature history for advanced WIPS

Release	Feature name	Feature information
Cisco IOS XE 17.5.1	Advanced WIPS Signatures	Up to 15 additional signatures are supported.
Cisco IOS XE 17.6.1	Syslog Support for Advanced WIPs	From 17.6.1 release onwards: • Two additional signatures are supported. • Syslog support has been added to the controller for advanced WIPS.

Advanced WIPS signatures and definitions

This section lists advanced WIPS (Wireless Intrusion Prevention System) signatures and their definitions, including alarms introduced in Cisco IOS XE Bengaluru 17.5.1 and later, as well as alarms introduced in prior releases.

The table lists alarms introduced starting with Cisco IOS XE Bengaluru Release 17.5.1.

Table 133: Advanced WIPS signatures and definitions (Cisco IOS XE Bengaluru 17.5.1 and later)

Advanced WIPS signature	Definition
RTS Virtual Carrier Sense Attack	This alarm extends the RTS flood alarm introduced in Cisco IOS XE Bengaluru 17.4.x. The alarm triggers when an RTS frame with a large duration value is detected. An attacker can use these frames to exhaust airtime and disrupt wireless client services.
CTS Virtual Carrier Sense Attack	This alarm extends the CTS flood alarm introduced in Cisco IOS XE Bengaluru 17.4.x. The alarm triggers when a CTS frame with a large duration value is detected. An attacker can use these frames to exhaust airtime and disrupt wireless client services.

Advanced WIPS signature	Definition
Deauthentication Flood by Pair	This alarm provides enhanced threat context by tracking both the source (attacker) and the destination (victim) involved in the attack.
Fuzzed Beacon	A fuzzed beacon occurs when an attacker introduces invalid, unexpected, or random data into a beacon frame and replays the modified frames over the air. This can cause unexpected behavior on the destination device, such as driver crashes, operating system crashes, and stack-based overflows, and may allow execution of arbitrary code.
Fuzzed Probe Request	A fuzzed probe request occurs when an attacker introduces invalid, unexpected, or random data into a probe request and replays the modified frames over the air.
Fuzzed Probe Response	A fuzzed probe response occurs when an attacker introduces invalid, unexpected, or random data into a probe response and replays the modified frames over the air.
PS Poll Flood by Signature	A PS poll flood occurs when an attacker spoofs the MAC address of a wireless client and sends a large number of PS poll frames. The access point sends buffered data frames to the client, which may cause the client to miss data frames while operating in power-save mode.
EAPOL Start Flood by Signature	An Extensible Authentication Protocol over LAN (EAPOL) start flood occurs when an attacker floods an access point with EAPOL start frames to exhaust its internal resources.
Reassociation Request Flood by Destination	A reassociation request flood occurs when a device floods an access point with a large number of spoofed client reassociation requests to exhaust its resources, particularly the client association table. When the table overflows, legitimate clients cannot associate, resulting in a denial-of-service (DoS) attack.
Beacon Flood by Signature	A beacon flood occurs when stations receive a large number of beacons generated with different MAC addresses and SSIDs. This flood prevents clients from detecting beacons sent by corporate access points and can result in a denial-of-service (DoS) attack.
Probe Response Flood by Destination	A probe response flood occurs when a device floods clients with a large number of spoofed probe responses. This prevents clients from detecting valid probe responses sent by corporate access points.
Block Acknowledgement Flood by Signature	A block acknowledgement flood occurs when an attacker sends an invalid Add Block Acknowledgement (ADDBA) frame to an access point while spoofing a valid client MAC address. The access point then ignores valid traffic from the client until traffic outside the invalid frame range is received.

Advanced WIPS signature	Definition
AirDrop Session	An AirDrop session uses Apple AirDrop to establish a peer-to-peer link for file sharing. This activity can introduce security risks by allowing unauthorized peer-to-peer networks to appear in the WLAN environment.
Malformed Association Request	A malformed association request occurs when an attacker sends a malformed request to an access point to exploit software defects, potentially resulting in a denial-of-service (DoS) attack.
Authentication Failure Flood by Signature	An authentication failure flood occurs when a device floods an access point with invalid authentication requests spoofed from a valid client, which can cause client disconnections.
Invalid MAC OUI by Signature	An invalid MAC OUI event occurs when a spoofed MAC address that does not contain a valid organizationally unique identifier (OUI) is used.
Malformed Authentication	Malformed authentication occurs when an attacker sends malformed authentication frames that may expose vulnerabilities in certain wireless drivers.

The table lists alarms introduced prior to Cisco IOS XE Bengaluru Release 17.5.1.

Table 134: Advanced WIPS signatures (prior to Cisco IOS XE Bengaluru 17.5.1)

Advanced WIPS signature
Authentication Flood Alarm
Association Flood Alarm
Broadcast Probe Flood Alarm
Disassociation Flood Alarm
Broadcast Disassociation Flood Alarm
Deauthentication Flood Alarm
Broadcast Deauthentication Flood Alarm
EAPOL Logoff Flood Alarm
CTS Flood Alarm
RTS Flood Alarm

Guidelines and Restrictions

Guidelines and restrictions for aWIPS profile and supported access points.

- In the aWIPS profile, Cisco Aironet 1850 Series Access Points, Cisco Catalyst 9117 Series Access Points, and Cisco Catalyst 9130AX Series Access Points can detect EAPOL logoff attack and raise alarms accordingly, only on off-channel. They can not detect EAPOL logoff attack and raise alarms on on-channel.

- From Cisco IOS XE 17.12.6 onwards, aWIPS profile download is supported when Cisco Catalyst Center is configured using the fully qualified domain name (FQDN). It strictly requires an IP name-server and a valid DNS record, as it bypasses local IP host entries.

Enable advanced WIPS

From Cisco IOS XE Release 17.5.1 onward, aWIPS security has higher priority than Hyperlocation/Fastlocate.

These are the possible scenarios.

All Catalyst APs that support Fastlocate can operate with aWIPS regardless of AP mode or configuration.

When both aWIPS and Hyperlocation are enabled on Cisco Aironet 4800 APs in any mode except Monitor mode, only aWIPS is available.

Table 135: Feature availability for Cisco Aironet 4800 AP (All Modes)

Hyperlocation/Fastlocate	Advanced WIPS	Cisco Aironet 4800 AP Mode	Cisco Aironet 4800 AP effective feature
Enable	Enable	Any Non-Monitor	aWIPS [2]
Enable	Disable	Any Non-Monitor	Hyperlocation/Fastlocate
Disable	Disable	Any Non-Monitor	Hyperlocation/Fastlocate and aWIPS are disabled.
Disable	Enable	Any Non-Monitor	aWIPS
Enable	Enable	Monitor	aWIPS and Hyperlocation [3]
Disable	Enable	Monitor	aWIPS [4]
Enable	Disable	Monitor	Hyperlocation/Fastlocate
Disable	Disable	Monitor	Hyperlocation/Fastlocate and aWIPS are disabled.

Syslog messages for advanced WIPS

This feature adds syslog support to the controller for Advanced WIPS.

A syslog message for advanced WIPS is a controller notification mechanism that

- logs alarms sent from APs
- applies throttling to prevent redundant messages during configured intervals, and
- omits client-specific information from logs to protect privacy.

Sample syslog message format

A typical syslog message for advanced WIPS includes key elements such as:

[2] In modes other than the Monitor mode, if both aWIPS and Hyperlocation/Fastlocate are enabled, only aWIPS is available.
[3] In Monitor mode, if both aWIPS and Hyperlocation/Fastlocate are enabled, both aWIPS and Hyperlocation/Fastlocate are available.
[4] To monitor the status of aWIPS and Hyperlocation/Fastlocate simultaneously on AP, use the **show capwap client rcb** command.

- AP name
- AP Ethernet MAC address
- AP Radio MAC address
- Alarm description (signature ID)

```
Nov 18 20:45:23.746: %APMGR_AWIPS_SYSLOG-6-APMGR_AWIPS_MESSAGE: Chassis 1 R0/0:
 wncd: AWIPS alarm:(AP00B0.E19A.5720) 00b0.e19a.5720 Radio MAC 00b0.e19b.c300
 detected Probe Response Flood by Destination (10019)
```

Additional notes

- When an AP sends an alarm to the controller, the controller raises the corresponding syslog message.
- Throttling ensures that if the same signature is detected from the same AP within a set interval (e.g., 1 minute), only one syslog message is generated for that period, regardless of the number of occurrences.
  - For example, 100 occurrences of the same signature from an AP within a minute will result in only one syslog message for that interval.
- Syslog messages generated for advanced WIPS do not display any client device information or context.

Advanced WIPS solution components and capabilities

Advanced WIPS solution components and capabilities.

The aWIPS solution comprises these components and capabilities.

Solution components

The aWIPS solution comprises these components

- Cisco Catalyst 9800 Series Wireless Controller
- Cisco Aironet Wave 2 APs
- Cisco Catalyst Center

Because aWIPS functionality is integrated into Cisco Catalyst Center, aWIPS can configure WIPS policies, monitor alarms, and report threats.

Capabilities

aWIPS supports these capabilities

- Static signatures

  Beginning with Cisco IOS XE version 17.4.1, Cisco Catalyst Center can change threshold values and push new signature files to the AP.
- Enable or disable signature forensic capture from Cisco Catalyst Center.
- Standalone signature detection only
- Only alarms are supported.
- GUI support
- CLI to view alarms

- The static signature file is packaged with the controller and AP image.
- Export alarms to Cisco Catalyst Center through WSA channel

> **Note**
>
> aWIPS alarm details such as the AP MAC address, alarm ID, alarm string, and signature ID are displayed on the Cisco Catalyst 9800 series wireless controller GUI.

Supported modes and platforms

Controllers supporting advanced WIPS.

aWIPS is supported on these controllers:

- Cisco Catalyst 9800 Series Wireless Controllers
- Cisco Embedded Wireless Controller on Catalyst Access Points

> **Note**
>
> aWIPS is not supported on Cisco IOS APs.

Enable advanced WIPS (GUI)

Enable advanced WIPs using GUI.

Enable advanced WIPS features on your access points for improved network security.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. Click **Add**. The **Add AP Join Profile** window is displayed.
3. In the **Add AP Join Profile** window, click the **Security** tab.
4. Under the **aWIPS** section, check the **aWIPS Enable** check box.
5. Click **Apply to Device**. You return to the **General** tab.
6. Click the **Security** tab.
7. Under the **aWIPS** section, check the **Forensic Enable** check box.
8. Click **Apply to Device**.

Advanced WIPS and forensic features are enabled for the selected AP profile.

Enable advanced WIPS (CLI)

Enable advanced WIPS.

Enable advanced WIPS to enhance network security and ensure proper priority over location services.

To enable aWIPS from the controller and ensure that aWIPS has higher priority than Hyperlocation/Fastlocate , perform these steps:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the default AP profile.

   Example

   ```
   Device(config)# ap profile ap-profile-name
   ```

3. Enable aWIPS.

   Example

   ```
   Device(config-ap-profile)# awips
   ```

   Note

   aWIPS is disabled by default on the controller.

4. Enable forensics for aWIPS alarms.

   Example

   ```
   Device(conf-ap-profile)# awips forensic
   ```

5. Enable Hyperlocation/Fastlocate on all the supported APs that are associated with this AP profile.

   Example

   ```
   Device(config-ap-profile)# hyperlocation
   ```

6. Return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-profile)# end
   ```

The controller is now configured with advanced WIPS and forensics, and prioritizes aWIPS above Hyperlocation/Fastlocate features on AP.

Configure syslog threshold for advanced WIPS (CLI)

Configure syslog threshold for advanced WIPS using command.

Set the syslog throttle interval for advanced WIPS, so you can manage syslog message frequency and reduce unnecessary log volume.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the syslog threshold for aWIPS.

   Example

   ```
   Device(config)# awips-syslog throttle period syslog-throttle-interval
   ```

   syslog-throttle-interval : Enter the syslog throttle interval, in seconds. The range is from 30 to 600.

   > **Note**
   >
   > The default throttling interval is 60 seconds.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The syslog throttle interval for advanced WIPS is configured. Syslog messages for WIPS events will now be generated according to the specified interval.

View advanced WIPS alarms (GUI)

View advanced WIPS alarms using GUI.

Identify and analyze current and historical WIPS alarms.

Procedure

1. Navigate to **Monitoring** > **Security** > **Advanced Wireless Intrusion Prevention System (aWIPS)** .
2. To view the details of the alarms in the last five minutes (0.08 hours), click the **Current Alarms** tab.
3. To view the alarm count for longer periods such as hourly or daily (24 hours), click the **Historical Statistics** tab.
4. Sort or filter the alarms based on these parameters:

- **AP Radio MAC address**
- **Alarm ID**
- **Time Stamp**
- **Signature ID**
- **Alarm Description**
- **Alarm Message Index**

Advanced WIPS alarms appear according to the selected criteria. You receive real-time and historical data for security monitoring and analysis.

Verify advanced WIPS

Command to verify advanced WIPS.

To view the aWIPS status, use the **show awips status** *radio_mac* command:

```
Device# show awips status 0xx7.8xx8.2xx0

AP Radio MAC  AWIPS Status  Forensic Capture Status   Alarm Message Count
------------------------------------------------------------------------------
0xx7.8xx8.2xx0      ENABLED       CONFIG_NOT_ENABLED  14691
```

The various aWIPS status indicators are:

- `ENABLED`: aWIPS enabled.
- `NOT_SUPPORTED`: The AP does not support AWIPS.
- `CONFIG_NOT ENABLED`: aWIPS is not enabled on the AP.

To view details of specific alarm signatures, use the **show awips alarm signature** *signature_id* command:

```
Device# show awips alarm signature 10001

AP Radio MAC    AlarmID   Timestamp          SignatureID    Alarm Description
     Message Index
_____________________________________________________________________________
0xx7.8xx8.2f80   1714    11/02/2020 13:02:19   10001      Authentication Flood
     3966
```

To view alarm message statistics, use the **show awips alarm statistics** command:

```
Device# show awips alarm statistics
```

To view a list of alarms since the last clear, use the **show awips alarm ap** *ap_mac* **detailed** command:

```
Device# show awips alarm ap 0xx7.8xx8.2f80 detailed
AP Radio MAC    AlarmID     Timestamp              SignatureID   Alarm
Description
_____________________________________________________________________________
0xx7.8xx8.2f80    2491    08/02/2022 17:44:40       10009         RTS Flood
```

To view detailed alarm information, use the **show awips alarm detailed** command:

```
Device# show awips alarm detailed

AP Radio MAC    AlarmID  Timestamp   SignatureID Alarm Description
-------------------------------------------------------------------------------------
7xx3.5xxd.d360  1        10/29/2020  23:21:27    10001 Authentication Flood
by Source
dxxc.3xx5.9460  71       10/29/2020  23:21:27    10001 Authentication Flood
by Source
7xx3.5xxd.d360  2        10/29/2020  23:21:28    10002 Association Request
Flood by Destination
dxxc.3xx5.9460  72       10/29/2020  23:21:28    10002 Association Request
Flood by Destination
```

To view the alarms on a specific AP, use the **show awips alarm ap** *radio_mac* **detailed** command:

Verify syslog configuration for advanced WIPS

To verify the syslog configuration for aWIPS, use this command:

```
Device# show awips syslog throttle

Syslog Throttle Interval (seconds)

-------------------------------------------------------------------------------------

38
```

152 Cisco TrustSec

Topics:

- Cisco TrustSec
- Security Group Access Control List (SGACL)
- Cisco TrustSec - ISE communication over IPv6
- TrustSec IP-SGT High Availability binding

Cisco TrustSec

Outlines Cisco TrustSec architecture, including user and device identity, scalable access controls, data confidentiality, and enforcement using Cisco ISE and SGACLs. Highlights SGACL and SGT inline tagging features, supported platforms, HA policy for FlexConnect, and commands for managing environment data.

A Cisco TrustSec solution is a network security architecture that

- strongly identifies users, hosts, and network devices within the network
- provides topology-independent and scalable access controls by classifying data traffic based on user or device roles, and
- ensures data confidentiality and integrity by establishing trust among authenticated peers and encrypting network links.

The key component of Cisco TrustSec is the Cisco Identity Services Engine (ISE). Cisco ISE can provision switches with TrustSec Identities and Security Group ACLs (SGACLs). SGACLs may be configured manually on the switch.

Note

Before changing CTS server to a new one, manually clear the CTS environment data using the **clear cts environment-data** command. Running the **show cts environment-data** command will return updated data.

Feature History

Feature Name	Release	Description
TrustSec IP-to-SGT High Availability binding	Cisco IOS XE 26.2.1	The feature synchronizes supported IP-to-SGT bindings between the active and standby devices, preserves binding information during a Stateful Switchover (SSO), and helps maintain consistent Security Group Access Control List (SGACL) enforcement after the switchover.
Trustsec policy HA support for FlexConnect mode APs	Cisco IOS XE 17.18.2	The feature ensures that CTS SGACL enforcement remains available and consistent during HA events such as Stateful Switchover (SSO) between wireless controllers. This provides uninterrupted security policy enforcement on Flex mode APs even during controller failover or redundancy events.
TrustSec support for Cisco Catalyst IW6300 Heavy Duty Series and 6300 Series Embedded Services APs	Cisco IOS XE 17.8.1	Enable and configure Cisco TrustSec Security Group ACL (SGACL) in FlexConnect and Flex+Bridge mode. SGACL enforcement on the controller is available for local and Bridge mode. Inline tagging and SXP are supported only in FlexConnect.

Feature Name	Release	Description
Support for SGT Inline Tagging Over Port-Channel Uplink	Cisco IOS XE 17.3.5a	SGT inline tagging over port-channel uplink is supported for Cisco Catalyst 9800-L, 9800-40, and 9800-80 Wireless Controllers. If you downgrade to releases that do not support SGT inline tagging over port-channel, the port-channel may be suspended.

Cisco TrustSec features

This section explains the Cisco TrustSec features.

This table lists the TrustSec features that will be implemented on TrustSec-enabled Cisco switches. Future releases will support more switches and provide additional TrustSec features.

Cisco TrustSec Feature	Description
802.1AE Tagging (MACsec)	Protocol for IEEE 802.1AE-based wire-rate hop-to-hop Layer 2 encryption. On MACsec-capable devices, your network traffic is encrypted when leaving one device, decrypted when entering the next device, while staying unencrypted inside the devices. This feature is only available between TrustSec hardware-capable devices.
Endpoint Admission Control (EAC)	EAC authenticates each device or user connecting to the TrustSec domain. This usually happens at the access level switch. If authentication and authorization are successful, your device or user session receives a Security Group Tag. You can use 802.1X, MAC Authentication Bypass (MAB), or Web Authentication Proxy (WebAuth) with EAC.
Network Device Admission Control (NDAC)	With NDAC, each network device in your TrustSec domain verifies the credentials and trustworthiness of peer devices. NDAC uses an authentication framework based on IEEE 802.1X port-based authentication and EAP-FAST as its EAP method. Successful authentication and authorization in NDAC process results in Security Association Protocol negotiation for IEEE 802.1AE encryption.
Security Group Access Control List (SGACL)	Security Group Access Control List (SGACL): An SGACL links a Security Group Tag with a policy that your devices enforce on SGT-tagged traffic leaving the TrustSec domain.
Security Association Protocol (SAP)	After NDAC authentication, the Security Association Protocol (SAP) automatically negotiates keys and the cipher suite for subsequent MACSec link encryption between TrustSec peers. SAP is defined in IEEE 802.11i.

Cisco TrustSec Feature	Description
Security Group Tag (SGT)	An SGT is a 16-bit label indicating the security classification of a source in the TrustSec domain. It is appended to an Ethernet frame or an IP packet.
SGT Exchange Protocol (SXP)	Security Group Tag Exchange Protocol (SXP). With SXP, devices that are not TrustSec-hardware-capable can receive SGT attributes for authenticated users and devices from the Cisco Identity Services Engine (ISE) or the Cisco Secure Access Control System (ACS). The devices can then forward a sourceIP-to-SGT binding to a TrustSec-hardware-capable device will tag the source traffic for SGACL enforcement.

When both ends of a link support 802.1AE MACsec, SAP negotiation occurs. An EAPOL-key exchange occurs between the supplicant and the authenticator to negotiate a cipher suite, exchange security parameters, and manage keys. Successful completion of these tasks results in the establishment of a security association (SA).

Depending on your software version, licensing, and link hardware support, SAP negotiation can use one of these modes of operation:

- Galois Counter Mode (GCM): authentication and encryption
- GCM authentication (GMAC): GCM authentication, no encryption
- No Encapsulation: no encapsulation (clear text)
- Null: encapsulation, no authentication or encryption

Configure a device SGT manually (CLI)

Use this procedure to manually configure a device STG using commands.

Configure a device Security Group Tag (SGT) using commands when the authentication server is unavailable.

When the authentication server is accessible, it assigns an SGT to your device for packets originating from your device. If the authentication server is not available, configure an SGT manually. The server-assigned SGT is always used instead of your manual configuration if both are present.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy profile-policy
   ```

3. Specify the Security Group Tag (SGT) number.

Example

```
Device(config-wireless-policy)# cts sgt sgt-value
```

The valid values are from 0 to 65,535.

4. Return to the global configuration mode.

Example

```
Device(config-wireless-policy)# exit
```

Configure ISE for TrustSec (CLI)

Use this procedure to configure ISE for TrustSec.

Set up ISE to enable secure network access using TrustSec policies using commands.

This task involves configuring a device to communicate with an ISE server as a RADIUS server and integrating TrustSec authorization.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Specify the RADIUS server name.

Example

```
Device(config)# radius server server-name
```

3. Specify the primary RADIUS server parameters.

Example

```
Device(config-radius-server)# address ipv6 ip-address
```

4. Specify the authentication and encryption key used between the Device and the key string RADIUS daemon running on the RADIUS server.

Example

```
Device(config-radius-server)# pac key key
```

5. Return to the configuration mode.

Example

```
Device(config-radius-server)# exit
```

6. Create a radius server-group identification.

Example

```
Device(config)# aaa group server radius server-group
```

> **Note**
>
> *server-group* refers to the server group name. The valid range is from 1 to 32 alphanumeric characters.

7. Create a CTS authorization list.

Example

```
Device(config)# cts authorization list mlist-name
```

8. Create an authorization method list for web-based authorization.

Example

```
Device(config)# aaa authorization network mlist-name group group-name
```

> **Note**
>
> Ensure that the ISE IP address configured on your controller is the same as the IP address configured on ISE (**Work Center** > **TrustSec** > **Components** > **Trustsec AAA Servers**).

> **Note**
>
> If the ISE version is 002.005(000.239), 002.004(000.357), 002.003(000.298), 002.002(000.470), 002.001(000.474), 002.000(001.130), or 002.000(000.306), use the **access-session tls-version 1.0** command to download PAC from ISE. For other ISE versions, the above command is not required.

Verify Cisco TrustSec configuration

This section explains the commands to verify Cisco TrustSec configuration.

To display the wireless CTS SGACL configuration summary, use this command:

```
Device# show wireless cts summary

Local Mode CTS Configuration

Policy Profile Name               SGACL Enforcement     Inline-Tagging
Default-Sgt
-------------------------------------------------------------------------------------
xyz-policy                        DISABLED              ENABLED          0

wireless-policy1                  DISABLED              DISABLED         0

w-policy-profile1                 DISABLED              DISABLED         0

default-policy-profile            DISABLED              DISABLED         0



Flex Mode CTS Configuration

Flex Profile Name                 SGACL Enforcement     Inline-Tagging
-----------------------------------------------------------------------
xyz-flex                          DISABLED              ENABLED
demo-flex                         DISABLED              DISABLED
flex-demo                         DISABLED              DISABLED
xyz-flex-profile                  DISABLED              DISABLED
default-flex-profile              DISABLED              DISABLED
```

To display CTS-specific configuration status for various wireless profiles, use this command:

```
Device# show cts wireless profile policy xyz-policy

Policy Profile Name           : xyz-policy
CTS
  Role-based enforcement         : ENABLED
  Inline-tagging                 : ENABLED
   Default SGT   : 100

Policy Profile Name            : foo2
CTS
  Role-based enforcement         : DISABLED
  Inline-tagging                 : ENABLED
  Default SGT           : NOT-DEFINED

Policy Profile Name                    : foo3
CTS
  Role-based enforcement         : DISABLED
  Inline-tagging                 : DISABLED
  Default SGT    : 65001
```

To display CTS configuration for a given wireless profile, use this command:

```
Device# show wireless profile policy detailed xyz-policy

Policy Profile Name           : xyz-policy
Description                   :
Status                        : DISABLED
VLAN                          : 1
Client count                  : 0
```

```
Passive Client                  : DISABLED
ET-Analytics                    : DISABLED
StaticIP Mobility               : DISABLED
!
.
.
.WGB Policy Params
  Broadcast Tagging             : DISABLED
  Client VLAN                   : DISABLED
Mobility Anchor List
  IP Address                                  Priority
CTS
  Role-based enforcement          : ENABLED
  Inline-tagging                  : ENABLED
  Default SGT                    : NOT-DEFINED
```

Security Group Access Control List (SGACL)

Explains how security group access control lists (SGACLs) enforce role-based, topology-independent access control in Cisco TrustSec deployments.

A security group access control list (SGACL) is a network security policy mechanism that

- enforces access control between groups of users, devices, and resources based on their assigned security group
- simplifies policy management by using unique security group tags (SGTs) to dynamically control permissions, and
- reduces administrative overhead by decoupling access policies from the network topology or addressing.

Security groups are defined by the administrator in Cisco Identity Services Engine (ISE). As new users and devices are added to the Cisco TrustSec domain, the authentication server assigns these new entities to the appropriate security groups. Cisco TrustSec assigns each security group a unique 16-bit number that is globally scoped within a Cisco TrustSec domain. The number of security groups on a wireless device is limited by the number of authenticated network entities. You do not need to configure the security group numbers manually.

After a device is authenticated, Cisco TrustSec tags each packet originating from that device with an SGT containing the device's security group number. The packet carries this SGT everywhere in the network, in the Cisco TrustSec header.

As the SGT contains the security group of the source, the tag can be referred to as the source SGT (S-SGT). The destination device is also assigned to a security group (destination SG) that can be referred to as the destination SGT (D-SGT), even though the Cisco TrustSec packet does not contain the security group number of the destination device.

You can control the operations that users can perform based on the security group assignments of users and destination resources, using the Security Group Access Control Lists (SGACLs). Policy enforcement in a Cisco TrustSec domain is represented by a permission matrix, with the source security group numbers on one axis and the destination security group numbers on the other axis. Each cell in the matrix body contains an ordered list of SGACLs, which specify the permissions that must be applied to packets originating from the source security group and destined for the destination security group. When a wireless client is authenticated, the controller downloads all the SGACLs in the matrix cells.

When a wireless client connects to the network, the client pushes all the ACLs to the controller .

Cisco TrustSec achieves role-based topology-independent access control in a network by assigning users and devices in the network to security groups and applying access control between the security groups. The SGACLs define access control policies based on the device identities. The security policy remains unchanged if the roles and permissions stay the same, even when the network topology changes. Assigning a user to an appropriate security group immediately grants the user permissions for that group when the user is added to the wireless group.

Role-based permissions reduce ACL size and simplify their maintenance. With Cisco TrustSec, the number of Access Control Entities (ACEs) that are configured is determined by the number of permissions specified, resulting in a much smaller number of ACEs.

For a list of Cisco APs that support SGACL, see the release notes.: https://www.cisco.com/c/en/us/support/wireless/catalyst-9800-series-wireless-controllers/products-release-notes-list.html

> **Note**
>
> Clients receive zero SGT value and DHCP clients receive an Automatic Private IP Addressing (APIPA) address when TrustSec policy “unknown to unknown” is denied in TrustSec matrix.
>
> Clients receive correct SGT values and DHCP clients receive an IP address when TrustSec policy “unknown to unknown” is permitted in TrustSec matrix.

Supported SGACL

The scenarios supported for SGACLs on the Cisco Catalyst 9800 Series Wireless Controller are:

- Wireless-to-wireless (within Enterprise network):
  - In FlexConnect mode with local switching, the egress AP enforces SGACL when a packet moves from a source wireless network to a destination wireless network.
  - In FlexConnect mode with central switching, SGACL enforcement is performed on the egress AP. To achieve this, the controller exports IP address to security group tag (IP-SGT) binding over SGT Exchange Protocol (SXP).
- Wired-to-wireless (DC-to-Enterprise network): Enforcement occurs when a packet reaches the destination AP.
- Wireless-to-wired (Enterprise network-to-DC): Enforcement occurs on the uplink switch when a packet reaches the ingress of the wired network.

Guidelines and restrictions of SGACL

This section explains the guidelines and restrictions of SGACL.

These are the guidelines and restrictions of Security Group Access Control List (SGACL).

- SGACL enforcement is carried out on the controller for local mode.
- CTS SGACL high availability is available on Flex mode. This provides uninterrupted security policy enforcement on Flex mode APs even during controller failover or redundancy events.
- SGACL enforcement is carried out on an AP for flex-mode APs performing local switching.
- SGACL enforcement for wireless clients is carried out either on the upstream switch or on the border gateway in a Branch-to-DC scenario.
- SGACL enforcement is not supported for non-IP or IP broadcast or multicast traffic.
- Per-WLAN SGT assignment is not supported. Static SGACL configuration on an AP is not supported.
- SGACL enforcement is not carried out for control-plane traffic between an AP and the wireless controller (for upstream or from upstream traffic).
- Non-static SGACL configurations are supported only for dynamic SGACL policies received from ISE.
- In case of Allow List model, you need to explicitly allow DHCP protocol for the client devices to get the DHCP IP address and then request the controller for SGACL policies.

Enable SGACL on the AP (GUI)

Use this procedure to enable SGACL on the APs using GUI.

Use the GUI to apply SGACL enforcement to APs as part of your security configuration workflow.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Flex**.
2. Click **Add**.
3. In the **General** tab, check the **Inline Tagging** and **SGACL Enforcement** check boxes and select the CTS profile name from the **CTS Profile Name** drop-down list.
4. Click **Apply to Device**.

Enable SGACL on the AP (CLI)

Use this procedure to enable SGACL on the APs using commands.

Enable role-based access control on APs for enhanced network security using commands.

> **Note**
>
> Use the **no** form of the commands given below to disable the configuration. For example, **cts role-based enforcement** disables role-based access control enforcement for APs.

Before you begin

- Security Group Access Control List (SGACL) on an AP is enabled only when the wireless controller is in FlexConnect mode.
- Configure the **cts manual** command on the uplink port to send or receive a tagged packet.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an RF profile and enter the RF profile configuration mode.

   Example

   ```
   Device(config)# wireless profile flex flex-profile
   ```

3. Enable role-based access control enforcement for the AP.

Example

```
Device(config-wireless-flex-profile)# cts role-based enforcement
```

4. Enable inline tagging on the AP.

   Example

   ```
   Device(config-wireless-flex-profile)# cts inline-tagging
   ```

5. Enable the CTS profile name and return to the global configuration mode.

   Example

   ```
   Device(config-wireless-flex-profile)# cts profile profile-name
   ```

   ```
   Device(config-wireless-flex-profile)# exit
   ```

6. Configure a site tag and enter the site tag configuration mode.

   Example

   ```
   Device(config)# wireless tag site site-name
   ```

7. Configure a FlexConnect profile and return to the global configuration mode.

   Example

   ```
   Device(config-site-tag)# flex-profile flex-profile-name
   ```

   ```
   Device(config-site-tag)# exit
   ```

8. Configure an AP and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap mac-address
   ```

9. Map a site tag to an AP.

   Example

   ```
   Device(config-ap-tag)# site-tag site-tag-name
   ```

What to do next
show cts ap sgt-info *ap-name* command to verify the SGACL configuration on the AP.

Policy enforcement

Policy enforcement for Cisco TrustSec access control.

A policy enforcement mechanism is a network security control that

- applies access control policies to network traffic using predefined rules
- uses attributes such as Security Group Tags (SGTs) to dynamically determine permitted actions, and
- enables context-based enforcement at key points within the network infrastructure.

Cisco TrustSec access control is implemented using ingress tagging and egress enforcement. At the ingress point to the Cisco TrustSec domain, the traffic from the source is tagged with an SGT containing the security group number of the source entity. The SGT is propagated across the domain with the traffic. At the egress point of the Cisco TrustSec domain, an egress device uses the security group of the destination entity (D-SGT) to determine the access policy to apply from the SGACL policy matrix.

> **Note**
>
> A Cisco AP must be in Listener mode or Both (Listener and Speaker) mode to enforce traffic because Listener mode maintains the complete set of IP-SGT bindings. After you enable enforcement on an AP, the corresponding policies are downloaded and pushed to the AP.

SGACL policies for guest wireless clients

This topic explains the SGACL policies for guest wireless clients.

A SGACL policy is an access control mechanism that

- defines the network permissions for wireless guest clients
- is enforced on the anchor controller in guest mobility scenarios, and
- uses integration with Cisco ISE for policy download and enforcement.

When a client joins the wireless network (WLAN), its session is managed by the Cisco Catalyst 9800 Series Wireless LAN Controller that the AP is connected to is the foreign controller. Auto-Anchor Mobility allows a specific WLAN (for example, Guest WLAN) to be anchored to a particular controller, regardless of the client's entry point into the network. Auto-Anchor Mobility is a wireless guest service in which all guest traffic tunnels back to the DMZ controller, regardless of the client's association point on the network.

In case of Auto-Anchor mobility, these apply to Cisco TrustSec support:

- Classification: Occurs during authentication and hence on Foreign for Layer 2 security WLANs and on Anchor for Layer 3 security cases.
- Propagation: Always occurs at the Anchor controller where the client traffic enters the wired network.
- Enforcement: SGACL download and enforcement occur on the Anchor controller; the Anchor controller must have connectivity to Cisco Identity Services Engine (ISE) and be registered as a Network Access Server (NAS). Enforcement is not supported on the Foreign controller even if the enforcement CLI is configured there.

This feature is supported in local mode and in FlexConnect Central Switching of the controller. FlexConnect mode with local switching and Fabric mode are not supported in guest scenarios because traffic does not pass through the controller.

Roaming of a guest client occurs only at the Guest Foreign controller, and the Guest Anchor remains fixed. The two types of supported roaming are inter-controller roaming and intra-controller roaming. Roaming during WebAuth pending is a special case that is also supported for Central Web Authentication (CWA) and Local Web Authentication (LWA).

Enable SGACL policy enforcement globally (CLI)

Enable SGACL policy enforcement globally using commands.

Enforce Cisco TrustSec SGACL policies on all routed interfaces to provide unified access control for both IPv4 and IPv6 traffic using commands.

You must enable SGACL policy enforcement globally on Cisco Catalyst 9800 Series Wireless Controller. The same configuration commands that are used for enforcement of IPv4 traffic apply for IPv6 traffic as well.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable Cisco TrustSec SGACL policy enforcement on routed interfaces.

   Example

   ```
   Device(config)# cts role-based enforcement
   ```

Enable SGACL policy enforcement per interface (CLI)

Use this procedure enable SGACL policy enforcement per interface using commands.

Enforce access control policies on selected network interfaces for enhanced security using SGACLs using commands.

After enabling the SGACL policy enforcement globally, enable Cisco TrustSec-on the uplink interfaces.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify interface on which to enable or disable SGACL enforcement.

   Example

   ```
   Device(config)# interface gigabitethernet interface number
   ```

3. Enable Cisco TrustSec SGACL policy enforcement on routed interfaces.

   Example

   ```
   Device(config-if)# cts role-based enforcement
   ```

4. Verify that SGACL enforcement is enabled.

Example

```
Device(config-if)# do show cts interface
```

Inline tagging

This section explains Inline tagging as a transport mechanism using which a controller or AP understands the source SGT.

An inline tag is a transport mechanism that

- carries the source SGT in packet headers as wireless traffic moves through the network
- enables enforcement of Security Group Access Control Lists (SGACLs) by controllers, and
- supports both centrally and locally switched wireless deployments.

There are two types of transport mechanisms: central switching and local switching.

- For central switching, the controller performs inline tagging of all packets sourced from wireless clients associated with the controller by tagging these packets with the Cisco Meta Data (CMD) tag. For packets inbound from the distribution system, inline tagging also requires the controller to strip the CMD header from the packet to identify the S-SGT tag. Afterward, the controller forwards the packet, which includes the S-SGT, for SGACL enforcement.
- To transmit locally switched traffic, an AP performs inline tagging for packets that are associated with the AP and sourced from clients. To receive traffic, the AP handles both locally and centrally switched packets, uses the S-SGT tag, and applies the SGACL policy.

When wireless Cisco TrustSec is enabled on the controller , enabling and configuring SXP to exchange tags with the switches is optional. Both wireless Cisco TrustSec and SXP modes are supported; however, there is no use case for enabling both wireless Cisco TrustSec (on an AP) and SXP concurrently.

Consideration and restriction for inline tagging over port-channel

- Configure the **cts manual** command on port-channel and its member interfaces to send or receive a tagged packet.
- If you downgrade to Cisco IOS XE releases that do not support inline tagging over port-channel, the port-channel may be suspended.

> **Note**
>
> Inline tagging over port-channel is supported in Cisco IOS XE 17.3.5 17.6.3 17.8.1 releases.

Configure SGACL, inline tagging, and SGT in local mode (GUI)

Use this procedure to configure SGACL, inline tagging, and SGT in local mode using GUI.

Configure SGACL enforcement, inline tagging, and default SGT values in local mode through the GUI.

Enable Security Group Access Control Lists (SGACL), inline tagging, and assign default Security Group Tags (SGT) on a policy profile operating in local mode.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click the Policy Profile Name. The **Edit Policy Profile** is displayed.
3. Choose **General** tab.
4. In the **CTS Policy** settings, check or uncheck the **Inline Tagging** and **SGACL Enforcement** check boxes, and enter the **Default SGT** value.
5. Click **Update & Apply to Device**.

Configure SGACL, inline tagging, and SGT in local mode (CLI)

Use this procedure to configure SGACL, inline tagging, and SGT in local mode using commands.

Use this procedure to configure Secure Group Access Control Lists (SGACL), inline tagging, and Security Group Tag (SGT) enforcement in local mode on a controller using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a policy profile for the WLAN.

   Example

   ```
   Device(config)# wireless profile policy profile-name
   ```

3. Enable CTS inline tagging.

   Example

   ```
   Device(config-wireless-policy)# cts inline-tagging
   ```

   Note

   Configure the **cts manual** on the physical interface. If the **cts manual** is configured on the physical interface and **cts inline-tagging** is not enabled, packets remain tagged at egress in the controller.

4. Enable CTS SGACL enforcement.

   Example

   ```
   Device(config-wireless-policy)# cts role-based enforcement
   ```

5. Optional: Set the default Security Group Tag (SGT).

Example

```
Device(config-wireless-policy)# cts sgt sgt-value
```

Note

SGT is required for a user session only when the client uses open authentication, and not the ISE server.

SGACL and environment data download over REST

Cisco TrustSec leverages a REST-based transport protocol for improved policy provisioning and environment data download from Cisco Identity Services Engine (ISE). This protocol offers enhanced security, reliability, and speed compared to traditional RADIUS methods for delivering Security Group Access Control List (SGACL) policies and environment data.

Cisco TrustSec uses the REST-based transport protocol that

- ensures more secure communication for provisioning Security Group Access Control List (SGACL) policies and environment data compared to older RADIUS protocols
- provides faster and more reliable policy and data downloads across the network,
- simplifies integration between Cisco TrustSec and ISE, supporting efficient delivery and enforcement of security group policies.

Both REST API-based and RADIUS-based protocols for downloading Cisco TrustSec data are supported. Only one protocol can be active on a device at a time. You can change the protocol to RADIUS by configuring the **cts authorization list** command.

Note

Cisco TrustSec Change of Authorization (CoA) will still use RADIUS as the protocol.

Cisco TrustSec SGACL and environment data are synchronized from the active device to the standby device, after the policy is installed. However, REST API connections or sessions are not synchronized during a switchover.

Prerequisites for SGACL and environment data download over REST

Learn the requirements that must be met to ensure successful SGACL and environment data downloads over REST.

To ensure proper operation of SGACL and environment data downloads over REST, observe these requirements:

- Cisco Identity Services Engine (ISE) must use version 2.7 or later.
- Cisco TrustSec-enabled devices must use Cisco IOS XE 26.1.1 or later releases.
- Update the network device configuration on Cisco ISE to allow REST API calls from a network device IP address (NAS-IP). The network device uses the device ID and password specified in the Cisco ISE configuration as its username and password when making REST API calls to Cisco ISE.

Restrictions for SGACL and environment data download over REST

Outlines the requirements you must follow when configuring SGACL and environment data downloads using the REST interface.

To ensure proper operation of SGACL and environment data downloads over REST, observe these requirements:

- Cisco TrustSec Change of Authorization (CoA) uses RADIUS as the protocol.
- Only port 9063 is supported as the ERS server port.
- Server statistics are not persistent after a refresh of the environment data.
- Only one Fully Qualified Domain Name (FQDN) per server is supported.

Cisco TrustSec environment data

Environment data comprises of operational data that supplement Cisco TrustSec functions. The environment data request from a device to Cisco ISE consists of:

- Device name: Specifies the name of the device.
- Device capability: Specifies additional data.

The environment data response from Cisco ISE to a device consists of:

- Device security group tag (SGT): Derived from Cisco ISE based on the device name.
- Server list: Displays the list of Cisco TrustSec servers specified in Cisco ISE.
- SG-Name Table: Displays the mapping between SGT and the device name. SGT is displayed in numerals and the device name in text format.
- Refresh time: Indicates the time when the environment data will be refreshed.

Note

- As part of Cisco TrustSec environment data refresh, the last received servers are deleted and newly received servers are added to the server list. As a result of the refresh, the server list statistics restarts from zero, and the server status is set to *Inactive* and the IP address state is set to *Reachable*. The device then updates the server statistics and status based on the subsequent policy request and response.
- Starting with Cisco IOS XE Bengaluru 17.4.1, you can configure automated tester to be VRF aware. You can use the **vrf** keyword with the **automate-tester** command to enable automate-tester for a non-default VRF.

  For VRF aware automate-tester to work, you have to configure **global config ipv4/ipv6 source interface** interface-name **vrf** vrf-name command.

Message flow between a network device and a server

Learn how REST API communication is established and managed between a network device and a Cisco ISE server.

Summary

This process outlines how secure REST API communication is managed between a network device and a Cisco Identity Services Engine (ISE) server using TLS and authentication steps.

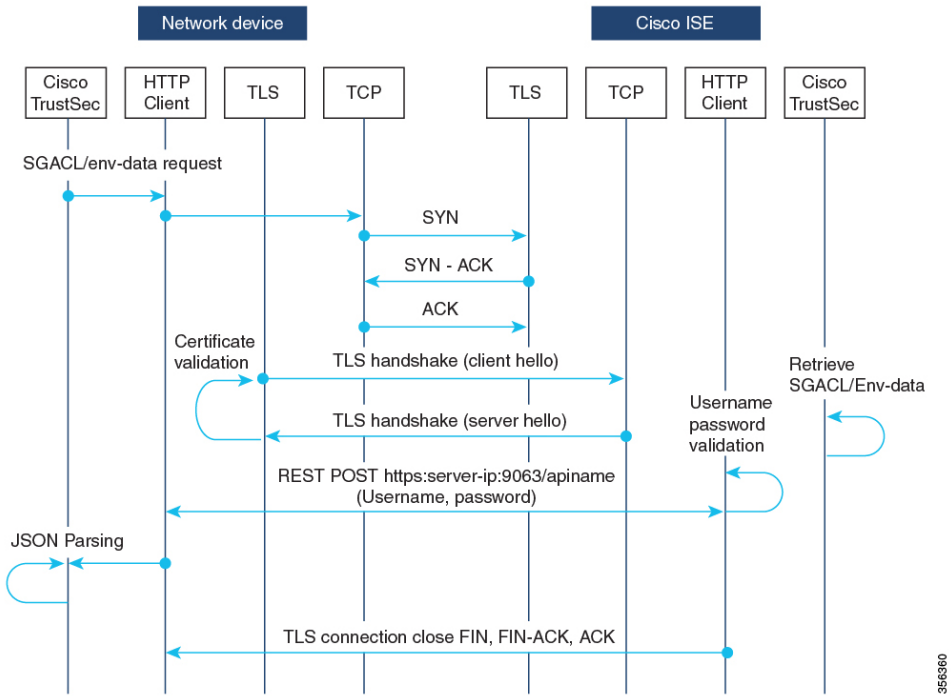
The key components involved in the process are:

- Cisco ISE server: Hosts the REST API service and handles incoming requests over secure connections.
- Network device: Initiates REST API calls to request SGACL and environment data from the server.
- TLS connection: Ensures encrypted, authenticated communication between the device and the server.

The server terminates the connection through a TCP-FIN message after all REST requests are processed.

Workflow

Figure 22: Figure 1. Message flow between a network device and a server



1. Cisco ISE REST API service runs on a secure socket using TLS 1.2 on port 9063. This service handles network device requests for SGACL and environment data.
2. The device uses a make-or-break approach to the TLS connection establishment, and there is no persistent TLS connection between the device and Cisco ISE. The server terminates the connection through a TCP FIN message after all REST requests are processed. For new REST API calls a new connection must be established with the server.
3. The REST API call from the device to Cisco ISE starts with a TCP connection establishment. Cisco ISE must be configured with device IP address to allow ingress connections from the device. If you send a TCP connection request from a source IP address not configured on Cisco ISE, Cisco ISE drops the request and creates an audit log.
4. Username and password–Every REST API call must include username and password authentication while requesting access to a resource uniform resource identifier (URI). The authentication helps the server to determine if the caller should be given access to the resource or to deny the request.
5. "To establish a secure TLS connection with Cisco ISE, you must install its server certificate signature or PEM as the trustpoint on the device. Use the **crypto pki trustpoint** command for this step. Export and install only the fingerprint or signature of the server certificate on the device under a trustpoint. You do not need to import the server certificate's private key.
6. After establishing the TLS connection, the HTTP client on the device initiates a REST call to Cisco ISE on the specified resource.

Result

The process allows devices to securely communicate and exchange environment or policy data with Cisco ISE servers by using authenticated and encrypted REST API sessions.

Policy server selection criteria

Learn how Cisco TrustSec devices select among multiple configured HTTP policy servers, detailing the criteria and algorithms used for server selection.

To explain the criteria and methods used by Cisco TrustSec devices to select a policy server from multiple configured HTTP servers, ensuring efficient communication with Cisco ISE and optimized load distribution.

You can configure multiple HTTP policy servers on your Cisco TrustSec device. After a server is selected, the device uses that server to interact with Cisco ISE until the server is inactive.

There are two types of server selection:

- In-order selection: This is the default behavior, where servers are picked in the order in which they are configured (from the public server list) or downloaded (from the private server list). After the device selects a server, it continues using that server until it becomes inactive. Then, the device selects the next server in the list.

  When environment data is successfully downloaded, and a server-list is available, these servers are added to the private server list.

- Random server selection: When multiple HTTP policy servers are configured on a device, a single Cisco ISE instance may get overloaded if the device always selects the first configured server. To avoid this situation, each device will randomly select a server. The device generates a random number, and based on this number, selects a server. For different devices to generate random numbers, the unique board ID and the Cisco TrustSec process ID of the device is used to initialize the random number generator.

  After the device selects a server, it sends all future requests to this server until it becomes inactive. When a server becomes inactive, the device uses the random server selection logic to pick the next available server. The device does not include inactive servers when selecting a new server. The device numbers servers starting at zero.

  Selected Server = (Generated Random Number) % (Total Number of Active Servers).

To change the server selection logic to random, use the **cts policy-server order random** command.

Server and IP address selection process

Learn how devices select servers and IP addresses using prioritized lists and selection algorithms in Cisco IOS XE.

Summary

The server and IP address selection process determines how a device chooses which server and IP address to use for connecting to Cisco ISE. This process provides failover, supports IPv4 and IPv6, and prioritizes server lists to ensure reliable connectivity.

Workflow

The order of server selection is the private server list, which is received as part of server list download, followed by the public server list, which contains the configured servers. The device selects a server at random or in order, depending on whether you enable the **cts policy-server order random** command.

In Cisco IOS XE 17.2.1 and later releases, each server can have multiple IP addresses (both IPv4 and IPv6). The device selects IPv4 addresses first (IPv6 addresses second), and lastly FQDN.

Here is how the device selects a server and IP address:

- When a device boots up for the first time, a server from the public (configured) list is selected.
- If the **cts environment-data enable** command is configured, the device uses the public server to download the private server-list from Cisco ISE.
- After selecting a server and IP address, the device connects to Cisco ISE using that combination.

- After the server and IP address are selected, the device connects to Cisco ISE using the server and IP address combination. This server will interact with Cisco ISE until it fails to get a response.
- If the device does not receive a response from the current active server in the private list, it switches to the next server. When the server is selected for the first time, the device searches for the first reachable IP or IPv6 address.
- After selecting a server and IP address, the device uses this combination until the device goes down.
- If none of the servers in the private list are reachable, the device attempts to connect to the servers in the public list. The server switching logic and IP selection are the same for private and public list.
- The server change happens only when the server list is refreshed.
- If all servers in both the private and public server list are dead, the device restarts the server and IP address selection logic from the start of the private list.
- When a specific server and IP address combination fails, the device waits for 60 seconds before it attempts a new combination.

Result

The device reliably connects to Cisco ISE by selecting available servers and IP addresses to maximize resilience and compatibility.

Server liveliness check

This topic explains how Cisco ISE checks server availability and how it responds during server selection

Summarize the methods and criteria Cisco ISE uses to assess and respond to server availability in the server selection process.

Cisco ISE determines whether a server is alive after receiving an environment-data or SGACL request. Cisco ISE does not run a liveliness detection phase on the server after it is configured or downloaded as part of a server list. By default, Cisco ISE considers all types of servers to be alive.

When a request is sent to Cisco ISE, if the server is not reachable or the response is lost, the server is marked as dead. The server selection logic will choose the same server and the next IP address (if multiple addresses are configured) to send the next set of Cisco ISE requests. If the device receives an overloaded response (HTTP 429) from Cisco ISE, the logic will select the next server in the list.

A server can be marked as dead because of any of these reasons that include:

- The configured IP address is not reachable.
- Incorrect port number.
- The Cisco ISE instance with the IP address is down.
- The interface towards Cisco ISE is down.
- A Transport Layer Security (TLS) handshake failure.
- An HTTP response timeout.
- An incorrectly configured domain name (if a domain name is used).

If you configure both a static IP address and a domain name for a server, Cisco ISE uses the static IP address first. If there is no response to the static IP address, the device tries with the domain name. If no response is received from both the static IP address and the domain name, the server is marked as dead.

When all servers in the private list are marked as dead, the device uses the public list. If all remaining servers are also marked as dead, the recovery mechanism starts. The device waits for the next Cisco TrustSec request, such as a policy refresh, environment data download, or a similar action. Then it marks all servers as alive to retry the download. If there is no new Cisco TrustSec request, the servers remain in the dead state.

Configure Cisco TrustSec username and password

Configure a Cisco TrustSec username and password on your device to enable REST API-based authentication with Cisco ISE.

Set up a username and password for Cisco TrustSec on your device. This enables REST API-based policy communication with Cisco Identity Services Engine (ISE) for secure network access enforcement.

Before you begin

Configure the username and password in Cisco ISE as the REST API access credentials, before configuring it on the device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a Cisco TrustSec policy server and enter policy-server configuration mode.

   Example

   ```
   Device(config)# cts policy-server name ISE-server
   ```

3. Exit policy-server configuration mode and return to global configuration mode.

   Example

   ```
   Device(config-policy-server)# exit
   ```

4. Configure a username and password.

   Example

   ```
   Device(config)# cts policy-server username username password {0 | 6 |7 |
   password} {password}
   ```

   Note

   This username and password must be created on Cisco ISE as the REST API access credentials before configuring it on the device.

5. Optional: Exit global configuration mode and return to privileged EXEC mode.

Example

```
Device(config)# end
```

The device is now configured to use the specified username and password to communicate with Cisco ISE through REST API for TrustSec operations.

```
Device# configure terminal
Device(config)# cts policy-server name ISE-server
Device(config-policy-server)# exit
Device(config)# cts policy-server username admin password 6 password1
Device(config)# end
```

Download certificate from ISE

Learn how to configure and download the server certificate from Cisco Identity Services Engine (ISE) for secure TLS communications.

Obtain the TLS-based server certificate from ISE to enable secure authentication with external services.

This task is typically performed when you need to export a self-signed server certificate from ISE for use in network authentication protocols or to integrate with other secure services.

Procedure

---

1. Log in to the Identity Services Engine GUI with your administrative user credentials.
2. Choose **Administration** > **Certificates**. The **System Certificates** page is displayed.
3. From the list of certificates, select **Default self-signed server certificate** for TLS-based authentication. The **Used By** column lists the certificate's purpose.
4. Click **Export**.
   The **Export Certificate 'Default self-signed server certificate'** page is displayed.
5. Click the **Export** button in the dialog box.
   The certificate file is downloaded and saved to your local machine.

The system successfully downloads the TLS-based server certificate from ISE. You can now use it in secure authentication workflows.

Configure certificate enrollment

Learn to configure certificate enrollment on your device to establish trust with a Certificate Authority (CA). These steps ensure your device can authenticate and obtain digital certificates required for secure communications.

Use this task to enroll your device with a Certificate Authority, which is necessary for implementing secure certificate-based authentication and encrypted communications within your network.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Define the trustpoint and a name, and enter CA trustpoint configuration mode.

   Example

   ```
   Device(config)# crypto pki trustpoint name
   ```

3. Enroll the CA certificate through the terminal. The certificate is imported through the terminal.

   Example

   ```
   Device(ca-trustpoint)# enrollment terminal
   ```

4. Exit CA trustpoint configuration mode and return to global configuration mode.

   Example

   ```
   Device(ca-trustpoint)# exit
   ```

5. Retrieve the Certificate Authority (CA) certificate and authenticate it. Check the certificate fingerprint if prompted.

   Example

   ```
   Device(config)# crypto pki authenticate name
   ```

   You will receive a message to enter the base 64 encoded certificate. Open the certificate that you had downloaded in your local folder, copy the contents and paste it at the prompt.

   The certificate fingerprints are displayed. Accept the certificate when prompted.

   The trustpoint CA certificate is accepted and successfully imported.

   > **Note**
   >
   > This command is optional if the CA certificate is already loaded into the configuration.

6. Optional: Exit global configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

After completing these steps, your device will have established a trust relationship with the Certificate Authority and be ready to request and use digital certificates for secure operations.

```
Device# configure terminal
Device(config)# crypto pki trustpoint ca-trustpoint
Device(ca-trustpoint)# enrollment terminal
Device(ca-trustpoint)# exit
Device(config)# crypto pki authenticate ca-trustpoint
<Cut and paste the downloaded CA certificate>
Device(config)# end
```

What to do next

Associate the trustpoint with the policy server.

Configure CTS policy server and parameters

Learn how to configure a Cisco TrustSec (CTS) policy server and specify its settings to enable secure policy enforcement within your network.

Define the parameters and set up a Cisco TrustSec policy server to enable secure policy enforcement within your network for certificate enrollment, communication, and security.

Before you begin

- Configure certificate enrollment policy.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a Cisco TrustSec policy server and enter policy-server configuration mode.

   Example

   ```
   Device(config)# cts policy-server name ISE-server
   ```

3. Configure the domain name address, IPv4 address, or IPv6 address of the policy server.

   Example

   ```
   Device(config-policy-server)# address {domain-name name| ipv4
   policy-server-address | ipv6 policy-server-address}
   ```

   IPv6 addresses are supported starting with Cisco IOS XE 26.1.1.

4. Configure the Transport Layer Security (TLS) certificate trustpoint.

   Example

   ```
   Device(config-policy-server)# tls server-trustpoint trustpoint-name
   ```

5. Optional: Configure the response timeout in seconds.

   Example

   ```
   Device(config-policy-server)# timeout seconds
   ```

   The range is from 1 to 60 seconds. The default is 5 seconds.

6. Optional: Configure the maximum number of retries from the server.

Example

```
Device(config-policy-server)# retransmit number-of-retries
```

The range is from 0 to 5. The default is 4.

7. Optional: Specify the policy server port number.

Example

```
Device(config-policy-server)# port port-number
```

The range is from 1025 to 65535.

> **Note**
>
> The ERS server port number must be 9063. You cannot change this port number.

8. Optional: Set the content type so the system sources SGACL and environment data from Cisco ISE.

Example

```
Device(config-policy-server)# content-type json
```

9. Optional: Exit policy-server configuration mode and return to privileged EXEC mode.

Example

```
Device(config-policy-server)# end
```

The CTS policy server uses specified parameters, enabling secure communication and policy enforcement for your network devices.

```
Device# configure terminal
Device(config)# cts policy-server name ISE-server
Device(config-policy-server)# address ipv6 2001.DB8::1
Device(config-policy-server)# tls server-trustpoint tls15
Device(config-policy-server)# timeout 15
Device(config-policy-server)# retransmit 4
Device(config-policy-server)# port 9063
Device(config-policy-server)# content-type json
Device(config-policy-server)# end
```

Download environment data

This feature enables the device to retrieve and apply environment-specific configuration data from Cisco ISE. This data is required for secure network access and policy enforcement.

This task allows you to configure your device to download environment data from Cisco ISE. Obtaining this information ensures that policies are applied accurately, and it enables seamless integration with your network access devices.

Before you begin

The source interface to use for HTTP connections must be specified in the **ip http client source-interface** command.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the policy server device ID to send environment data requests to Cisco ISE.

   Example

   ```
   Device(config)# cts policy-server device-id device-id
   ```

   Use the same device-ID that you used to add the network access device (NAD) in Cisco ISE.

3. Enable environment data downloads from Cisco ISE.

   Example

   ```
   Device(config)# cts environment-data enable
   ```

   > **Note**
   >
   > The **cts environment-data enable** command and the **cts authorization list** command are mutually exclusive. These commands cannot be configured together.

4. Optional: Exit global configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

Your device is now configured to download environment data from Cisco ISE. This step keeps policy and access-control configuration synchronized with the central ISE server.

```
Device# configure terminal
Device(config)# cts policy-server device-id server1
Device(config)# cts environment-data enable
Device(config)# end
```

Debug the SGACL and environment data over REST configuration

This reference lists debug commands used to troubleshoot SGACL and environment data configurations over REST, including command syntax and functional descriptions.

This reference provides a list of debug commands to assist in troubleshooting SGACL and environment data configurations over REST.

Use these **debug** commands for debugging the configuration.

- **debug cts policy-server http**

  Enables HTTP client debugging.

- **debug cts policy-server json**

  Enables JSON client debugging.

Verify the SGACL and environment data download over REST

This reference section presents command outputs and error descriptions used to verify SGACL and environment data downloads through REST.

To provide reference information for commands used to verify SGACL and environment data downloads over REST. You will find instructions for viewing policy server details, interpreting server statistics, understanding error codes, and displaying environment data server lists.

To display information about the specified policy server, use this command:

```
Device# show cts policy-server details name ise_server_1
Server Name   : ise_server_1
Server Status : Active
  IPv4 Address     : 10.64.69.84
  IPv6 Address     : 2001:DB::2
  Trustpoint       : ISE84
  Port-num         : 9063
  Retransmit count : 3
  Timeout          : 15
  App Content type : JSON
```

To display statistics information about active policy servers, use this command:

If you use the command without the **active**, the statistics of all servers will be listed.

```
Device# show cts policy-server statistics active
Server Name  : ise_server_1
Server State : ALIVE
  Number of Request sent        : 7
  Number of Request sent fail   : 0
  Number of Response received   : 4
  Number of Response recv fail  : 3
    HTTP 200 OK                 : 4
    HTTP 400 BadReq             : 0
    HTTP 401 UnAuthorized Req   : 0
    HTTP 403 Req Forbidden      : 0
    HTTP 404 NotFound           : 0
    HTTP 408 ReqTimeout         : 0
    HTTP 415 UnSupported Media  : 0
    HTTP 500 ServerErr          : 0
    HTTP 501 Req NoSupport      : 0
    HTTP 503 Service Unavailable: 0
    TCP or TLS handshake error  : 3
    HTTP Other Error            : 0
```

Table 136: List of errors displayed in the show command and their descriptions

Errors	Description
HTTP 200 OK	Indicates that the client's request to the server has been processed.

Errors	Description
HTTP 400 BadReq	Indicates that the server cannot process the request due to a problem on the client's end.
HTTP 401 UnAuthorized Req	Indicates that the request has not been completed because it lacks valid authentication credentials for the requested resource.
HTTP 403 Req Forbidden	Indicates that the server understood the request but refuses to authorize it. This is often due to insufficient permissions or, less commonly, server misconfiguration.
HTTP 404 NotFound	Indicates that the web server cannot find the requested resource.
HTTP 408 ReqTimeout	Indicates that the server did not receive a complete request from the client within the time that the server was prepared to wait.
HTTP 415 UnSupported Media	Indicates that the server refuses to accept the request because the payload format is in a format not supported by the server for the target resource.
HTTP 500 ServerErr	Indicates that the server encountered an unexpected condition preventing it from fulfilling a request.
HTTP 501 Req NoSupport	Indicates that the server cannot fulfill the request because it does not support the required functionality.
HTTP 503 Service Unavailable	Indicates a temporary server failure, often due to overloading, maintenance, or misconfigured application pools.
TCP or TLS handshake error	Occurs when a client and server fail to establish a secure, encrypted connection.

To view the list of servers that are downloaded as part of the environment data, use this command.

These servers are included in the private server list.

This output shows the HTTP based download information:

```
Device# show cts server-list
HTTP Server-list:
  Server Name      : cts_private_server_0
  Server State     : ALIVE
  IPv4 Address     : 10.64.69.151
  IPv6 Address     : 2001:DB8:8086:6502::
  IPv6 Address     : 2001:db8::2
  IPv6 Address     : 2001:db8::402:99
  IPv6 Address     : 2001:DB8:4::802:16
  Domain-name      : ise-267.cisco.com
  Trustpoint       : cts_trustpoint_0

  Server Name      : cts_private_server_1
  Server State     : ALIVE
  IPv4 Address     : 10.10.10.3
  IPv4 Address     : 10.10.10.2
```

```
IPv6 Address     : 2001:DB8::20
IPv6 Address     : 2001:DB8::21
Domain-name      : www.ise.cisco.com
Trustpoint       : cts_trustpoint_1
```

Cisco TrustSec - ISE communication over IPv6

Outlines Cisco TrustSec's IPv6 capabilities, including policy-centric security management, RADIUS-based ISE communication, enforcement of security and identity policies, SGACL provisioning, and IPv6 address support to enable secure, scalable network modernization.

IPv6 transition for Cisco TrustSec

Cisco TrustSec is a security framework that

- implements business security needs through policy-centric management,
- provides integrated security features such as segmentation, identity management, access control, and data protection,
- facilitates communication between network devices and policy servers using the RADIUS protocol over IPv6.

This framework allows organizations to enforce security policies effectively across their networks, adapting to the growing demand for IPv6 addressing.

The transition to IPv6 for Cisco TrustSec involves:

- Supporting IPv6 addresses for end-user policy enforcement and identity management.
- Enabling communication with the Identity Services Engine (ISE) for provisioning Security Group ACL (SGACL) policies over IPv6.
- Maintaining a list of IPv6 addresses and load balancing them during policy download.

This transition is essential for organizations looking to modernize their network security infrastructure and ensure compatibility with future technologies.

Limitations of IPv4

The limitations of IPv4 addressing include:

- **Limited address space**: Due to its 32-bit addressing system, IPv4 is limited to approximately 4.3 billion unique addresses, which is insufficient for the increasing number of internet-connected devices.
- **Dependency on NAT**: It requires Network Address Translation (NAT) to extend the address space, which complicates network design and can hinder connectivity.
- **Lack of native support**: It lacks native support for modern features such as end-to-end encryption and mobility.
- **Routing inefficiencies**: It creates inefficiencies in routing tables, leading to increased processing overhead for routers.

Configure ISE for TrustSec (CLI)

Use this procedure to configure ISE for TrustSec.

Set up ISE to enable secure network access using TrustSec policies using commands.

This task involves configuring a device to communicate with an ISE server as a RADIUS server and integrating TrustSec authorization.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the RADIUS server name.

   Example

   ```
   Device(config)# radius server server-name
   ```

3. Specify the primary RADIUS server parameters.

   Example

   ```
   Device(config-radius-server)# address ipv6 ip-address
   ```

4. Specify the authentication and encryption key used between the Device and the key string RADIUS daemon running on the RADIUS server.

   Example

   ```
   Device(config-radius-server)# pac key key
   ```

5. Return to the configuration mode.

   Example

   ```
   Device(config-radius-server)# exit
   ```

6. Create a radius server-group identification.

   Example

   ```
   Device(config)# aaa group server radius server-group
   ```

   Note

   server-group refers to the server group name. The valid range is from 1 to 32 alphanumeric characters.

7. Create a CTS authorization list.

   Example

   ```
   Device(config)# cts authorization list mlist-name
   ```

8. Create an authorization method list for web-based authorization.

Example

```
Device(config)# aaa authorization network mlist-name group group-name
```

Note

Ensure that the ISE IP address configured on your controller is the same as the IP address configured on ISE (**Work Center** > **TrustSec** > **Components** > **Trustsec AAA Servers**).

Note

If the ISE version is 002.005(000.239), 002.004(000.357), 002.003(000.298), 002.002(000.470), 002.001(000.474), 002.000(001.130), or 002.000(000.306), use the **access-session tls-version 1.0** command to download PAC from ISE. For other ISE versions, the above command is not required.

Configure TrustSec SXP with IPv6 (GUI)

Use this task to configure TrustSec to support IPv6 communication for enhanced security policy management.

Enable IPv6 communication for Cisco TrustSec to enhance security policy management.

Use this task to configure RADIUS server settings to support IPv6 addresses for TrustSec operations.

Before you begin

- Ensure that your network devices are capable of supporting IPv6.
- Verify that the Identity Services Engine (ISE) is configured to accept IPv6 connections.

Procedure

1. Choose **Configuration** > **Security** > **Trustsec**.
2. On the Trustsec page, click the **SXP** tab. The **Peer Connections** section is displayed.
3. In the **SXP Parameters** section, enable the **SXP Status**.
4. Specify the **Default Source IPv6** address and the default password.
5. In the **Peer Connections** section, click **Add**.
6. On the **Add Peer Connection** page, configure these parameters:
   a) From the **Mode of Local Device** drop-down list, select either **listener**, **speaker**, or **both**.
   b) From the **IP Type** drop-down list select **IPv6**.
   c) Enter the **Peer IP** and the **Source IP** addresses.
   d) Enter the **Password**.
   e) Click **Apply to Device**.

7. Click **Apply**.

IPv6 is now enabled for TrustSec operations and RADIUS communications.

Configure TrustSec SXP with IPv6

Use this task to configure TrustSec to support IPv6 communication for enhanced security policy management.

Enable IPv6 communication for Cisco TrustSec to enhance security policy management.

Use this task to configure RADIUS server settings to support IPv6 addresses for TrustSec operations.

Before you begin

- Ensure that your network devices are capable of supporting IPv6.
- Verify that the Identity Services Engine (ISE) is configured to accept IPv6 connections.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure CTS SXP with IPv6 address.

   Example

   ```
   Device(config)# cts sxp default source-ipv6 <X:X:X:X::X source ipv6 address>
   ```

3. Complete these steps to configure the CTS SXP connection parameters.
   a) Configure CTS SXP connection peer IP address. You can configure either IPv4 or IPv6 IP address of the peer.
   b) Set the password option. You can choose a default password, or use a configured key-chain for authentication, or opt for no password.
   c) Set the connection role as either local or peer and then set an SXP mode.
   d) Specify the VRF. If the SXP connection should be in a specific VRF, include the VRF name.

   Example

   ```
   Device(config)# cts sxp connection peer {A.B.C.D | X:X:X:X::X}
                                              password {default | key-chain
    | none}
                                              mode {local | peer} {both |
   listener hold-time | speaker hold-time}
                                              vrf <vrf-name>
   ```

   ```
   Device(config)# cts sxp connection peer 2001:db8::1 password default mode
    local both vrf mgmt-vrf
   ```

4. Configure CTS SXP node ID.

Example

```
Device(config)# cts sxp node-id {interface | ipv4 | ipv6}
```

> **Note**
>
> The node ID is a 32-bit value. When enabling SXP over IPv6, a node ID must be explicitly configured; an IPv6 address cannot be used as the node ID.

5. Configure CTS SXP export-import VRF group.

Example

```
Device(config)# cts sxp export-import-group <group-name>
```

6. Configure CTS SXP export-import VRF group.

Example

```
Device(config)# cts sxp export-import-group <group-name>
```

TrustSec SXP is successfully configured to support IPv6 communication, enabling enhanced security policy management and seamless connectivity between devices using IPv6 addresses.

Verify Cisco TrustSec PAC over IPv6 RADIUS

To verify Cisco TrustSec PAC over IPv6 RADIUS, use these **show** commands:

To view the A-ID and PAC information for PACs in the key store, use this command:

```
Device# show cts pacs
AID: 60DABBF8AD435A7B63900EE07E68D2FD
PAC-Info:
  PAC-type = Cisco Trustsec
  AID: 60DABBF8AD435A7B63900EE07E68D2FD
  I-ID: ha_senth_cat9k
  A-ID-Info: Identity Services Engine
  Credential Lifetime: 16:17:20 UTC Mon Feb 24 2025
PAC-Opaque:
[redacted]
Refresh timer is set for 12w4d
```

To view the queue for provisioning requests, use this command:

```
Device# show ctw provisioning queue
Server: 20XX:420:54XX:4::400:11, Type: Radius, Provisioned: YES
AID: 3ec5d7e02bde9fd04c453918a5e2d3b4
```

To view the Cisco TrustSec environment data, use this command:

```
Device# show cts environment-data
CTS Environment Data
```

```
====================
Current state = COMPLETE
Last status = Successful
Service Info Table:
Local Device SGT:
  SGT tag = 2-48:TrustSec_Devices
Server List Info:
Installed list: CTSServerList1-0003, 3 server(s):
 *Server: 2001:420:54FF:4::400:11, port 1812, A-ID
1695AF86D38B22DC7C9500408E2DD35D
          Status = DEAD
          auto-test = TRUE, keywrap-enable = FALSE, idle-time = 60 mins,
deadtime = 20 secs
 *Server: 2001:420:54FF:4::400:12, port 1812, A-ID
1695AF86D38B22DC7C9500408E2DD35D
          Status = ALIVE
          auto-test = TRUE, keywrap-enable = FALSE, idle-time = 60 mins,
deadtime = 20 secs
 *Server: 2001:420:54FF:4::400:13, port 1812, A-ID
1695AF86D38B22DC7C9500408E2DD35D
          Status = DEAD
          auto-test = TRUE, keywrap-enable = FALSE, idle-time = 60 mins,
deadtime = 20 secs
Security Group Name Table:
    0-48:Unknown
    2-48:TrustSec_Devices
    3-62:Network_Services
    4-56:Employees
    5-48:Contractors
    6-49:Guests
    7-48:Production_Users
    8-49:Developers
    9-49:Auditors
    10-48:Point_of_Sale_Systems
    11-48:Production_Servers
    12-0177:Development_Servers
    13-48:Test_Servers
    14-56:PCI_Servers
    15-48:BYOD
    16-01:sgt_101
Environment Data Lifetime = 86400 secs
Last update time = 05:26:38 UTC Tue Jul 1 2025
Env-data expires in   0:23:52:36 (dd:hr:mm:sec)
Env-data refreshes in 0:23:52:36 (dd:hr:mm:sec)
Cache data applied           = NONE
State Machine is running
Retry_timer (60 secs) is not running
```

TrustSec IP-SGT High Availability binding

Introduces TrustSec IP-to-SGT High Availability binding, highlighting how it synchronizes IP-to-SGT bindings between active and standby devices, preserves bindings during SSO, and ensures consistent SGACL enforcement for uninterrupted policy application during failover events.

TrustSec IP-to-SGT binding High Availability is a redundancy capability that

- synchronizes supported IP-to-SGT bindings between the active and standby devices
- preserves binding information during a Stateful Switchover (SSO), and
- helps maintain consistent Security Group Access Control List (SGACL) enforcement after the switchover.

An IP-to-SGT binding associates an IP address with a Security Group Tag (SGT). TrustSec uses this association to classify traffic and apply the appropriate security policy.

Without binding synchronization, a new active device may need to re-establish connections and relearn bindings after a switchover. Policy enforcement can be delayed while this process occurs. High-availability support makes the required binding information available on the standby device before it assumes the active role.

The list of source-bindings supported for binding synchronization in Cisco TrustSec are:

- Local source: IP-SGT binding received through authentication of hosts, learned through Endpoint Policy Manager (EPM) and device tracking.
- SXP source: IP-SGT binding received through Security Group Tag Exchange Protocol (SXP) peers.
- VLAN source: IP-SGT binding received through VLAN-to-SGT mapping.
- L3IF binding: Binding received through Layer 3 Logical Interface (L3IF) learning.

Verify TrustSec IP-SGT High Availability binding support

As part of the system's synchronization functionality, active bindings are automatically propagated to and displayed on the standby unit to ensure High Availability and consistent policy enforcement across the cluster.

To view all the mappings of IP address to Security Group Tag, use this command:

```
Device# show cts role-based sgt-map all
Active IPv4-SGT Bindings Information

IP Address              SGT     Source
============================================
1.1.1.1                 3       CLI
20.1.1.1                1010    SXP
20.1.1.2                1001    SXP
22.1.1.1                2       SXP
33.1.1.2                2       SXP
55.1.1.1                3       SXP
200.1.1.1               9001    SXP
200.1.1.2               9002    SXP
200.1.1.3               9003    SXP
200.1.1.4               9004    SXP
200.1.1.5               9005    SXP
200.1.1.6               9006    SXP
200.1.1.7               9007    SXP
200.1.1.8               9008    SXP
200.1.1.9               9009    SXP
200.1.1.10              9010    SXP
200.1.1.11              9011    SXP
200.1.1.12              9012    SXP
200.1.1.13              9013    SXP
200.1.1.14              9014    SXP
200.1.1.15              9015    SXP
200.1.1.16              9016    SXP
200.1.1.17              9017    SXP
200.1.1.18              9018    SXP
200.1.1.19              9019    SXP
200.1.1.20              9020    SXP
200.1.1.21              9021    SXP
200.1.1.22              9022    SXP
200.1.1.23              9023    SXP
200.1.1.24              9024    SXP
200.1.1.25              9025    SXP
200.1.1.26              9026    SXP
200.1.1.27              9027    SXP
```

```
200.1.1.28              9028    SXP
200.1.1.29              9029    SXP
200.1.1.30              9030    SXP

IP-SGT Active Bindings Summary
============================================
Total number of CLI      bindings = 1
Total number of SXP      bindings = 35
Total number of active   bindings = 36

Active IPv6-SGT Bindings Information

IP Address                                    SGT     Source
============================================================
```

To view the standby mappings, use this command:

```
standby-device# show cts role-based sgt-map all
Active IPv4-SGT Bindings Information

IP Address              SGT     Source
============================================
1.1.1.1                 3       CLI
20.1.1.1                1010    SXP
20.1.1.2                1001    SXP
22.1.1.1                2       SXP
33.1.1.2                2       SXP
55.1.1.1                3       SXP
200.1.1.1               9001    SXP
200.1.1.2               9002    SXP
200.1.1.3               9003    SXP
200.1.1.4               9004    SXP
200.1.1.5               9005    SXP
200.1.1.6               9006    SXP
200.1.1.7               9007    SXP
200.1.1.8               9008    SXP
200.1.1.9               9009    SXP
200.1.1.10              9010    SXP
200.1.1.11              9011    SXP
200.1.1.12              9012    SXP
200.1.1.13              9013    SXP
200.1.1.14              9014    SXP
200.1.1.15              9015    SXP
200.1.1.16              9016    SXP
200.1.1.17              9017    SXP
200.1.1.18              9018    SXP
200.1.1.19              9019    SXP
200.1.1.20              9020    SXP
200.1.1.21              9021    SXP
200.1.1.22              9022    SXP
200.1.1.23              9023    SXP
200.1.1.24              9024    SXP
200.1.1.25              9025    SXP
200.1.1.26              9026    SXP
200.1.1.27              9027    SXP
200.1.1.28              9028    SXP
200.1.1.29              9029    SXP
200.1.1.30              9030    SXP

IP-SGT Active Bindings Summary
============================================
Total number of CLI      bindings = 1
```

```
Total number of SXP      bindings = 35
Total number of active   bindings = 36

Active IPv6-SGT Bindings Information

IP Address                                    SGT     Source
============================================================
```

153 SGT Inline Tagging and SXPv4

Topics:

- SGT inline tagging on AP and SXPv4

Introduces Security Group Tags and Security Exchange Protocol version 4 as foundational components of Cisco TrustSec architecture for network segmentation and policy enforcement.

SGT inline tagging on AP and SXPv4

Introduces SGT and SXPv4 concepts, and guides users through creating SXP profiles, configuring SGT inline tagging on APs, establishing SXP connections using GUI and CLI, and verifying successful SGT push to APs.

SGT inline tagging on AP and SXPv4 is a set of Cisco TrustSec enhancements that

- enables secure propagation of scalable group tag (SGT) information across network devices using the SXP protocol
- introduces loop detection in SXP version 4 to prevent stale binding in the network, and
- supports embedding SGTs in clear-text Ethernet packets with inline tagging.

When a wireless client is authenticated by Cisco Identity Services Engine (ISE), the IP-SGT binding is generated on the controller and pushed to the access point along with other client details.

Additional information

Cisco TrustSec (CTS) builds secure networks by establishing domains of trusted network devices. CTS uses SGTs and the SXP protocol to securely exchange group information and enhance network segmentation.

For details on SGT inline tagging on the AP and SXP version 4, see the Cisco TrustSec Configuration Guide.

Create an SXP profile

Create an SXP profile.

Configuring an SXP profile is essential for enabling Cisco TrustSec features in a wireless environment.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a wireless CTS profile and enter cts-sxp profile configuration mode.

   Example

   ```
   Device(config)# wireless cts-sxp profile profile-name
   ```

   The *profile-name* is the name of the SXP profile being created.

3. Enable SXP for Cisco TrustSec.

   Example

   ```
   Device(config-cts-sxp-profile)# cts sxp enable
   ```

The SXP profile is now created and SXP is enabled for Cisco TrustSec.

Configure SGT inline tagging on APs

Use this procedure to configure SGT inline tagging on access points (APs).

Enable SGT inline tagging on APs to support dynamic SGT allocation through ISE authentication.

Follow the procedure given below to configure SGT inline tagging on APs.

Before you begin

Ensure that the SGTs pushed to the AP for inline tagging will only be from dynamic SGT allocation through ISE authentication. It is not supported for static bindings configured on the controller . SGTs will be pushed to an AP only when it is operating in flex mode. To know the list of Cisco APs that support SGT inline tagging, see the release notes: https://www.cisco.com/c/en/us/support/wireless/catalyst-9800-series-wireless-controllers/products-release-notes-list.html.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

   Enters the global configuration mode.

2. Configure a wireless flex profile and enter the wireless flex profile configuration mode.

   Example

   ```
   Device(config)# wireless profile flex flex-profile
   ```

   The *flex-profile* is the name of the wireless flex profile being configured.

3. Enable inline-tagging on the AP.

   Example

   ```
   Device(config-wireless-flex-profile)# cts inline-tagging
   ```

   Enables inline-tagging on the AP.

SGT inline tagging is now configured on the access points, allowing for dynamic SGT allocation.

Configure an SXP connection (GUI)

Learn how to configure an SXP connection using the graphical user interface (GUI) by following step-by-step instructions for global, peer, and AP profile settings.

Configure an SXP (Security Group Tag Exchange Protocol) connection using the graphical user interface (GUI) to enable secure policy exchange between network devices.

Procedure

1. In the **Global** section, select the **SXP Enabled** check box to enable SXP.
2. Enter an IP address in the **Default Source IP** field.

3. Enter a value in the **Reconciliation Period (sec)** field.
4. Enter a value in the **Retry Period (sec)** field.
5. Select the **Set New Default Password** check box. Selecting this check box displays the **Password Type** and **Enter Password** fields.
6. Choose any one of the available types from the **Password Type** drop-down list.
7. Enter a value in the **Enter Password** field.
8. Click the **Apply** button.
9. In the **Peer** section, click the **Add** button.
10. Enter an IP address in the **Peer IP** field.
11. Enter an IP address in the **Source IP** field.
12. Choose any one of the available types from the **Password** drop-down list.
13. Choose any one of the available types from the **Mode of Local Device** drop-down list.
14. Click the **Save & Apply to Device** button.
15. In the **AP** tab, click the **Add** button. The **Add SXP AP** dialog box appears.
16. Enter a name for the profile in the **Profile Name** field.
17. Set the **Status** field to **Enabled** to enable AP.
18. Enter a value in the **Default Password** field.
19. Enter a value (in seconds) for the **CTS Speaker Seconds, CTS Recon Period, CTS Retry Period, CTS Listener Maximum** , and **CTS Listener Minimum**
20. In the **CTS SXP Profile Connections** section, click **Add** .
21. Enter an IP address in the **Peer IP** field.
22. Choose any one of the modes from the **Connection Mode** drop-down list. The available modes are **Both** , **Listener** , and **Speaker** .
23. From the **Password Type** drop-down list, choose either **None** or **Default** .
24. Click the **Add** button.
25. Click the **Save & Apply to Device** button.

You have a fully configured SXP connection, including global settings, peer devices, and AP (Access Point) profiles, ready for secure operation in your network.

Configure an SXP connection

Use this procedure to configure an SXP connection using commands.

Follow the procedure given below to configure an SXP connection:

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. Enable CTS SXP support.

Example

```
Device(config)# cts sxp enable
```

Enables CTS SXP support.

3. Configure the CTS-SXP peer address connection.

Example

```
Device(config)# cts sxp connection peer ipv4-address password password-value
 mode mode-value speaker
```

Configures the CTS-SXP peer address connection.

> **Note**
>
> The password need not be *none* always and the mode can either be Speaker or Listener, or Both.

What to do next

Use the following command to verify the configuration:

```
Device# show running-config | inc sxp
```

Verify SGT push to APs

Verifies SGT push to AP.

When a wireless client is connected and authenticated by ISE, the IP-SGT binding is generated on the controller . This can be verified using these commands:

```
Device# show cts role-based sgt-map all

Active IPv4-SGT Bindings Information

IP Address              SGT     Source
============================================
1.1.1.1                 100     CLI

IP-SGT Active Bindings Summary
============================================
Total number of CLI      bindings = 1
Total number of active   bindings = 1
```

Use this command to verify the SXP connections status:

```
Device# show cts sxp connections

 SXP              : Enabled
 Highest Version Supported: 4
```

```
 Default Password : Not Set
 Default Source IP: Not Set
Connection retry open period: 120 secs
Reconcile period: 120 secs
Retry open timer is running
Peer-Sequence traverse limit for export: Not Set
Peer-Sequence traverse limit for import: Not Set
----------------------------------------------
Peer IP          : 198.51.100.1
Source IP        : 198.51.100.2
Conn status      : On
Conn version     : 4
Conn capability  : IPv4-IPv6-Subnet
Conn hold time   : 120 seconds
Local mode       : SXP Listener
Connection inst# : 1
TCP conn fd      : 1
TCP conn password: none
Hold timer is running
Duration since last state change: 0:00:00:06 (dd:hr:mm:sec)


Total num of SXP Connections = 1
```

Use this command to see the bindings learnt over SXP connection:

```
Device# show cts role-based sgt-map all

Active IPv4-SGT Bindings Information

IP Address              SGT     Source
============================================
1.1.1.1                 100     CLI

IP-SGT Active Bindings Summary
============================================
Total number of CLI      bindings = 1
Total number of active   bindings = 1
```

Use these commands on the AP to check the status of inline tagging on the AP and its IP-SGT bindings:

```
AP# show capwap client rcb

AdminState                          : ADMIN_ENABLED
OperationState                      : UP
Name                                : AP2C33.1185.C4D0
SwVer                               : 192.6.230.41
HwVer                               : 1.0.0.0
MwarApMgrIp                         : 192.0.2.38
MwarName                            : test-ewlc
MwarHwVer                           : 0.0.0.0
Location                            : default location
ApMode                              : FlexConnect
ApSubMode                           : Not Configured
CAPWAP Path MTU                     : 1485
CAPWAP UDP-Lite                     : Enabled
IP Prefer-mode                      : IPv4
AP Link DTLS Encryption             : OFF
AP TCP MSS Adjust                   : Disabled
LinkAuditing                        : disabled
```

```
Efficient Upgrade State            : Disabled
Flex Group Name                    : anrt-flex
AP Group Name                      : default-group
Cisco Trustsec Config
      AP Inline Tagging Mode                : Enabled
! The status can be Enabled or Disabled and is based on the tag that is pushed
 to the AP.
      AP Sgacl Enforcement                  : Disabled
      AP Override Status                    : Disabled
```

```
AP# show cts role-based sgt-map all

Active IPv4-SGT Bindings Information
        IP SGT SOURCE
9.3.74.101  17  LOCAL

IP-SGT Active Bindings Summary
============================================
Total number of LOCAL    bindings = 1
Total number of active   bindings = 1

Active IPv6-SGT Bindings Information
                       IP SGT SOURCE
fe80::c1d5:3da2:dc96:757d  17  LOCAL

IP-SGT Active Bindings Summary
============================================
Total number of LOCAL    bindings = 1
Total number of active   bindings = 1
```

154 Ciphersuites for CAPWAP-DTLS

Topics:

- Default ciphersuites supported for CAPWAP-DTLS

Default ciphersuites supported for CAPWAP-DTLS

Lists the default ciphersuites used for CAPWAP-DTLS, AP priorities, platform exclusions, and configuration guidance.

Starting with Cisco IOS XE Bengaluru 17.5.1, the Elliptic Curve Diffie-Hellman Ephemeral (ECDHE) or Galois Counter Mode (GCM) ciphersuite with perfect forward secrecy (PFS) is included in the default list, together with the existing AES128-SHA ciphersuite. All Cisco AP models, except Cisco IOS APs, prioritize the PFS ciphersuite for CAPWAP-DTLS under the default configuration.

> **Note**
>
> If link encryption is enabled for secure data channel traffic, the COS AP (DTLS client) prioritizes DHE-RSA-AES128-SHA over the ECDHE or GCM ciphersuite.

- The preference order of the ciphersuites during the DTLS handshake is important. You can set the priority order when configuring cipher suites using this feature.
- When explicit ciphersuites are not configured, the default ciphersuites listed in the table apply.

Table 137: Default Ciphersuites

Security Mode	Ciphersuite
FIPS and non-FIPS	·TLS_RSA_WITH_AES_128_CBC_SHA · TLS_DHE_RSA_WITH_AES_128_CBC_SHA · TLS_DHE_RSA_WITH_AES_256_CBC_SHA · TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 · TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 · TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 · TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
WLANCC	·TLS_DHE_RSA_WITH_AES_128_CBC_SHA · TLS_DHE_RSA_WITH_AES_256_CBC_SHA · TLS_DHE_RSA_WITH_AES_256_CBC_SHA256 · TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 · TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 · TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256

This feature is supported on all variants of Cisco Catalyst 9800 Series Wireless Controllers and APs, except for Cisco Industrial Wireless 3702 AP.

Refer to the release notes for a list of supported controllers and APs for each release: https://www.cisco.com/c/en/us/support/wireless/catalyst-9800-series-wireless-controllers/products-release-notes-list.html.

Configure multiple ciphersuites (CLI)

Use this procedure to configure multiple ciphersuites using commands.

Configure the allowed ciphersuites for AP DTLS on the controller using commands.

Note

- If a controller is loaded with a startup configuration using a ciphersuite selection configuration that predates Cisco IOS XE Bengaluru 17.5.1, it is automatically converted to the latest ciphersuite selection version.
- Changing the ciphersuite configuration causes APs to disconnect and reconnect.
- If you downgrade to a Cisco IOS XE Bengaluru version earlier than 17.5.1, the ciphersuite configurations are lost.
- When downgrading to a version below 17.12.1 in FIPS mode or WLANCC mode, verify that the ECDHE-RSA-AES128-GCM-SHA256 cipher suite is selected for AP DTLS. By default, it is selected. Otherwise, the downgrade causes connectivity issues for all Cisco Wave 2 APs.
- This can be verified by using the **show wireless certification config** command.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Set priority for a particular cipher suite.

   Example

   ```
   Device(config)# ap dtls-ciphersuite priority-num ciphersuite
   ```

   Use zero to set the highest priority.

   Note

   Any configuration change will automatically disconnect all currently connected APs.

3. Return to the privileged EXEC mode.

   Example

   ```
   Device(config)# exit
   ```

Set server preference (CLI)

Use this procedure to set server preference using commands.

Configure whether the server dictates ciphersuite priority order during a DTLS handshake for AP join profiles using commands.

The ciphersuite configuration enforces the priority order during the DTLS handshake. To give equal priority to all configured ciphersuites, use the **no ciphersuite server-preference** command in the corresponding AP join profile. By default, server preference is enabled.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

3. Set the cipher suite server preference.

   Example

   ```
   Device(config-ap-profile)# [no] ciphersuite server-preference
   ```

   Use the **no** form of this command to disable server preference. By default, server preference is enabled.

4. Return to global configuration mode.

   Example

   ```
   Device(config)# exit
   ```

Verify operational ciphersuites and priority

This topic describes the verifying operational ciphersuites and priority command.

To view the operational ciphersuites and their priority, use this command:

```
Device# show wireless certification config

WLANCC                          : Not Configured
AP DTLS Version                 : DTLS v1.0 - v1.2

AP DTLS Cipher Suite List:

  Priority                Ciphersuite
------------------------------------------------------------------------------

   0                      AES128-SHA
   1                      DHE-RSA-AES256-SHA256
```

155 Configure Secure Shell

Topics:

- SSH

Introduces Secure Shell protocol fundamentals, configuration options, and security considerations for establishing encrypted remote connections to network devices.

> **Note**
>
> Starting with, Secure Shell Version 1 (SSHv1) is deprecated.

SSH

Explains Secure Shell (SSH), a network security protocol that establishes encrypted and authenticated connections between remote devices, protecting data confidentiality and integrity during communication.

A secure shell (SSH) protocol is a network security mechanism that

- establishes encrypted and authenticated connections between remote devices. and
- protects data confidentiality and integrity during communication.
- supports multiple version standards, including SSHv1 and SSHv2.

How SSH sorks and security benefits

Secure Shell (SSH) protocols offer a secure alternative to older remote access methods, such as Telnet, by encrypting data exchanged between a client and a server. Authentication further ensures that only authorized users can access devices. SSH increases security for network administrators by preventing eavesdropping and protecting credentials.

- Connecting securely to a router or switch for remote configuration.
- Transferring files securely between devices using SCP, which operates over SSH.

SSH and device access

Explains how Secure Shell (SSH) provides a secure, remote connection to a device, offering enhanced security over Telnet by using strong encryption and supporting both IPv4 and IPv6 environments.

A Secure Shell (SSH) protocol is a network access method that

- uses strong encryption and authentication to secure remote connections
- offers enhanced security over Telnet by encrypting session and authentication data, and
- supports both IPv4 and IPv6 environments for flexible device access.

Additional information

SSH is a secure protocol for remote device access, supporting both IPv4 and IPv6, and provides strong encryption for authenticated sessions.

SSH servers and integrated clients

The Secure Shell (SSH) Integrated Client feature is an application that runs over the SSH protocol to provide device authentication and encryption.

A SSH integrated client is a feature that - runs over the SSH protocol to provide device authentication - enables secure, encrypted connections to other devices, and - supports various user authentication methods.

Supported versions and functionalities

The SSH client enables a Cisco device to make a secure, encrypted connection to another Cisco device or to any other device running the SSH server.

This connection provides functionality similar to that of an outbound Telnet connection except that the connection is encrypted.

With authentication and encryption, the SSH client allows for secure communication over an unsecured network.

The SSH server and SSH integrated client are applications that run on the switch.

The switch supports an SSHv1 or an SSHv2 server.

The switch supports an SSHv1 client.

> **Note**
>
> The SSH client functionality is available only when the SSH server is enabled.

User authentication is performed like that in the Telnet session to the device. SSH also supports the following user authentication methods:

- TACACS+
- RADIUS
- Local authentication and authorization

SSH configuration guidelines

Provides guidelines for configuring the switch as an SSH server or SSH client.

General SSH configuration guidelines

Follow these guidelines when configuring the switch as an SSH server or SSH client:

- An RSA key pair generated by a SSHv1 server can be used by an SSHv2 server, and the reverse.
- If the SSH server is running on an active switch and the active switch fails, the new active switch uses the RSA key pair generated by the previous active switch.
- If you get CLI error messages after entering the **crypto key generate rsa** global configuration command, an RSA key pair has not been generated. Reconfigure the hostname and domain, and then enter the **crypto key generate rsa** command.
- When generating the RSA key pair, the message No host name specified might appear. If it does, you must configure a hostname by using the **hostname** global configuration command.
- When generating the RSA key pair, the message No domain specified might appear. If it does, you must configure an IP domain name by using the **ip domain-name** global configuration command.
- When configuring the local authentication and authorization authentication method, make sure that AAA is disabled on the console.

SCP

Provides an overview of Secure Copy Protocol (SCP).

A Secure Copy Protocol is a method for securely transferring files that

- provides a secure and authenticated method for copying switch configurations or switch image files
- relies on Secure Shell (SSH) for secure transport, and
- requires correct configuration of SSH, authentication, and authorization on the switch.

Supporting reference information

For SSH to work, the switch needs an RSA public or private key pair. This is the same with SCP, which relies on SSH for its secure transport.

Because SSH also relies on AAA authentication, and SCP relies further on AAA authorization, correct configuration is necessary.

- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the router must have a Rivest, Shamir, and Adelman (RSA) key pair.

> **Note**
>
> When using SCP, you cannot enter the password into the copy command. You must enter the password when prompted.

Secure copy protocol

The Secure Copy Protocol (SCP) feature provides a secure and authenticated method for copying device configurations or switch image files.

A Secure Copy Protocol is a network protocol that

- provides a secure and authenticated method for copying files
- relies on SSH for security, and
- requires authentication, authorization, and accounting (AAA) authorization.

Additional information on SCP

The behavior of SCP is similar to that of remote copy (rcp), which comes from the Berkeley r-tools suite.

To configure the SCP, you should understand the SCP concepts.

SFTP support

Provides Secure File Transfer Protocol (SFTP) support.

SFTP client support is introduced from Cisco IOS XE Gibraltar 16.10.1 release onwards. SFTP client is enabled by default and no separate configuration required. The SFTP procedures can be invoked using the **copy** command, which is similar to that of **scp** and **tftp** commands. A typical file download procedure using **sftp** command can be carried out as shown below:

copy sftp://*user* **:***password* **@***server-ip***/***file-name* **flash0://** *file-name*

For more details on the **copy** command, see the following URL:
https://www.cisco.com/c/m/en_us/techdoc/dc/reference/cli/nxos/commands/fund/copy.html

Prerequisites for configuring SSH

Outlines prerequisites for configuring secure shell.

- For SSH to work, the switch needs an Rivest, Shamir, and Adleman (RSA) public/private key pair. This is the same with Secure Copy Protocol (SCP), which relies on SSH for its secure transport.
- Before enabling SCP, you must correctly configure SSH, authentication, and authorization on the switch.
- Because SCP relies on SSH for its secure transport, the router must have an Rivest, Shamir, and Adelman (RSA) key pair.
- SCP relies on SSH for security.

- SCP requires that authentication, authorization, and accounting (AAA) authorization be configured so the router can determine whether the user has the correct privilege level.
- A user must have appropriate authorization to use SCP.
- A user who has appropriate authorization can use SCP to copy any file in the Cisco IOS File System (IFS) to and from a switch by using the **copy** command. An authorized administrator can also do this from a workstation.
- The Secure Shell (SSH) server requires an IPsec (Data Encryption Standard [DES] or 3DES) encryption software image; the SSH client requires an IPsec (DES or 3DES) encryption software image.
- Configure a hostname and host domain for your device by using the **hostname** and **ip domain-name** commands in global configuration mode.

Restrictions for configuring SSH

Restrictions for configuring secure shell.

These are restrictions for configuring the device for secure shell.

- From Cisco IOS XE Dublin 17.10.x, Key Exchange and MAC algorithms like diffie-hellman-group14-sha1, hmac-sha1, hmac-sha2-256, and hmac-sha2-512 are not supported by default and it may impact some SSH clients that only support these algorithms. However, you can add them manually if required. For information on manually adding these algorithms, see the https://www.cisco.com/c/en/us/td/docs/routers/ios/config/17-x/sec-vpn/b-security-vpn/m_sec-secure-shell-algorithm-ccc.html.
- The switch supports Rivest, Shamir, and Adelman (RSA) authentication.
- SSH supports only the execution-shell application.
- The SSH server and the SSH client are supported only on Data Encryption Standard (DES) (56-bit) and 3DES (168-bit) data encryption software. In DES software images, DES is the only encryption algorithm available. In 3DES software images, both DES and 3DES encryption algorithms are available.
- The device supports the Advanced Encryption Standard (AES) encryption algorithm with a 128-bit key, 192-bit key, or 256-bit key. However, symmetric cipher AES to encrypt the keys is not supported.
- When using SCP, you cannot enter the password into the **copy** command. You must enter the password when prompted.
- The login banner is not supported in Secure Shell Version 1. It is supported in Secure Shell Version 2.
- The -l keyword and userid :{number} {ip-address} delimiter and arguments are mandatory when configuring the alternative method of Reverse SSH for console access.
- To authenticate clients with FreeRADIUS over RADSEC, you should generate an RSA key longer than 1024 bit. Use the **crypto key generate rsa general-keys exportable label** *label-name* command to achieve this.

Set up the device to run SSH

Use this procedure to set up the device to run SSH.

Enable SSH on the device for secure remote access.

Follow the procedure given below to set up your device to run SSH:

Before you begin

Configure user authentication for local or remote access.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the hostname and IP domain name for your device.

   Example

   ```
   Device(config)# hostname your_hostname
   ```

   Note

   Follow this procedure only if you are configuring the device as an SSH server.

3. Configure a host domain for your device.

   Example

   ```
   Device(config)# ip domain name your_domain
   ```

4. Enable the SSH server and generate an RSA key pair.

   Example

   ```
   Device(config)# crypto key generate rsa
   ```

   Generating an RSA key pair for the device automatically enables SSH.

   We recommend that a minimum modulus size of 1024 bits.

   When you generate RSA keys, you are prompted to enter a modulus length. A longer modulus length might be more secure, but it takes longer to generate and to use.

   Note

   Follow this procedure only if you are configuring the device as an SSH server.

5. Exit configuration mode.

   Example

   ```
   Device(config)# end
   ```

The device is now set up to run SSH, allowing secure remote access.

Configure the SSH server

Learn how to configure the SSH server to allow secure remote access to the device.

Set up the SSH server to allow secure remote access to the device.

Follow the procedure given below to configure the SSH server:

> **Note**
>
> This procedure is only required if you are configuring the device as an SSH server.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Optional: Configure the device to run SSH Version 2.

   Example

   ```
   Device(config)# ip ssh version 2
   ```

   If you do not enter this command or do not specify a keyword, the SSH server selects the latest SSH version supported by the SSH client.

3. Specify the SSH window size.

   Example

   ```
   Device(config)# ip ssh window-size
   window-size-value
   ```

   The recommended window size is 32K or less. The default window size is 8912.

   Selecting **window-size** greater than 32K might have some impact on the CPU, unless:

   - The network bandwidth is good.
   - Client can accommodate this size.
   - No latency in network.

> **Note**
>
> This CLI is recommended only for SCP operations and can be disabled once the copy is done.

4. Configure the SSH control parameters.

   Example

   ```
   Device(config)# ip ssh timeout
   seconds
   authentication-retries
   number
   ```

   Specify the time-out value in seconds; the default is 120 seconds. The range is 0 to 120 seconds. This parameter applies to the SSH negotiation phase.

   By default, up to five simultaneous, encrypted SSH connections for multiple CLI-based sessions over the network are available (session 0 to session 4).

5. Optional: Use one or both of the following:

   - **line vty** *line_number ending_line_number*
   - **transport input ssh**

   Example

   ```
   Device(config)# line vty
   line-number
   ending-line-number
   ```

   or

   ```
   Device(config-line)# transport input ssh
   ```

   Enters line configuration mode to configure the virtual terminal line settings. For *line-number* and *ending-line-number* , specify a pair of lines. The range is 0 to 15.

   Specifies that the device prevent non-SSH Telnet connections. This limits the router to only SSH connections.

6. Return to privileged EXEC mode.

   Example

   ```
   Device(config-line)# end
   ```

   Returns to privileged EXEC mode.

The SSH server is now configured and ready for secure remote access.

Monitor the SSH configuration and status

Provides commands for displaying the SSH server configuration and status.

This table displays the SSH server configuration and status.

Command	Purpose
show ip ssh	Shows the version and configuration information for the SSH server.
show ssh	Shows the status of the SSH server.

156 MACsec support on APs

Topics:

- MACsec support on AP
- Auto MACsec for wireless APs

Introduces MACsec support on APs, describing key terminology, supported AP models, MACsec modes, supported ciphers, and restrictions. Details step-by-step configuration and verification procedures for enabling MACsec with PSK and 802.1X authentication across controllers, APs, and switches.

MACsec support on AP

MACsec support on AP is a security feature that encrypts and protects the integrity of data transmitted between devices using the IEEE 802.1X standard.

MACsec protection is a network security feature that

- uses the IEEE 802.1AE standard to apply hop-by-hop link encryption between the AP and the access switch
- secures data confidentiality and integrity to defend against attacks such as denial of service, intrusion, eavesdropping, and man-in-the-middle, and
- supports both pre-shared key (PSK) and IEEE 802.1X authentication methods.

Feature History

Feature Name	Release	Description
MACsec support on APs	Cisco IOS XE 26.1.1	With this release, MACsec support is introduced on APs to provide Layer 2 hop-by-hop encryption and integrity protection for data transmitted between the AP and the connected switch. This feature leverages the IEEE 802.1AE standard and ensures robust security against eavesdropping and man-in-the-middle attacks without impacting AP performance. These commands are introduced: • **key chain** *chain name* **macsec** • **ap profile** *name* **macsec** • **show ap macsec summary** • **show macsec** • **debug macsec**
Failover mechanism	Cisco IOS XE 26.1.2	This release introduces automatic MACsec failover for AP wired uplinks. If an AP cannot detect a valid MACsec peer, it falls back after a timeout to maintain controller connectivity. On APs with multiple uplinks, fallback occurs only when MACsec fails on all applicable links; if any uplink establishes MACsec successfully, the AP continues secure operation on that link.

Feature Name	Release	Description
MACsec replay protection window support	Cisco IOS XE 26.1.2	With this release, a configurable replay protection window has been added to securely accommodate out-of-order packet delivery. The setting defines how many sequence numbers may arrive out of order before packets are identified as potential replay attacks and dropped.

Expanded explanation

This feature uses MACsec (IEEE 802.1AE) to secure the link between the AP and the access switch. It employs hop-by-hop link encryption to mitigate attacks such as denial of service, intrusion, eavesdropping, and man-in-the-middle attacks.

Common MACsec terminology

This section lists the key MACSec security and key management terms used in this release, along with their definitions.

- **MKA**: or MACsec Key Agreement is the protocol that creates, distributes, and manages the encryption keys used by MACsec. It is defined in **IEEE 802.1X-2010** and works on top of EAPOL (Extensible Authentication Protocol over LAN).
- **MSK**: or Master Session Key is generated during EAP exchange. Supplicant and authentication server use the MSK to generate the CAK.
- **CAK**: or Connectivity Association Key is derived from MSK. CAK is a long-lived master key used to generate all other keys used for MACsec.
- **CKN**: or Connectivity Association Key Name identifies the CAK.
- **SAK**: or Secure Association Key is derived from the CAK and is the key used by supplicant and switch to encrypt traffic for a given session.
- **KS**: or Key Server is responsible for selecting and advertising a cipher suite. KS is responsible for generating the Secure Association Key (SAK) from the Connectivity Association Key (CAK).
- **Key server priority**: is a value used by MKA (MACsec Key Agreement) to determine which device becomes the Key Server in a MACsec session.
- **Key chain**: is a configuration structure used to store and manage cryptographic keys, mainly for protocols that require authentication or key exchange. For MACsec, the key chain holds the CAK (Connectivity Association Key) and CKN (Connectivity Key Name) used by MKA (MACsec Key Agreement).

Supported AP

Find the list of supported Cisco Catalyst access point models that support MACsec across Wi-Fi 7, Wi-Fi 6E, and Wi-Fi 6 generations. This helps you identify compatible hardware for different wireless standards, including CW917x and CX91xx series devices. Use this information to select appropriate APs for your network deployment.

You can find a list of supported APs in the Feature Matrix for Cisco Wireless Access Points.

MACsec modes, authentication options, and supported ciphers

Lists the MACsec authentication modes supported, including PSK and IEEE 802.1X with its available EAP methods.

Here are the modes supported by MACsec.

- Pre-Shared Key (PSK)
- IEEE 802.1X
  - Extensible Authentication Protocol - Transport Layer Security (EAP-TLS)
  - Extensible Authentication Protocol - Flexible Authentication through Secure Tunneling (EAP-FAST)

Here are the ciphers supported by MACsec.

- AES-128-GCM
- AES-256-GCM

Restrictions

Lists the configuration restrictions for MACsec.

The AP cannot act as a key server (KS).

MACsec failover mechanism

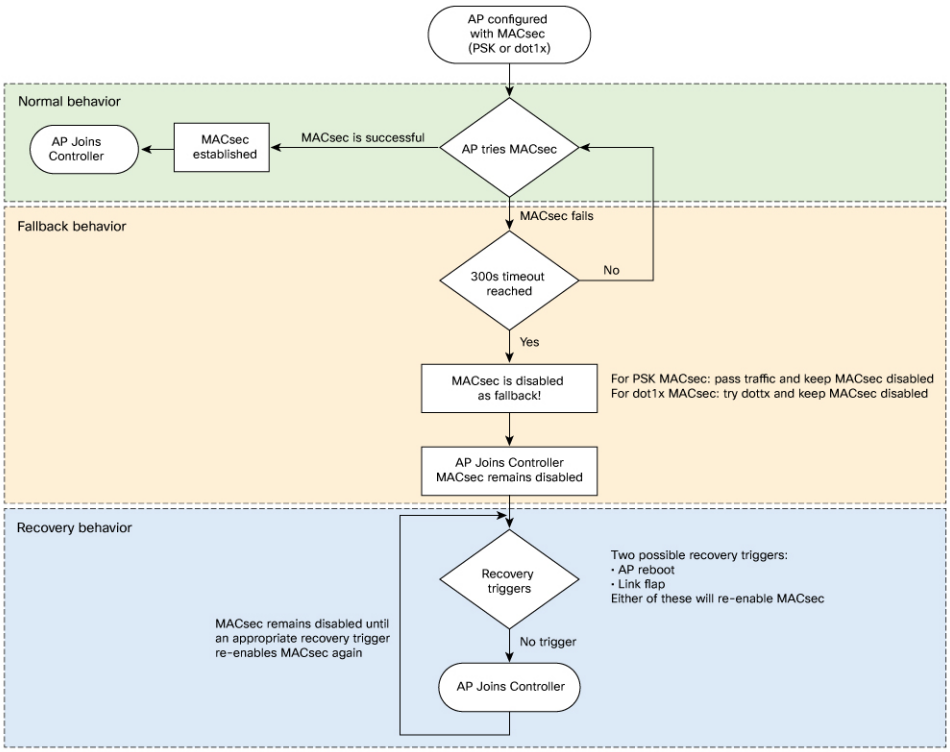
Describes how an access point (AP) manages retry attempts and timeout periods when attempting a connection.

Summary

When MACsec is enabled, the AP checks its wired uplinks for a valid MACsec peer. If MACsec cannot be established on any applicable uplink within the timeout period, the AP reports the failure to the controller and falls back to maintain connectivity. If at least one uplink establishes MACsec, the AP continues secure operation on that link. After fallback, MACsec restart remains blocked until recovery is triggered by an AP reboot, or by a controller rejoin followed by a wired-port link flap.

Workflow

Figure 23: MACsec failure mechanism



1. The AP starts with MACsec configured using **PSK** or **802.1X**.
2. The AP attempts to establish MACsec on all applicable wired uplinks.
3. If MACsec succeeds:
   - MACsec is enabled.
   - The AP joins the controller securely.
4. If MACsec fails:
   - The AP continues retrying.
   - The process checks whether the **300-second timeout** has expired.
5. If the timeout has not expired, the AP retries MACsec.
6. When the timeout expires, the AP enters fallback mode:
   - PSK MACsec: The AP passes traffic with MACsec disabled.
   - 802.1X MACsec: The AP attempts 802.1X authentication while keeping MACsec disabled.
7. The AP joins the controller with MACsec remaining disabled.
8. MACsec stays disabled until one of these recovery triggers occurs:
   - AP reboot
   - Wired-link flap
9. After a recovery trigger, the AP enables MACsec processing and attempts to establish the session again.

Result

The AP remains connected when MACsec establishment fails by falling back after 300 seconds. MACsec stays disabled until an AP reboot or link flap triggers recovery. Wireless service during fallback depends on the AP Join Profile configuration.

Configuring MACsec with PSK

This process describes configuring MACsec using pre-shared key (PSK) mode to enable secure encryption between a switch and an access point (AP), including required steps, configuration elements, and verification commands.

MACsec can be deployed in PSK mode. This process requires that you configure the same password on both network endpoints. The Connectivity Association Key (CAK), which serves as the PSK, and the Connectivity Association Key Name (CKN) must match on both sides.

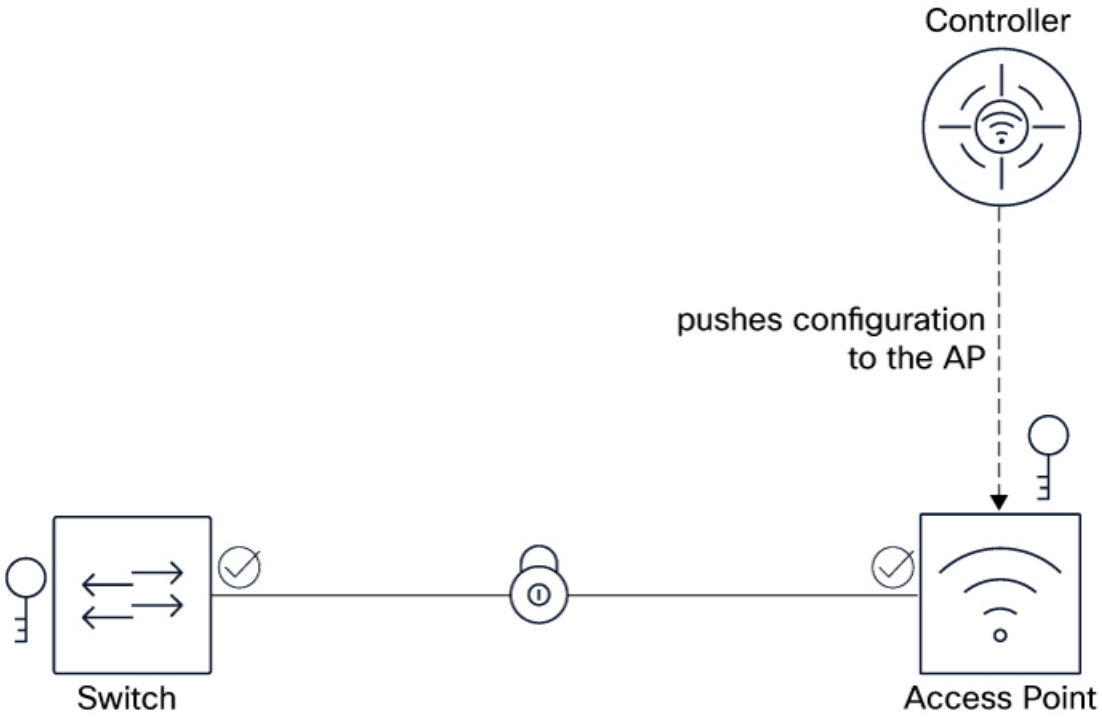
Summary

The key components involved in the process are:

- **Controller:** For key chain creation and AP join profile configuration.
- **Switch**: For key chain setup, MKA policy configuration, and interface settings.
- **AP**: Joins the network under controller management and uses the shared keys for secure communication.

Workflow

Figure 24: MACsec PSK flow diagram



The process involves these stages:

1. *Configure MACsec PSK on the controller*: On the controller, add a new key chain and configure it with a key. The key string (PSK or CAK) is a hex string that matches with the switch.

   In this PSK scenario, the CAK (same as PSK) and the CKN must be manually entered.

2. *Configure MACsec PSK on the AP join profile*: On the controller, configure an AP join profile, enable MACsec, and select the previously created key chain.

3. *Configure MACsec PSK on the switch*: Create a key chain, configure the MKA policy, and configure the desired interface.

4. *Verify MACsec configuration:* Use various show commands on the controller, AP, and the switch.

Result

After completing these steps, MACsec PSK is successfully enabled between the AP and the switch. You now have encrypted and authenticated communication across the network.

Configure MACsec PSK on controller (GUI)

Enable MACSec to secure data transmission between the access point and the connected switch.

Enable MACsec to secure data transmission between the AP and the connected switch. This ensures network traffic between the AP and switch is encrypted, enhancing security.

MACsec must be configured on both the AP and the switch to ensure proper encryption and security.

Before you begin

Ensure that the AP is connected to a switch that supports MACsec.

Procedure

1. Choose **Configuration** > **Security** > **MACsec**.

2. Select the new MACsec key chain.
3. In the **Add Key Chain** window that is displayed, configure the key chain.
   a) Add a **Key Identifier**. This is a unique identifier for the key used in the MACsec protocol. Enter a hexadecimal value with an even number of digits. This is the **Connectivity Association Key name** (CKN) and must match on both sides.
   b) Select a **Crypto Algorithm** to protect the control-plane communications.

      You can choose either AES-128-CMAC or AES-256-CMAC .

      > **Note**
      >
      > Data-plane encryption is set by the Key server, which is the switch.

   c) Select a **Key String Type**.
      - Clear Text: Stores and transmits the key string in an unencrypted, readable format.
      - Hidden: Masks the key string from clear display, though it may still be retrievable.
      - Encrypted: Stores and transmits the key string in encrypted form for maximum security.

   d) Enter the **Key String**. This must be a hexadecimal string with 32 digits for AES-128, 64 digits for AES-256. This is the **Pre-Shared Key** (PSK), which is equivalent to the **Connectivity Association Key** (CAK). This key must match on both network endpoints.
   e) Add a **Key Name**. This is the key chain name.
4. Click **Save**. Click **Apply to Device**.
5. Choose **Configuration > Tags & Profiles** > **AP Join**.
6. Click **Add** to create a new AP Join profile.
7. In the **Add AP Join Profile** window, click the **AP** tab.
8. Perform these steps:
   a) Enable or disable MACsec on the AP wired interface.

      By default, MACsec is disabled.

      When enabled, communication between the AP and the connected switch is secured using MACsec encryption.

   b) Select a **Pre-Shared Key Chain**.

      Choose the key chain configured earlier in the **Add Key Chain** window.

      This key chain is used to configure the Pre-Shared Key (PSK) Chain for MACsec authentication and to derive the Secure Association Key (SAK).

      If IEEE 802.1X is enabled, it takes priority over the PSK configuration.
9. Save the configuration and apply the changes.

MACsec is successfully enabled on the AP. This secures the data transmission with the connected switch.

What to do next

Monitor the MACsec status and performance to ensure proper operation.

Configure MACsec PSK on controller(CLI)

Learn how to configure MACsec security on a wireless LAN controller by creating a MACsec key chain, specifying encryption parameters, and enabling MACsec on an AP's wired interface using the CLI.

To configure MACsec and an AP join profile on the controller to enable secure, encrypted communication on AP wired interfaces.

This task creates a MACsec key chain, defines encryption parameters, and applies MACsec settings within an AP join profile. This ensures that data in transit between APs and the wired infrastructure is protected.

Procedure

1. Enter the global configuration mode.

```
Device(config)# configuration terminal
```

2. Create a MACsec key chain.

```
Device(config)# key chain chain-name macsec
```

The device enters the keychain MACsec configuration mode.

3. Configure the MACsec key parameters.

```
Device(config-keychain-macsec)# key Hex String
```

The device enters the keychain MACsec key configuration mode.

4. Configure the MACsec cryptographic authentication algorithm.

```
Device(config-keychain-macsec-key)# cryptographic-algorithm algorithm
```

Values are AES-128-CMAC or AES-256-CMAC .The default value is AES-128-CMAC.

5. Configure the MACsec key string.

```
Device(config-keychain-macsec-key)# key-string key_string
```

6. Exit the keychain MACsec key configuration mode.

```
Device(config-keychain-macsec-key)# exit
```

7. Exit the keychain MACsec configuration mode.

```
Device(config-keychain-macsec)# exit
```

MACsec is enabled on the controller

```
Device(config)#key chain MACsecKeyChain macsec
Device(config-keychain-macsec)#key ABCDEF
Device(config-keychain-macsec-key)# cryptographic-algorithm aes-128-cmac
Device(config-keychain-macsec-key)# key-string ABCDEF0123456789ABCDEF0123456789
Device(config-keychain-macsec)# exit
Device(config-keychain)# exit
```

What to do next
Apply the key to the AP's wired interfaces.

Apply MACsec PSK on AP(CLI)

Enable MACsec on an AP’s wired interface using the CLI.

The task aims to apply a pre-shared key (PSK) on an AP using the controller CLI and enable MACsec encryption on the AP’s wired interface. This enhances security by encrypting communication over the wired Ethernet links between the controller and the AP.

Procedure

1. Enter the global configuration mode.

```
Device(config)# configuration terminal
```

2. Enter the AP profile configuration mode.

```
Device(config)# ap profile <name>
```

3. Apply the MACsec PSK configured earlier.

```
Device(config-ap-profile)# macsec psk-chain Pre-shared-key-name
```

4. Exit the global configuration mode.

MACsec is enabled on the AP wired interfaces, providing encrypted communication over the wired Ethernet links.

Verify MACsec configuration on controller (CLI)

Learn how to verify MACsec encryption on the controller to ensure secure packet transmission.

To verify that MACsec is correctly configured on the controller, ensuring that the AP profile is properly set, MACsec is enabled with the correct key chain and cipher suite, and that secure sessions are established and transmitting encrypted and validated packets.

Verify the AP profile and ensure that MACsec is enabled and the correct key chain is applied.

```
Device# show ap profile name <ap-profile-name> detailed
AP Profile Name                :<ap-profile-name>
[…]
Macsec :
  Enabled                       : True
  Auth Method                   : PSK
  PSK chain name                : MACsecKeyChain
  Replay protection window size: 0
```

Verify the MACsec summary.

```
Device# show ap macsec summary
AP Name                             AP Mac                              Capable
   Port 0          Port 1
-------------------------------------------------------------------------------------
AP11AA.22BB.33CC                    99ff.88ee.77dd                      Yes
SUCCESS         UNKNOWN
```

Verify MACsec configuration on AP

Learn how to verify the macsec configuration and operational status on an Access Point to ensure secure communication.

Verify and troubleshoot MACsec configuration and operational status on an AP.

Ensure secure communication by confirming MACsec encryption is enabled, the correct cipher suite and key chain are in use, and the AP authentication and physical layer status are properly functioning.

Check the MACsec status.

> **Note**
>
> The output of this command varies according to the AP model.

```
Device# show macsec status
-----------------------
wired0: Phy Address 0
-----------------------
MACsec 100M: Enabled
      1000M: Enabled
      2500M: Enabled
      5000M: Enabled
Capabilities:
    Ciphers supported: GCM-AES-128
                       GCM-AES-256
Egress SC:
    AN Roll Over: Disabled
    Egress Protect Frames: True
    Egress Cipher: GCM-AES-128
Ingress SC:
    Replay Protect: False
    Replay Window: 0
    AN Roll Over: Disabled
    Validate Frames: Strict
Device# show macsec statistics
-----------------------
wired0: Phy Address 0
-----------------------
Egress SC:
    Protected Not Encrypted Packets: 0
    Protected and Encrypted Packets: 1185
    Plain Text Octets Protected Not Encrypted: 0
    Plain Text Octets Protected and Encrypted: 567830
Egress SA:
    Dropped Packets: 0
    Protected Not Encrypted Packets: 0
    Protected and Encrypted Packets: 1185
Egress Common:
    Control Packets: 445
    Unknown Packets: 2
    Untagged Packets: 0
    Too Long Packets: 0
    ECC Error Packets: 0
    Dropped Packets: 0
-----------------------
Ingress SA:
    Untagged Packets: 0
```

```
    Dropped Packets: 0
    Not Using Packets: 0
    UnUsed Packets: 0
    Not Valid Packets: 0
    Invalid Packets: 0
    Validated Packets: 2149
    Late Packets: 0
    Delayed Packets: 0
    Unchecked Packets: 0
    Octets of Plaintext Not Encrypted: 0
    Octets of Plaintext Encrypted: 359574
Ingress Common:
    Control Packets: 445
    Tagged but Miss Match Packets: 0
    Untagged and Miss Match Packets: 1
    Untagged and Match that Validate is Strict Packets: 1
    Untagged and Match that Validate is Not Strict Packets: 0
    Invalid Packets: 0
    Unknown SCI and Match that Validate is Strict Packets: 0
    Unknown SCI and Match that Validate is Not Strict Packets: 0
    Controlled Port Pass the Check Packets: 2149
    Uncontrolled Port Pass the Check Packets: 1
    Controlled Port Failed the Check Packets: 0
    UnControlled Port Failed the Check Packets: 0
    Too Long Packets: 0
    Control Packets by Post-MACSec Filter: 0
    ECC Error Packets: 0
    Uncontrolled Port Dropped Packets: 0
```

Confirm that MACsec is enabled and the correct cipher suite is in use.

Verify AP authentication status.

```
Device# show ap authentication status
Wired Link Status:
wired0 link: Up

Wired 0 Session:
key_mgmt=NONE
wpa_state=COMPLETED
address=99:ff:88:ee:77:dd
PAE KaY status=Active
Authenticated=No
Secured=Yes
Failed=No
Actor Priority=255
Key Server Priority=200
Is Key Server=No
Number of Keys Distributed=0
Number of Keys Received=1
MKA Hello Time=2000
actor_sci=8c:88:81:54:6e:70@1
key_server_sci=aa:11:22:bb:33:dd@27
participant_idx=0
ckn=abcdef
mi=a1d1eb46248ea1b53a883f3c
mn=525
active=Yes
participant=No
retain=No
live_peers=1
potential_peers=0
```

```
is_key_server=No
is_elected=Yes
```

- "Secured = Yes": Indicates that MACsec is in place.
- "Failed=No": Indicates if the MACsec establishment failed.

Debug the MACsec summary.

```
Device# debug macsec phy
wpa_supplicant logs:
debug ap authentication {error | events | information | packet}
```

Configuring MACsec with IEEE 802.1X

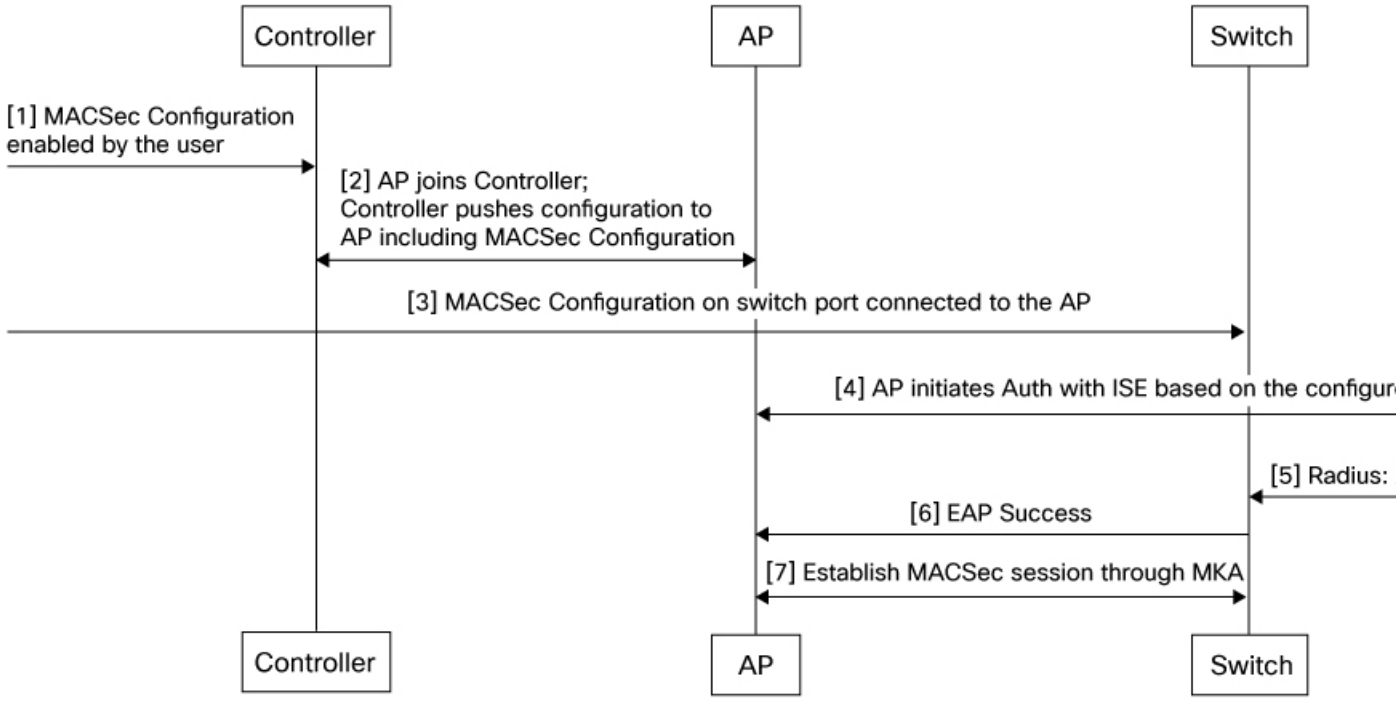
Learn how to configure MACsec with IEEE 802.1X authentication on controllers, APs, and switches for encrypted and authenticated network access.

MACsec can also be configured with IEEE 802.1X. Only two IEEE 802.1X flavors are supported. Here are the basic differences:

Table 138: EAP Types

EAP Type	Client Certificate	Server Certificate	Username
EAP-TLS	Yes	Yes	No
EAP-FAST	No	No	Yes

Figure 25: MACsec IEEE 802.1X flow chart



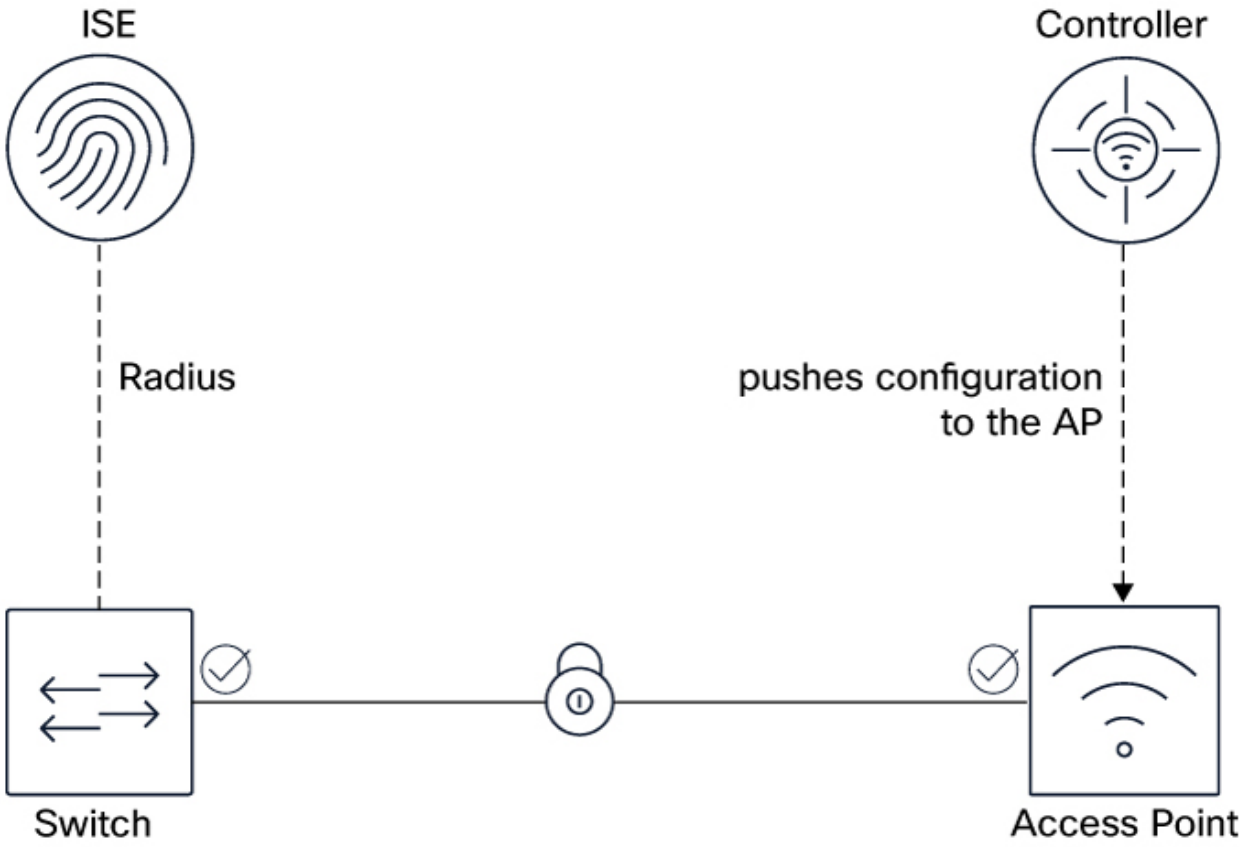
Summary

The key components involved in the process are:

- **Controller:** For IEEE 802.1X confguration, key chain creation, and AP Join Profile configuration.
- **Switch**: For IEEE 802.1X confguration, MKA policy configuration, and interface settings.
- **RADIUS Server**: For authentication.
- **AP**: Joins the network under the controller management and after authenticating with 802.1X establishes secure communication using MACsec.

Workflow

Figure 26: Understanding MACsec IEEE 802.1X



Ensure you have a working IEEE 802.1X setup before attempting MACsec.

The process involves these stages in this recommended order:

1. Configure IEEE 802.1X on the switch, AP, controller, and RADIUS server.
2. Configure MACsec on th AP using the controller.
3. Configure MACsec on the switch.
4. Verify MACsec configuration: Use various show commands on the controller, AP, and the switch.

What's next

For more details, see:

- Configure 802.1X Supplicant for Access Points with 9800 Controller
- Configure 802.1X on APs for PEAP or EAP-TLS with LSC

- EAP-FAST Authentication with Wireless LAN Controllers and Identity Services Engine

Configure MACsec IEEE 802.1X on controller (GUI)

Enable MACsec to secure data transmission between the access point and the connected switch.

IEEE 802.1X provides robust, centralized authentication for devices connecting to your network, ensuring that only authorized users and devices can access network resources. Unlike PSK, which relies on a shared password, IEEE 802.1X leverages individual credentials and a RADIUS server for enhanced security and better management.

IEEE 802.1X provides strong Layer 2 security by authenticating devices and encrypting all traffic between the APs and its uplink switch, ensuring secure and protected data transmission.

Procedure

1. Choose **Configuration > Tags & Profiles** > **AP Join**.
2. Click **Add**.
3. In the **Add AP Join Profile** window, click the **AP** tab.
4. In the **AP MACsec Configuration** section, configure these values:
   a) Click **MACSec** to enable or disable MACsec on the AP wired interface.

      MACsec is disabled by default.

   b) Leave the **Pre-Shared Key Chain** field empty.

      A pre-shared key chain is not required for this configuration.
5. Click **Save**. Click **Apply to Device**.

What to do next

Configure MACsec on the switch. For example, refer to MACsec with encryption on Catalyst 9300

Verify MACsec configuration on controller (CLI)

Learn how to verify MACsec encryption on the controller to ensure secure packet transmission.

To verify that MACsec is correctly configure on the controller, ensuring that the AP profile is properly set, MACsec is enabled with the correct key chain and cipher suite, and that secure sessions are established and transmitting encrypted and validated packets.

Verify MACsec configuration and operational status

This task involves confirming the MACsec configuration status on the controller, validating the AP profile, checking MACsec session summaries, verifying the status on the AP, and ensuring that the MKA session on the switch is secured with the correct key chain and cipher suite. It includes reviewing MACsec statistics and session security to confirm encrypted and validated packet transmission.

Verify the AP profile and ensure that MACsec is enabled and the correct key chain is applied.

```
Device# show ap profile name <ap-profile-name> detailed
 […]
Macsec :
  Enabled                       : True
  Replay protection window size: 0
```

Verify the MACsec summary.

```
Device# show ap macsec summary
AP Name                          AP Mac                          Capable
   Port 0          Port 1
-------------------------------------------------------------------------------
AP11AA.22BB.33CC                 99ff.88ee.77dd                  Yes
SUCCESS        UNKNOWN
```

Verify MACsec configuration on AP

Learn how to verify the macsec configuration and operational status on an Access Point to ensure secure communication.

Verify and troubleshoot MACsec configuration and operational status on an AP.

Ensure secure communication by confirming MACsec encryption is enabled, the correct cipher suite and key chain are in use, and the AP authentication and physical layer status are properly functioning.

Verify MACsec status.

> **Note**
>
> The output of this command varies according to the AP model.

```
Device# show macsec status

------------------------
wired0: Phy Address 16
------------------------
MACsec: Enabled
Capabilities:
        Max. Egress SecY: 32
        Egress FlowIDTcam Table Size: 32
        Egress SecyPolicy Table Size: 32
        Egress SaPolicy Table Size: 64
        Egress SecyToSaMap Table Size: 32
        Ciphers supported: GCM-AES-128
                           GCM-AES-256
        Max. Ingress SecY: 32
        Ingress FlowIDTcam Table Size: 32
        Ingress SecyPolicy Table Size: 32
        Ingress ScCamLookupKey Table Size: 32
        Ingress AnPerSc: 4
        Ingress ScAnToSaMap Table Size: 128
        Ingress SaPolicy Table Size: 64
Port Configuration:
        MACsec Port Count: 1
        MACsec Ingress pnThreshold: 0xffffffff
        MACsec Egress pnThreshold: 0xffffffff
SecY Configuration:
        Egress Controlled Port Enable: True
        Egress Protect Frames: True
        Egress Cipher: GCM-AES-256
        Ingress Replay Protect: False
        Ingress Replay Window: 3
        Ingress Validate Frames: Strict
        Ingress Cipher: GCM-AES-256
```

```
SC Configuration:
        Ingress SecY: 0, SCI: 0xecce13c791130002, enable: 1
```

Verify MACsec statistics.

```
Device# show macsec statistics
-----------------------
wired0: Phy Address 16
-----------------------
Egress SecY:
        Ifoutcommonoctets: 247290
        Ifoutunctloctets: 19856
        Ifoutctloctets: 227434
        Ifoutunctlucpkts: 0
        Ifoutunctlmcpkts: 136
        Ifoutunctlbcpkts: 0
        Ifoutctlucpkts: 378
        Ifoutctlmcpkts: 18
        Ifoutctlbcpkts: 0
        Outpktssecyuntagged: 0
        Outpktssecytoolong: 0
        Outpktssecynoactivesa: 0
        Outpktsctrlportdisabled: 0
Egress SC:
        Outoctetsscprotected: 0
        Outoctetsscencrypted: 222682
Egress SA:
        Outoctetssaprotected: 0
        Outoctetssaencrypted: 396
Egress Port:
        Outpktsflowidtcammiss: 0
        Outpktsparseerr: 0
        Outpktssectaginsertionerr: 0
        Outpktsearlypreempterr: 0
Egress Flow:
        Outpktsflowidtcamhit: 532
-----------------------
Ingress SecY:
        Ifinunctloctets: 176021
        Ifinctloctets: 130341
        Ifinunctlucpkts: 133
        Ifinunctlmcpkts: 352
        Ifinunctlbcpkts: 441
        Ifinctlucpkts: 133
        Ifinctlmcpkts: 216
        Ifinctlbcpkts: 441
        Inpktssecyuntaggedornotag: 136
        Inpktssecybadtag: 0
        Inpktssecyctl: 136
        Inpktssecytaggedctl: 0
        Inpktssecyunknownsci: 0
        Inpktssecynosci: 0
        Inpktsctrlportdisabled: 0
Ingress SC:
        Inoctetsscvalidate: 0
        Inoctetsscdecrypted: 120861
        Inpktsscunchecked: 0
        Inpktssclateordelayed: 0
        Inpktssccamhit: 790
Ingress SA:
        Inpktssaok: 790
        Inpktssainvalid: 0
```

```
        Inpktssanotvalid: 0
        Inpktssaunusedsa: 0
        Inpktssanotusingsaerror: 0
Ingress Port:
        Inpktsflowidtcammiss: 0
        Inpktsparseerr: 0
        Inpktsearlypreempterr: 0
Ingress Flow:
        Inpktsflowidtcamhit: 926
```

Debug the MACsec summary.

```
debug macsec phy
```

Auto MACsec for wireless APs

Introduces Auto MACsec for wireless APs, detailing peer detection via LLDP, 802.1X authentication, encrypted link establishment, default settings, onboarding processes, compatibility requirements, and instructions for disabling the feature in AP join profiles.

Auto MACsec is a link-security capability that

- detects a compatible peer through LLDP
- authenticates the AP through 802.1X using a hardware-backed identity, and
- establishes hop-by-hop MACsec protection before the AP joins the wireless controller.

Auto MACsec reduces the manual configuration required to protect the wired uplink between a wireless AP and its directly connected access switch. The encrypted link helps protect traffic on the last network hop from interception and man-in-the-middle attacks.

Auto MACsec peer detection functionality is supported on Wi-Fi 7 APs equipped with MACsec-capable physical layer (PHY) interfaces.

> **Note**
>
> Cisco IOS XE 26.2 uses LLDP for Auto MACsec peer detection.

During initial Auto MACsec onboarding, the access point can use its hardware SUDI when no locally significant certificate (LSC) or other port-authentication configuration is available.

This feature supports Day 0 APs and APs that do not have an 802.1X username or MACsec preshared key (PSK) configured.

Auto MACsec is enabled by default. If an AP does not have an 802.1X username or MACsec PSK configuration, it automatically attempts to establish an Auto MACsec connection.

You can disable Auto MACsec in the AP join profile. If the AP currently uses an Auto MACsec connection, disabling the feature immediately interrupts the connection and causes the AP to rejoin. If the AP cannot recover, verify the switch configuration and remove the corresponding Auto MACsec configuration from the switch.

If an AP has an 802.1X username or MACsec PSK configured, it uses that configuration instead of Auto MACsec, even when Auto MACsec is enabled.

Benefits of auto MACsec

Outlines how auto MACsec improves network security and simplifies deployment through automated discovery, authentication, and configuration.

Automatic MACsec provides these benefits:

- Encrypts traffic between a supported access point and access switch.
- Mitigates packet snooping and man-in-the-middle attacks on the last network hop.
- Discovers Auto MACsec-capable access points automatically through LLDP.
- Applies the required switch access-session configuration automatically.
- Uses HW SUDI to authenticate a new access point without requiring a previously provisioned access-point identity.
- Reduces manual MACsec configuration on individual access-point-facing ports.
- Supports access-port deployments and customized trunk-port deployments.
- Rediscovers and resecures validated access-point links after a switch reload.

Enable auto MACsec (GUI)

Configure an access point join profile to enable automatic MACsec link encryption and secure communication between AP and connected switch.

Enable auto MACsec to ensure secure wired communication between access points and their connected switches using MACsec encryption.

Auto MACsec simplifies the process of enabling secure link encryption for access points. By following these instructions, you configure an AP join profile to automatically enable MACsec, protecting data on the wired interface and preventing replay attacks.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. Click **Add** to create an AP join profile or edit the AP Join Profile assigned to your AP.
3. In the AP join profile page, click the **AP** tab.
4. In the **General** tab, under the **AP MACsec Configuration** section, complete these steps:
   a) **MACsec**: Enable or disable MACsec on the AP wired interface. By default, MACsec is disabled. When enabled, communication between the AP and the connected switch is secured using MACsec encryption.
   b) **Replay Protection Window Size**: Enter a value to configure the security parameter that defines how many sequence numbers (packets) can arrive out of order and still be accepted before the device starts treating them as replay attacks and drops them. The range is from 0 to 4294967295.
   c) **Pre-Shared Key Chain**: Select a **Pre-Shared Key Chain**. Choose the key chain configured earlier in the **Add Key Chain** window.
   d) **Auto MACsec**: Indicates whether automatic MACsec link encryption is enabled on the AP. By default, **Auto MACsec** is enabled.
5. Click **Update & Apply to Device**.

Auto MACsec is successfully enabled on the access point, securing wired communication with the connected switch using MACsec encryption.

Enable auto MACsec in AP join profile (CLI)

Configure auto MACsec on an AP wired interface within an AP join profile using command-line interface (CLI) commands.

Secure AP wired interfaces by enabling automatic MACsec within the AP join profile.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device (config)# ap profile profile-name
   ```

3. Enable auto MACsec on the AP wired interface.

   Example

   ```
   Device (config-ap-profile)# [no] auto secure macsec
   ```

   Use the **no** form of this command to disable the feature.

The AP join profile is configured to automatically enable MACsec encryption on the wired interface of access points, enhancing network security.

Verify auto MACsec

Lists the commands for verifying auto MACsec operation and provides key criteria for assessment.

Provide clear criteria and methods for verifying the proper operation and security status of auto MACsec on network devices, helping users assess whether auto MACsec is correctly configured and functioning.

To view the MACsec configuration summary for all APs, use this command:

```
Device# show ap macsec summary
Capability Codes:
(M) MACSec, (A) Auto-MACSec

AP Name                          AP Mac                          Capability
  Auto        Port 0          Port 1
-------------------------------------------------------------------------------
AP6849.926F.A870         10a8.29f3.d9a0                              M
  Disabled   NA                NA
AP780F.819F.2140          2c65.8db6.e0e0                             M,A
   Disabled   SUCCESS       UNKNOWN
APF01D.2DB7.4138         84f1.476e.0b40                              -
   Disabled    NA                NA
```

To view the common information for a particular Cisco AP, use this command:

```
Device# show ap name Cisco-AP config general
Macsec Capability                                : Capable
  Auto MACsec                                    : Disabled
  Port 0
    Macsec Status                                : SUCCESS
    Macsec Ciphersuite                           : GCM_AES_128
  Port 1
    Macsec Status                                : UNKNOWN
    Macsec Ciphersuite                           : GCM_AES_128
```

Recovery for Auto MACsec

Outlines the recovery procedure for Auto MACsec configuration mismatches, guiding you through verification of switch and AP settings, identification of handshake failures, temporary disabling of MACsec, restoring connectivity, and re-enabling MACsec to ensure successful AP controller joins.

Summary

The recovery process for Auto MACsec addresses issues where an Access Point (AP) fails to join a controller or obtain an IP address due to Auto-MACsec configuration mismatches. The process involves verifying switch and AP configurations, identifying handshake or MACsec failures, temporarily disabling Auto-MACsec on the switch interface to allow the AP to reboot and connect without MACsec security, and then restoring the MACsec configuration after confirming connectivity and controller settings.

Workflow

1. Verify switch-side configuration: Run these commands on the switch to check interface status and MACsec sessions.
   - `show interface <interface_id> status`
   - `show macsec interface <interface_id>`
   - `show run interface <interface_id>`
   - `show derived-config interface <interface_id>`
2. Verify AP-side status: On the Access Point, check MACsec and authentication status.
   - `show macsec status`
   - `show ap authentication status details`
3. Look for EAP handshake failures or MACsec failures in the outputs.
4. Trigger recovery mode: Temporarily disable Auto MACsec on the switch interface by executing:

   ```
   shutdown
   no access-session interface-template sticky
   access-session inherit disable interface-template-sticky
   no auto detect macsec
   no shutdown
   ```

   This allows the AP to reboot, obtain an IP address, and join the Wireless LAN Controller without MACsec security.
5. Restore connectivity: Verify controller management settings with `show ip http secure-server status` command.

   Ensure secure access policies are not blocking discovery.
6. Reapply MACsec configuration:

- Confirm the configuration template and MKA parameters are correct on the switch.
- Re-enable Auto MACsec on the switch port.

Result

This workflow ensures recovery from Auto-MACsec configuration mismatches by temporarily disabling MACsec to restore network connectivity and then re-enabling it after verification.

157 FIPS

Topics:

- Federal Information Processing Standard (FIPS)
- Common criteria
- TLS signature algorithm enhancements for NDcPP v3.0e
- HTTPS server security considerations with NDcPP 4.0

This chapter explains enabling and verifying federal cryptographic compliance mode on controllers, including self-test behavior, restrictions (SSH, TrustSec, PAC keys), HA setup with matching authorization keys, CLI commands, and troubleshooting notes.

Federal Information Processing Standard (FIPS)

Introduces FIPS, outlining its guidelines and restrictions, providing instructions for self-tests and configuration, explaining FIPS operation in HA setups, and describing procedures for monitoring and verifying FIPS compliance.

A Federal Information Processing Standard (FIPS) is a security standard that

- defines requirements for cryptographic modules intended to secure sensitive but unclassified (SBU) information
- is mandated for use by U.S. government agencies and adopted by many regulated industries such as finance and healthcare, and
- establishes assurance levels to validate the strength and reliability of cryptographic implementations.

Sensitive but unclassified (SBU) information: Data that, while not classified for national security, still requires protection due to its sensitive nature and potential impact if disclosed.

Additional reference information
Federal Information Processing Standard (FIPS) 140-2 is a well-known FIPS standard specifying security requirements for cryptographic modules protecting SBU information. These cryptographic modules are produced by the private sector for use in government and regulated environments.

With FIPS in the enabled state, some passwords and pre-shared keys must have the following minimum lengths:

- For Software-Defined Access Wireless, between the controller and map server, a pre-shared key (such as the LISP authentication key) used for authenticating TCP messages must be at least 14 characters long.
- The ISAKMP key (for example, the Crypto ISAKMP key) must also be at least 14 characters long.

Guidelines and restrictions for FIPS

Guidelines and restrictions for FIPS.

- Controller switches use a legacy key to support legacy APs. In FIPS mode, the crypto engine identifies the key as weak and displays this error message: **% Error in generating keys: could not generate test signature.** Ignore error messages during controller startup in FIPS mode.
- When you enable FIPS, SSH clients that use SHA1 cannot access the controller.
- When you configure a WLAN, set the PSK to at least 15 characters. Otherwise, APs cannot join the controller after you change tags.
- You need to use FIPS-compliant SSH clients to access the controller.
- You cannot use TrustSec in FIPS mode.
- You cannot configure PAC keys in FIPS mode.
- The console of APs get disabled when the controller is operating in FIPS mode.
- The weak or legacy cipher like SHA1 is not supported in FIPS mode.
- APs would not reload immediately, if you change the FIPS status.
- We recommend a minimum RSA key size of 2048 bits under RADSEC when operating in FIPS mode. Otherwise, the RADSEC fails.

FIPS self-tests

This concept explains FIPS self-tests, which are mechanisms performed by cryptographic modules to confirm integrity and operational soundness during startup and throughout device operation by running power-up and conditional checks as required by FIPS 140-2 standards.

A FIPS self-test is a cryptographic module validation mechanism that

- verifies functionality and integrity during device power-up
- performs conditional checks whenever security functions are invoked, and
- ensures FIPS 140-2 compliance by enforcing automated self-testing procedures.

Power-up self-tests

Power-up self-tests run automatically after the device powers up. A device enters FIPS mode only after all self-tests are successfully completed. If any self-test fails, the device logs a system message and transitions into an error state. If the power-up self-test fails, the device also fails to boot.

The module uses a known-answer test (KAT), in which a cryptographic algorithm runs on data for which the correct output is already known, and then the calculated output is compared to this expected value. If the calculated output does not equal the known answer, the known-answer test fails.

Power-up self-tests include:

- Software integrity: Validates the integrity of module software as it loads.
- Algorithm tests: Verifies correct operation of cryptographic algorithms using KATs.

Conditional self-tests

Conditional self-tests are performed each time an applicable security function or operation is accessed, unlike power-up self-tests that run only at startup.

The device uses a cryptographic algorithm known-answer test (KAT) on each FIPS 140-2-approved function, which includes encryption, decryption, authentication, and random number generation. The algorithm is applied to a set of data with a known correct output, and the calculated output is compared to this known value. If the calculated output is different, the KAT fails.

Conditional self-tests run automatically whenever their associated security function or operation is invoked. They include the following:

- Pair-wise consistency test: Runs when a public or private key-pair is generated to confirm the correct relationship.
- Continuous random number generator test: Runs each time a random number is generated to check the quality of randomness.
- Bypass: Evaluates bypass operations if performed.
- Software load: Validates integrity when new software is loaded.

Examples

- When a device boots, it performs power-up self-tests to confirm its cryptographic algorithms work as intended before entering FIPS mode.
- Whenever the device generates a new cryptographic key pair, a pair-wise consistency test is executed.
- If a random number is generated, the system runs a continuous random number generator test, verifying the output for compliance.

Configure FIPS (CLI)

Use this procedure to enable or disable FIPS mode on both active and standby controllers to comply with federal standards.

Enable or disable FIPS mode on controllers for compliance and security.

Ensure that both the active and standby controllers have the same FIPS authorization key.

> **Note**
>
> Changing the FIPS status removes AES-encrypted password configurations from the controller. After enabling or disabling FIPS mode and rebooting the controller, reconfigure all affected password-based settings, including the AP management credentials.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device
   # configure terminal
   ```

2. Enable FIPS mode.

   Example

   ```
   Device
   (config)#
   fips authorization-key key
   ```

   The key length must be 32 hexadecimal characters.

   > **Note**
   >
   > When FIPS is enabled, you may need to trigger more than one factory reset using the reset button.

   To disable FIPS mode on the device, use the **no** form of this command.

3. Return to privileged EXEC mode.

   Example

   ```
   Device
   ```

```
(config)#
end
```

Enable or disable FIPS mode. Reboot the controller to apply changes. After the controller reboots, access points also reboot once they rejoin.

What to do next

Reboot the controller each time you enable or disable FIPS mode.

How FIPS configuration works in high availability setups

This process describes how to configure both active and standby controllers with the same FIPS authorization key independently before forming a high availability (HA) pair. This prevents synchronization issues and reload loops.

Set up a high availability (HA) pair in FIPS mode by configuring both controllers with the same FIPS authorization key before forming the HA pair.

If you configure the key after forming the HA pair, it will not sync to the standby. This may cause reload loops when you reboot.

- Break the HA pair if you configured the FIPS key after pairing.
- Configure the same FIPS authorization key on both members independently.
- Form the HA pair again.

Summary

In this process, you configure FIPS in an HA setup by setting the same FIPS authorization key on both members before forming the HA pair.

- You must perform the configuration steps on both controllers.
- Configure and pair both controllers (member1 and member2) in HA mode.

Matching FIPS authorization keys on both controllers in the HA pair prevent synchronization issues and reload loops.

Workflow

These stages describe the steps to configure FIPS in an HA setup.

1. Turn off both members of the stack.
2. Power on only member1 and wait for the controller to start. Log in from the console when prompted.
3. Log in to member1 with valid credentials and use the following commands to verify FIPS status and configuration:
   - Use the **show fips status** command
   - Use the **show fips authorization-key** command
   - Use the **show romvar** command
   - Use the **show chassis** command

> **Note**
>
> Keep the configured FIPS authorization key available.

4. Configure the FIPS key on member1 if you have not already configured it.
   - conf t
   - fips authorization-key <thirty-two hexadecimal characters>
5. Save your changes and power off member1.
6. Power on only member2 and wait for the controller to start. Log in from the console when prompted.
7. Log in to member2 with valid credentials and use the following commands to verify FIPS status and configuration:
   - Use the **show fips status** command
   - Use the **show fips authorization-key** command
   - Use the **show romvar** command
   - Use the **show chassis** command

> **Note**
>
> Keep the configured FIPS authorization key available.

8. Configure the FIPS key on member2 if you have not already configured it.

> **Note**
>
> The key value must be the same on both members of the stack.

   - conf t
   - fips authorization-key <thirty-two hexadecimal characters>
9. Save your changes and power off member2.
10. Power on both members together and wait for the stack to form.
11. Monitor the system for any crash or unexpected reload events.

> **Note**
>
> The members are not expected to reload because of FIPS issues.

Result

Both controllers in the HA pair operate in synchronized FIPS mode. Correct configuration prevents synchronization issues and reload loops caused by mismatched or improperly applied FIPS authorization keys.

Verify FIPS configuration

Verify FIPS configuration.

You can verify FIPS configuration using these commands:

Use this **show** command to display the installed authorization key:

```
Device# show fips authorization-key
FIPS: Stored key (16) : 12345678901234567890123456789012
```

Use this **show** command to display the status of FIPS on the device:

```
Device# show fips status
Chassis is running in fips mode
```

Common criteria

Details Common Criteria, offering configuration procedures, verification steps, and guidelines for operating in CC mode to achieve and maintain relevant security certification requirements.

A Common Criteria certificate is a security assurance certification that

- provides independent verification that IT products meet specified security requirements,
- twenty-four countries worldwide officially recognize this certification,
- and this scheme uses defined protection profiles for requirements, tests, and evaluation methodologies.

Common Criteria protection profiles and further information

Common Criteria (CC) is a global testing standard. It verifies whether a device provides the security functionalities claimed by product developers. The standard establishes requirements, tests, and an evaluation methodology to ensure that IT products comply with recognized security standards.

The Target of Evaluation (ToE) is the product or system under evaluation. It must conform to protection profiles under the Common Criteria scheme. The applicable protection profiles are:

- Collaborative Protection Profile for Network Devices (NDcPP) version 2 dated 2017-05-05,
- Wireless Local Area Network (WLAN) Access Systems Extended Package version 1 dated 2015-05-29.

The CC certificate's recognition by twenty-four countries ensures that vendors can demonstrate compliance with internationally accepted security requirements.

For more information about CC, refer to the Cisco Common Criteria documentation.

Configure the common criteria (CLI)

Use this procedure to enable common criteria mode on a controller, including configuring DTLS cipher suites and versions required for regulatory compliance.

Enable common criteria mode for enhanced security compliance on your wireless controller.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the common criteria mode for the controller.

   Example

   ```
   Device(config)# wireless wlancc
   ```

   > **Note**
   >
   > Reboot the controller after enabling the common criteria mode.

3. Configure the cipher suite supported by DTLS.

   Example

   ```
   Device(config)# ap dtls-cipher ciphersuite
   ```

   > **Note**
   >
   > Reboot the controller to activate the selected cipher suite.

4. Configure DTLS version 1.0 or 1.2.

   Example

   ```
   Device(config)# ap dtls-version dtls_1_0 dtls_1_2
   ```

> **Note**
>
> Save the configuration and reload the controller for the changes to take effect.

5. Exit the configuration mode and enter the privileged EXEC mode.

Example

```
Device(config)# end
```

The controller operates in common criteria mode with the chosen DTLS cipher suite and protocol version. The changes take effect after you reboot the controller.

Verify CC configuration

Verify CC configuration.

Use the this **show** command to display the wireless certification configurations:

```
Device# show wireless certification config
Wireless Certification Configurations

WLANCC                                    : Configured
AP DTLS Cipher Suite                      : DHE-RSA-AES128-SHA
                                            DHE-RSA-AES256-SHA
                                            DHE-RSA-AES256-SHA256
                                            ECDHE-ECDSA-AES128-GCM-SHA256
                                            ECDHE-ECDSA-AES256-GCM-SHA384
AP DTLS Version                           : DTLS v1.2
```

Check points for CC mode operation

Check points for CC mode operation.

This topic describes the key check points and considerations for CC mode operation on Cisco Catalyst 9800 Series Wireless Controllers.

Check points for CC mode operation

You need to be aware of these for CC mode operation:

Table 139: Check points for CC mode operation

Features	Description
Link encryption	Data link encryption is not supported for C91xx wireless mobility express platform.
Link encryption	For non-C91xx wireless mobility express platfoms - Enabling Data link encryption (using ECDHE keypair) would make AP flap continually.

Features	Description
LDAP	Secure LDAP does not support strong ciphers and is not part of CC certification.
Mobility	Mobility between Cisco Catalyst 9800 Series Wireless controllers is possible with LSC as wireless management trustpoint (having RSA based keys).
Mobility	Mobility between AireOS WLC and Cisco Catalyst 9800 Series Wireless controllers is supported (using SUDI and MIC certificates for wireless management trustpoint).
Mobility	Mobility between AireOS WLC and Cisco Catalyst 9800 Series Wireless controllers is not supported (if using LSC certficates for wireless management trustpoint).
CC mode	The **show wireless certification config** command displays the configured values for WLANCC, or AP-dtls-ciphersuite, or AP-dtls-version, and needs reload after re-configuring these parameters.
CC mode	The AES128-SHA option is not supported for AP-dtls-ciphersuite when Cisco Catalyst 9800 Series Wireless Controller is operating in CC mode.
CC mode	The AES128-SHA option is supported for AP-dtls-ciphersuite when Cisco Catalyst 9800 Series Wireless Controller is operating in FIPS mode.
CC mode	If you want your Cisco Catalyst 9800 Series Wireless Controller to operate in CC mode (you need to enable both FIPS mode and CC mode).
LSC	To secure communication between Cisco Catalyst 9800 Series Wireless Controller and LSC server, you need to deploy ESTCA as LSC server (which uses TLS to secure related communication).
LSC	Cisco Catalyst 9800 Series Wireless Controllers do not support HTTPS to secure its communication with the LSC server.
LSC	During LSC provisioning, APs generate EC based keys only when related Cisco Catalyst 9800 Series Wireless Controller is operating in CC mode.
LSC	During LSC provisioning, APs generate RSA based keys when related Cisco Catalyst 9800 Series Wireless Controller is operating in FIPS mode.
LSC	During LSC provisioning, APs generate RSA based keys when related Cisco Catalyst 9800 Series Wireless Controlle is operating in non-FIPS or non-CC mode.

Features	Description
Password Obfuscation	You can use the following commands for password obfuscation: • **key config-key password-encrypt** • **service password-encryption** • **password encryption aes** • **passwd key obfuscate**
CC mode	APs reload immediately, if you change the **wlancc** status.
FIPS mode	APs do not reload immediately, if you change the FIPS status.
Cisco 1562 AP	To assist Cisco 1562 APs join the Cisco Catalyst 9800 Series wireless controller, you need to have the ethernet MAC of the AP in the username list.
AP serial number authorization	Serial number authorization is possible only when Cisco Catalyst 9800 Series wireless controller is in FIPS and CC mode, and with LSC based trustpoints/certficates only (not with SUDI trustpoint).
Display	FIPS suitability displays **Suitable only** if the controller is in CC mode and LSC certificate is compatible. Both wireless management and Certs CN should match the hostname of the controller and length of RSA Key (> 2048) (or) EC keys being used.
RADSEC	RSA key size must contain a minimum of 2048 bits (of certificate under RADSEC) when operating in FIPS or CC mode, else RADSEC fails.

TLS signature algorithm enhancements for NDcPP v3.0e

Outlines TLS signature algorithm enhancements for NDcPP v3.0e, including compliance in FIPS CC mode, removal of outdated algorithms and ciphersuites, and strengthened protection against vulnerabilities for both TLS clients and servers.

The TLS signature algorithm enhancements in FIPS Common Criteria (CC) mode is a security posture that

- ensures compliance with the NDcPP v3.0e standard
- removes outdated and less secure signature algorithms and ciphersuites, and
- protects against known vulnerabilities.

Feature history

Table 140: Feature history for TLS signature algorithm enhancements for NDcPP v3.0e

Feature Name	Release Information	Feature Description
TLS signature algorithm enhancements for NDcPP v3.0e	Cisco IOS XE 17.18.2	This feature ensures that Cisco products operating in FIPS Common Criteria (CC) mode are fully compliant with the NDcPP v3.0e security standard. It achieves this by: • Removing outdated and unapproved algorithms • Eliminating weaker algorithms thereby ensuring robust cryptographic protection for sensitive communications. • Enabling both TLS clients (listing supported signature algorithms in the Client Hello message) and TLS/DTLS servers (listing signature algorithms in the Certificate Request message) to operate within the strict requirements of NDcPP v3.0e.

Common criteria certifications using NDcPP v3.0e mandate that TLS Signature Algorithms that are outdated or unapproved are removed from being offered in TLS 1.2 and TLS 1.3, DTLS 1.2. For TLS clients, the signature algorithms are listed in the **Client Hello** message whereas for TLS/DTLS server, the signature algorithms are listed in the **Certificate Request** message.

As security standards evolve and threats become more sophisticated, it is crucial to update and remove outdated and less secure signature algorithms and ciphersuites. This enhancement prevents potential non-compliance and security risks, ensuring robust and compliant network operations. This requirement is vital for maintaining secure communications in environments adhering to the NDcPP v3.0e standard.

Signature algorithms and ciphersuites in TLS and DTLS applications

Allowed signature algorithms

In FIPS Common Criteria mode, you can use the following signature algorithms with TLS and DTLS:

- ecdsa_secp256r1_sha256 (0x0403)
- ecdsa_secp384r1_sha384 (0x0503)
- ecdsa_secp521r1_sha512 (0x0603)
- rsa_pss_pss_sha256 (0x0809)
- rsa_pss_pss_sha384 (0x080a)
- rsa_pss_pss_sha512 (0x080b)
- rsa_pss_rsae_sha256 (0x0804)
- rsa_pss_rsae_sha384 (0x0805)

- rsa_pss_rsae_sha512 (0x0806)
- rsa_pkcs1_sha256 (0x0401)
- rsa_pkcs1_sha384 (0x0501)
- rsa_pkcs1_sha512 (0x0601)

> **Note**
>
> No signature algorithms other than those listed are allowed in CC mode.

These sections provide a brief description of the allowed and deprecated signature algorithms and ciphersuites for EST TLS client, RadSec, Syslog, and HTTP Secure (TLS) Server in IOS-XE under FIPS Common Criteria mode:

EST TLS client

Allowed signature algorithms: The EST TLS client offers only allowed signature Algorithms in CC mode, where the IOS-XE device has a TLS client connection to an EST capable CA-server.

Ciphersuites: This is the list of plain RSA ciphers supported from the HTTP side. As part of FIPS Common Criteria mode these ciphersuites are not offered from default client 'Hello' message.

- AES128-SHA
- AES256-SHA
- AES128-SHA256
- AES256-SHA256
- AES128-GCM-SHA256
- AES256-GCM-SHA384

> **Note**
>
> - Because the changes are in HTTP client infrastructure, all HTTP client applications in FIPS Common Criteria mode are affected.
> - If the peer supports a wide range of ciphersuites, applications are not affected.
> - If the peer supports only the listed ciphers, the TLS handshake fails in FIPS+CC mode.

RadSec

For RadSec TLS and DTLS, these are the ciphers that are supported and deprecated with FIPS common criteria mode.

Supported cipher list when FIPS + CC mode is enabled:

- **DTLS (IOS and BINOS)**
  - ECDHE-ECDSA-AES128-GCM-SHA256 and ECDHE-ECDSA-AES256-GCM-SHA384
  - ECDHE-ECDSA-AES128-SHA256 and ECDHE-ECDSA-AES256-SHA384

- ECDHE-RSA-AES128-GCM-SHA256 and ECDHE-RSA-AES256-GCM-SHA384
- ECDHE-RSA-AES128-SHA256 and ECDHE-RSA-AES256-SHA384

- **TLS (BINOS)**
  - ECDHE-ECDSA-AES128-GCM-SHA256 and ECDHE-ECDSA-AES256-GCM-SHA384
  - ECDHE-RSA-AES128-GCM-SHA256 and ECDHE-RSA-AES256-GCM-SHA384

- **TLS (IOS)**
  - ECDHE-ECDSA-AES128-GCM-SHA256
  - ECDHE-RSA-AES128-GCM-SHA256

Deprecated Cipher list when FIPS + CC mode is enabled:

- **DTLS (IOS and BINOS)**:
  - AES128-SHA256
  - AES256-SHA256

- **TLS (BINOS)**:
  - RC4-MD5
  - NULL-SHA
  - AES128-SHA

- **TLS (IOS)**:
  - AES128-SHA

> **Note**
>
> - When FIPS mode is not enabled, the current existing set of ciphers is supported, which is a combination of both strong and weak ciphers.
> - RadSec TLS connections allow only TLS signature algorithms listed in the 'Allowed signature algorithms' section.

Syslog

These are the list of ciphersuites from the Syslog component that are deprecated for FIPS Common Criteria mode.

- AES128-SHA
- AES256-SHA
- AES128-SHA256
- AES256-SHA256
- AES128-GCM-SHA256
- AES256-GCM-SHA384

> **Note**
>
> These ciphersuites are excluded in the Syslog default ciphersuite and can be configured through commands.

HTTP secure (TLS) server

Signature algorithm:

When client authentication is enabled on the IOS-XE device, the HTTP secure (TLS) server includes the list of signature algorithms in the Certificate Request Message.

Ciphersuites:

The HTTP side supports these plain RSA ciphers. In FIPS Common Criteria mode, these ciphersuites are not included in the default client 'Hello'.

- AES128-SHA
- AES256-SHA
- AES128-SHA256
- AES256-SHA256
- AES128-GCM-SHA256
- AES256-GCM-SHA384

Configure syslog TLS profile ciphersuites (CLI)

Configure syslog TLS profile ciphersuites (CLI)

To specify the allowed ciphersuites for secure Syslog communication in FIPS Common Criteria mode.

In FIPS Common Criteria mode, certain weaker ciphersuites are deprecated and removed from the default list for Syslog. You can view the available ciphersuites and configure them.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Access the TLS profile for syslog. Replace `tls-profile-name` with your profile name.

   Example

   ```
   Device(config)# logging tls-profile tls-profile-name
   ```

3. Select and configure a ciphersuite.

Example

```
Device(config-tls-profile)# ciphersuite ecdhe-rsa-aes-cbc-sha2
```

You can view the ciphersuites by running the `Device(config-tls-profile)# ciphersuite ?` command.

The syslog TLS profile is configured with the specified ciphersuite.

Configure HTTP secure ciphersuites

To explicitly configure secure ciphersuites for HTTP client and server communications.

This feature ensures that HTTP client and server communications adhere to enhanced security standards by allowing explicit configuration of secure ciphersuites. There is no change in default ciphersuite behavior for HTTP client and server.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure secure ciphersuites for the HTTP server. You can specify one or more ciphersuites.

   Example

   ```
   Device(config)# ip http secure-ciphersuite ciphersuites
   ```

3. Configure secure ciphersuites for the HTTP client.

   Example

   ```
   Device(config)# ip http client secure-ciphersuite ciphersuites
   ```

HTTP client and server communications use the explicitly configured secure ciphersuites.

Verify Syslog TLS profile

To verify the configured Syslog TLS profile, use this command.

```
Device# show running-config | section logging tls-profile syslog_1
logging tls-profile syslog_1
ciphersuite ecdhe-ecdsa-aes-gcm-sha2
```

Verify HTTP client and server applications

To verify the configured HTTP secure ciphersuites for the server, use this command.

```
Device# show running-config | include ip http secure-ciphersuite
ip http secure-ciphersuite ecdhe-ecdsa-aes-gcm-sha2 tls13-aes256-gcm-sha384
```

To verify the configured HTTP secure ciphersuites for the client, use this command.

```
Device# show running-config | include ip http client secure-ciphersuite
ip http client secure-ciphersuite ecdhe-rsa-aes-gcm-sha2 tls13-aes128-gcm-sha256
```

To verify the configured HTTP secure server status information, use this command.

```
Device# show ip http server secure status
HTTP secure server status: Enabled
HTTP secure server port: 443
HTTP secure server ciphersuite:  dhe-aes-cbc-sha2 dhe-aes-gcm-sha2
       ecdhe-rsa-aes-cbc-sha2 ecdhe-rsa-aes-gcm-sha2 ecdhe-ecdsa-aes-gcm-sha2

        tls13-aes128-gcm-sha256 tls13-aes256-gcm-sha384
HTTP secure server TLS version:  TLSv1.2
HTTP secure server client authentication: Disabled
HTTP secure server PIV authentication: Disabled
HTTP secure server PIV authorization only: Disabled
HTTP secure server trustpoint: TP-self-signed-745339156
HTTP secure server peer validation trustpoint:
HTTP secure server ECDHE curve: secp256r1
HTTP secure server active session modules: ALL
```

To verify the configured HTTP secure client status information, use this command.

```
Device# show ip http client secure status
HTTP secure client ciphersuite:  dhe-aes-cbc-sha2 dhe-aes-gcm-sha2
       ecdhe-rsa-aes-cbc-sha2 ecdhe-rsa-aes-gcm-sha2 ecdhe-ecdsa-aes-gcm-sha2

        tls13-aes128-gcm-sha256 tls13-aes256-gcm-sha384
HTTP secure client TLS version:  TLSv1.3 TLSv1.2
HTTP secure client trustpoint:
```

HTTPS server security considerations with NDcPP 4.0

Outlines HTTPS server security configurations that enforce advanced cryptographic protocols, ensure NDcPP v4.0 compliance, and allow administrators to restrict access to modern TLS versions and secure cipher suites for enhanced data protection.

HTTPS server security configurations are security standards for the Cisco web-based management interface that

- enforce advanced cryptographic protocols for client-server communication
- align with NDcPP v4.0 certification requirements for enhanced data protection, and
- allow administrators to restrict usage to modern TLS versions and secure cipher suites.

These configurations replace legacy default settings by mandating the use of secure elliptic curves and high-strength encryption algorithms, thereby reducing the attack surface of the web management interface.

Feature history

Table 141: Feature history for HTTPS server security considerations with NDcPP 4.0

Feature Name	Release Information	Feature Description
HTTPS server security considerations with NDcPP 4.0	Cisco IOS XE 26.2.1	This feature hardens the Cisco web-based management interface by enforcing advanced cryptographic protocols for client-server communication, ensuring alignment with NDcPP v4.0 certification requirements for robust data protection while providing administrators with granular control to restrict interface access to modern TLS versions and secure cipher suites.

Best practices for HTTPS server security

Ensure that you maintain a secure management interface using these guidelines:

- Enforce TLS 1.3: Configure the HTTPS server to exclusively use TLS 1.3 to ensure the highest level of connection security.
- Maintain updated curves: Use the secp384r1 curve for ECDHE key exchange to align your device with current industry security standards
- Validate configurations: Verify your security policy settings immediately after any changes by executing the `show ip http server secure status` command to confirm that the active settings are correctly applied.

Configure HTTP server security settings (CLI)

Provides instructions for configuring HTTP server security settings, including specifying TLS versions, setting cipher suites, configuring elliptic curve parameters, and verifying security status to secure device web access.

Configure network device HTTP server security to ensure management access is encrypted and resilient to modern security threats.

Perform this task when securing web-based device access, or after a security audit recommends updating transport security standards.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the allowed TLS version.

   Example

   ```
   Device(config)# ip http tls-version TLSv1.3
   ```

3. Set the secure cipher suite.

   Example

   ```
   Device(config)# ip http secure-ciphersuite tls13-aes256-gcm-sha384
   ```

4. Configure the default elliptic curve for key exchange.

   Example

   ```
   Device(config)# ip http secure-ecdhe-curve secp384r1
   ```

5. Verify the current security status.

   Example

   ```
   Device# show ip http server secure status
   ```

Device web management interfaces now enforce strong encryption and current best-practice TLS configuration.

Configure HTTP client security settings (CLI)

Provides instructions for configuring HTTP client security settings, including specifying TLS versions, setting cipher suites, and verifying security status to secure device web access.

Configure network device HTTP client security to ensure management access is encrypted and resilient to modern security threats.

Perform this task when securing web-based device access, or after a security audit recommends updating transport security standards.

Procedure

---

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the allowed TLS version.

   Example

   ```
   Device(config)# ip http client tls-version TLSv1.3
   ```

3. Set the secure cipher suite.

   Example

   ```
   Device(config)# ip http client secure-ciphersuite tls13-aes256-gcm-sha384
   ```

4. Verify the current security status.

Example

```
Device# show ip http client secure status
```

Device web management interfaces now enforce strong encryption and current best-practice TLS configuration.

158 Internet Protocol Security

Topics:

- Internet Protocol Security (IPsec)

Guides users through configuring Internet Protocol Security protocols and settings to establish secure network communications and protect data transmission.

Internet Protocol Security (IPsec)

Introduces IPSec fundamentals covering IKEv1 and IKEv2 transform sets, configuration procedures for both protocol versions, X.509 certificate implementation, session interruption handling, and traffic verification methods with practical configuration examples.

Internet Protocol Security (IPsec) ia framework of open standards that

- ensures confidentiality by encrypting data packets before transmission
- guarantees data integrity by authenticating packets, and
- provides data origin authentication to verify the source

IPsec includes an anti-replay feature that helps you detect and reject replayed packets.

An IPsec tunnel is a secure communication channel connecting two IPsec peers. It encrypts selected network traffic, uses security associations (SAs) to determine protocols, algorithms, and keying material for traffic protection, and applies access lists and crypto map sets to define which packets are encrypted and routed through the tunnel.

A security association (SA) is an agreement between two IPsec peers that defines the security protocols and algorithms to apply to protected packets, specifies the keying material used, and operates unidirectionally for each protocol.

Key Features of IPsec

IPsec provides the following security features:

- Confidentiality: IPsec encrypts packets to protect sensitive information during transmission.
- A framework of open standards, IPsec verifies the source of the packets, enhancing security.
- Anti-replay: The IPsec receiver can detect and reject replayed packets.

The controller supports IPsec configuration, securing syslog traffic and enabling a robust network-wide security policy.

If an you wants to encrypt all traffic between two sites, you can:

- Create access lists specifying relevant IP address ranges
- Apply those access lists to a crypto map set, and
- Bind the crypto map set to the device interface facing the remote site.

All matching packets is encrypted and transmitted through the IPsec tunnel.

Think of access lists as the guest list at a secure building.

Crypto maps are the guard's special instructions for who receives extra screening.

Transform sets are the security checks both guard and guest agree on.

Only guests who pass the right checks get inside; others are turned away.

IKEv1 transform sets

This concept explains how IKEv1 transform sets define a combination of security protocols and algorithms used to protect data flows in IPsec VPNs.

An IKEv1 transform set is a cryptographic configuration set that

- specifies a combination of security protocols and algorithms
- enables peers to negotiate security parameters for IPsec Security Associations (SAs), and

- is assigned to specific data flows through crypto map entries.

Transform set behavior during IPsec SA negotiation

During IPsec SA negotiation:

- Both peers search for a transform set that matches on each side.
- If a matching transform set is found, it is applied to the relevant protected traffic as part of the IPsec SA.

Additional reference information

- Administrators can define multiple transform sets and assign one or more to crypto map entries.
- The transform set associated with a crypto map entry determines the security protocols and algorithms used to protect specified data flows.
- Changing a transform set during operation does not affect existing SAs.
  - The new configuration applies only to future SA negotiations.
  - To apply changes immediately, you can clear all or part of the SA database using the **clear crypto sa** command.

Example

The configuration snippet shows how to configure an IPsec IKEv1 transform set using AES-CBC-128 for payload encryption. To use AES-CBC-256, specify **encryption aes 256**.

```
Device # conf t
Device (config)#crypto isakmp policy 1
Device (config-isakmp)# hash sha
Device (config-isakmp)# encryption aes
```

Configure IPsec using IKEv1

Use this procedure to configure IPSec using Internet Key Exchange version 1.

Follow the procedure given below to configure IPsec IKEv1 to use AES-CBC-128 for payload encryption.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. Define an Internet Key Exchange (IKE) policy and assign a priority to the policy.

   Example

   ```
   Device(config)# crypto isakmp policy priority
   ```

- *priority*: Uniquely identifies the IKE policy and assigns a priority to the policy. Valid values: 1 to 10,000; 1 is the highest priority.

3. Specify the hash algorithm.

   Example

   ```
   Device(config-isakmp)# hash sha
   ```

   Specifies the hash algorithm.

4. Configure IPsec IKEv1 to use AES-CBC-128 for payload encryption.

   Example

   ```
   Device(config-isakmp)# encryption aes
   ```

   AES-CBC-256 can be selected with 'encryption aes 256'.

   > **Note**
   >
   > The authorized administrator must ensure that the keysize for this setting is greater than or equal to the keysize selected for ESP in section IPsec Transforms and Lifetimes. If AES 128 is selected here, then the highest keysize that can be selected on the device for ESP is AES 128 (either CBC or GCM).
   >
   > Both confidentiality and integrity are configured with the **hash sha** and **encryption aes** commands respectively. As a result, confidentiality-only mode is disabled.

5. Configure IPsec to use the specified preshared keys as the authentication method.

   Example

   ```
   Device(config-isakmp)# authentication pre-share
   ```

   Preshared keys require that you separately configure these preshared keys.

6. Exit config-isakmp configuration mode.

   Example

   ```
   Device(config-isakmp)# exit
   ```

   Exits config-isakmp configuration mode.

7. Configure a preshared authentication key.

   Example

   ```
   Device(config)# crypto isakmp key keystring address peer-address
   ```

> **Note**
>
> To ensure a secure configuration, we recommend that you enter the pre-shared keys with at least 22 characters in length and can be composed of any combination of upper and lower case letters, numbers, and special characters (that include: "!", "@", "#", "$", "%", "^", "&", "*", "(", and ")").
>
> The device supports pre-shared keys up to 127 characters in length. While longer keys increase the difficulty of brute-force attacks, longer keys increase processing time.

8. Specify the Diffie-Hellman (DH) group identifier as 2048-bit DH group 14 and selects DH Group 14 (2048-bit MODP) for IKE.

   Example

   ```
   Device(config-isakmp)# group 14
   ```

   However, 19 (256-bit Random ECP), 24 (2048-bit MODP with 256-bit POS), 20 (384-bit Random ECP), 15 (3072 bit MODP), and 16 (4096-bit MODP) are also allowed and supported.

9. Specify the lifetime of the IKE SA.

   Example

   ```
   Device(config-isakmp)# lifetime seconds
   ```

   The default time value for Phase 1 SAs is 24 hours (86400 seconds), but this setting can be changed using the command above with different values.

   - *seconds*: Time, in seconds, before each SA expires. Valid values: 60 to 86,400; default value: 86,400.

   > **Note**
   >
   > The shorter the lifetime (up to a point), the more secure your IKE negotiations will be. However, with longer lifetimes, future IPsec SAs can be set up more quickly.

10. Ensure all IKEv1 Phase 1 exchanges will be handled in the default main mode.

    Example

    ```
    Device(config-isakmp)# crypto isakmp aggressive-mode disable
    ```

11. Exit config-isakmp configuration mode.

    Example

    ```
    Device(config-isakmp)# exit
    ```

IKEv2 transform sets

This concept explains how IKEv2 transform sets define the parameters used in the negotiation of IKEv2 Security Associations during the IKE_SA_INIT exchange.

An IKEv2 transform set is a security proposal that

- consists of a set of cryptographic transforms (encryption, integrity, and Diffie-Hellman group) used during IKEv2 negotiation,
- is required for a valid IKEv2 Security Association (SA), and
- enables the devices to negotiate compatible security parameters during the IKE_SA_INIT phase.

Expanded explanation

If you do not attach a user-configured IKEv2 proposal to an IKEv2 policy, the system uses the default proposal for negotiation.

Example

This configuration snippet demonstrates how to define and apply an IKEv2 transform set and related parameters:

```
device # conf t
device(config)#crypto ikev2 proposal sample
device(config-ikev2-proposal)# integrity sha1
device (config-ikev2-proposal)# encryption aes-cbc-128
device(config-ikev2-proposal)# group 14
device(config-ikev2-proposal)# exit
device(config)# crypto ikev2 keyring keyring-1
device (config-ikev2-keyring)# peer peer1
device (config-ikev2-keyring-peer)# address 192.0.2.4 255.255.255.0
device (config-ikev2-keyring-peer)# pre-shared-key cisco123!cisco123!CISC
device (config-ikev2-keyring-peer)# exit
device(config)#crypto ikev2 keyring keyring-1
device (config-ikev2-keyring)# peer peer1
device (config-ikev2-keyring-peer)# address 192.0.2.4 255.255.255.0
device (config-ikev2-keyring-peer)# pre-shared-key cisco123!cisco123!CISC
device (config-ikev2-keyring-peer)# exit
device(config)#crypto logging ikev2
```

Configure IPsec using IKEv2

Use this procedure to configure IPSec using Internet Key Exchange version 2.

Follow the procedure given below to configure the IPsec with IKEv2:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Define an IKEv2 proposal name.

Example

```
Device(config)# crypto ikev2 proposal name
```

3. Define the integrity algorithm for the IKEv2 proposal.

Example

```
Device(config-ikev2-proposal)# integrity sha1
```

4. Configure IPsec IKEv2 to use AES-CBC-128 for payload encryption.

Example

```
Device(config-ikev2-proposal)# encryption aes-cbc-128
```

AES-CBC-256 can be selected with encryption aes-cbc-256. AES-GCM-128 and AES-GCM-256 can also be selected similarly.

> **Note**
>
> The authorized administrator must ensure that the keysize for this setting is greater than or equal to the keysize selected for ESP in section IPsec Transforms and Lifetimes. If AES 128 is selected here, then the highest keysize that can be selected on the device for ESP is AES 128 (either CBC or GCM).
>
> Both confidentiality and integrity are configured with the **hash sha** and **encryption aes** commands respectively. As a result, confidentiality-only mode is disabled.

5. Selects DH Group 14 (2048-bit MODP) for IKE.

Example

```
Device(config-ikev2-proposal)# group 14
```

However, 19 (256-bit Random ECP), 24 (2048-bit MODP with 256-bit POS), 20 (384-bit Random ECP), 15 (3072 bit MODP), and 16 (4096-bit MODP) are also allowed and supported.

6. Exit IKEv2 proposal configuration mode.

Example

```
Device(config-ikev2-proposal)# exit
```

7. Define an IKEv2 keyring.

Example

```
Device(config)# crypto ikev2 keyring keyring-name
```

8. Define the peer or peer group.

Example

```
Device(config-ikev2-keyring)# peer peer-name
```

9. Specify an IPv4 or IPv6 address or range for the peer.

Example

```
Device(config-ikev2-keyring)# address 192.0.2.4 255.255.255.0
```

> **Note**
>
> This IP address is the IKE endpoint address and is independent of the identity address.

10. Specify the preshared key for the peer.

Example

```
Device(config-ikev2-keyring)# pre-shared-key cisco123!cisco123!CISC
```

You can enter the local or remote keyword to specify an asymmetric preshared key. By default, the preshared key is symmetric.

> **Note**
>
> To ensure a secure configuration, we recommend that you enter the pre-shared keys with at least 22 characters in length and can be composed of any combination of upper and lower case letters, numbers, and special characters (that include: “!”, “@”, “#”, “$”, “%”, “^”, “&”, “*”, “(“, and “)”).
>
> The device supports pre-shared keys up to 127 characters in length. While longer keys increase the difficulty of brute-force attacks, longer keys increase processing time.
>
> HEX keys generated off system can also be input for IKEv2 using the **pre-shared-key hex** *[hex key]* command instead of the pre-shared-key command above:. For example: pre-shared-key hex 0x6A6B6C. This configures IPsec to use pre-shared keys.

11. Exit IKEv2 keyring peer configuration mode.

Example

```
Device(config-ikev2-keyring)# exit
```

12. Enable IKEv2 syslog messages.

Example

```
Device(config)# crypto logging ikev2
```

> **Note**
>
> The configuration above is not a complete IKE v2 configuration, and that additional settings will be needed.

IPsec transforms and lifetimes

This section lists configuration commands, available algorithms, and default values for IPsec transforms and security association (SA) lifetimes for IPsec ESP encryption and integrity.

Regardless of the IKE version, the device must be configured with appropriate IPsec ESP transforms (encryption and integrity algorithms) and SA lifetimes. The tables and commands below show how to set these parameters.

This configuration sets IPsec ESP to use HMAC-SHA-1 and AES-CBC-128. To change this to any of the other allowed algorithms, replace **esp-aes 128** in the command

IPsec transform command example

To configure an IPsec ESP transform set with AES-CBC-128 and HMAC-SHA-1:

```
device(config)# crypto ipsec transform-set example esp-aes 128 esp-sha-hmac
```

This command sets ESP to use HMAC-SHA-1 for integrity and AES-CBC-128 for encryption.

Supported Encryption Algorithms

The table lists supported encryption algorithms and their commands:

Encryption Algorithm	Command
AES-CBC-256	esp-aes 256
AES-GCM-128	esp-gcm 128
AES-GCM-256	esp-gcm 256

> **Note**
>
> The key size for the algorithm in the transform set must not exceed the key size selected for IKE encryption. For example, if AES-CBC-128 is selected for IKE, then only AES-CBC-128 or AES-GCM-128 may be selected here.

Configuring IPsec modes

Set the operational mode for IPsec using one of these commands:

```
device(config-crypto)# mode tunnel
```

- Requests and accepts only tunnel mode for IPsec. Tunnel is the default mode.

```
device(config-crypto)# mode transport
```

- Configures transport mode for IPsec.

Configuring SA lifetimes

By time (seconds):

The default lifetime for Phase 2 SAs is 3600 seconds (1 hour). To change it (for example to 8 hours or 28800 seconds) use this command:

```
device(config)# crypto ipsec security-association lifetime seconds 28800
```

By data volume (kilobytes):

Default is 2560 KB (minimum configurable). The maximum is 4 GB. For example, to set a 100 MB (100000 KB) limit use this command:

```
device(config)# crypto ipsec security-association lifetime kilobytes 100000
```

Additional notes

- No configuration is needed for default values; configure only for custom values to meet policy or compliance.
- Always ensure that your IPsec transforms and lifetimes are compatible between your IPsec peers to prevent negotiation failures.

Use of X.509 with IKE version

Use this procedure to configure X.509 certificate-based authentication for IKEv1 on the Cisco Catalyst 9800 Series Wireless Controller.

Enable X.509v3 certificate authentication for secure IKEv1 negotiation.

The controller supports RSA and ECDSA based certificates. Once X.509v3 keys are installed on the device, they can be set for use with IKEv1 with the commands:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Define an Internet Key Exchange (IKE) policy and assign a priority to the policy.

   Example

   ```
   Device(config)# crypto isakmp policy policy-name
   ```

3. Use RSA based certificates for IKEv1 authentication.

   Example

   ```
   Device(config-isakmp)# authentication rsa-sig
   ```

4. Use ECDSA based certificates for IKEv1 authentication.

Example

```
Device(config-isakmp)# authentication ecdsa-sig
```

The X.509 certificates are now configured for use with IKEv1.

IPsec session interruption and recovery

This section describes the behavior of IPsec sessions when an unexpected interruption occurs and explains the automatic recovery process when the peer reconnects..

If an IPsec session with a peer is unexpectedly interrupted, the connection will be broken. In this scenario, no administrative interaction is required. The IPsec session will be reestablished (a new SA set up) once the peer is back online.

Example: Configure IPsec using ISAKMP

This section provides a sample configuration for IPsec using ISAKMP.

This example displays the IPSec **isakmp** configuration.

The sample outputs display the IPSec **isakmp** configuration:

```
crypto isakmp policy 1
                 encr aes 256
                 hash sha256
                 authentication pre-share
                 group 14
                 lifetime 28800
                crypto isakmp key 0 Cisco!123 address 192.0.2.4
                crypto isakmp peer address 192.0.2.4
                crypto ipsec transform-set aes-gcm-256 esp-gcm 256
                 mode tunnel
                crypto map IPSEC_ewlc_to_syslog 1 ipsec-isakmp
                 set peer 192.0.2.4
                 set transform-set aes-gcm-256
                 match address acl_ewlc_to_syslog
                interface Vlan15
                 crypto map IPSEC_ewlc_to_syslog
                end
```

Verify IPSec traffic

This section lists sample commands and their outputs to help you verify the status of IPSec tunnels and Internet Key Exchange Version 2 (IKEv2) traffic on a device.

This example shows how to verify the IPSec traffic configuration in isakmp configuration:

```
Device# show crypto map
Crypto Map IPv4 "IPSEC_ewlc_to_syslog" 1 ipsec-isakmp
        Peer = 192.0.2.4
        Extended IP access list acl_ewlc_to_syslog
            access-list acl_ewlc_to_syslog permit ip host 192.0.2.2 host
192.0.2.4
        Current peer: 192.0.2.4
        Security association lifetime: 4608000 kilobytes/3600 seconds
        Responder-Only (Y/N): N
        PFS (Y/N): N
```

```
        Mixed-mode : Disabled
        Transform sets={
                aes-gcm-256:  { esp-gcm 256  } ,
        }
        Interfaces using crypto map IPSEC_ewlc_to_syslog:
                Vlan15
```

```
Device# show crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status
192.0.2.5       192.0.2.4       QM_IDLE           1011 ACTIVE

IPv6 Crypto ISAKMP SA
```

```
Device# show crypto ipsec sa

interface: Vlan15
    Crypto map tag: IPSEC_ewlc_to_syslog, local addr 192.0.2.5

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (192.0.2.5/255.255.255.255/0/0)
   remote ident (addr/mask/prot/port): (192.0.2.4/255.255.255.255/0/0)
   current_peer 192.0.2.4 port 500
     PERMIT, flags={origin_is_acl,}
    #pkts encaps: 1626, #pkts encrypt: 1626, #pkts digest: 1626
    #pkts decaps: 1625, #pkts decrypt: 1625, #pkts verify: 1625
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 0, #recv errors 0

     local crypto endpt.: 192.0.2.5, remote crypto endpt.: 192.0.2.4
     plaintext mtu 1446, path mtu 1500, ip mtu 1500, ip mtu idb Vlan15
     current outbound spi: 0x17FF2F4C(402599756)
     PFS (Y/N): N, DH group: none

     inbound esp sas:
      spi: 0x4B77AD78(1266134392)
        transform: esp-gcm 256 ,
        in use settings ={Tunnel, }
        conn id: 2041, flow_id: HW:41, sibling_flags FFFFFFFF80004048, crypto
 map: IPSEC_ewlc_to_syslog
        sa timing: remaining key lifetime (k/sec): (4607904/1933)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)

     inbound ah sas:

     inbound pcp sas:

     outbound esp sas:
      spi: 0x17FF2F4C(402599756)
        transform: esp-gcm 256 ,
        in use settings ={Tunnel, }
        conn id: 2042, flow_id: HW:42, sibling_flags FFFFFFFF80004048, crypto
 map: IPSEC_ewlc_to_syslog
        sa timing: remaining key lifetime (k/sec): (4607904/1933)
        IV size: 8 bytes
        replay detection support: Y
        Status: ACTIVE(ACTIVE)
```

```
     outbound ah sas:
     outbound pcp sas:
```

```
Device# show ip access-lists acl_ewlc_to_syslog
Extended IP access list acl_ewlc_to_syslog
    10 permit ip host 192.0.2.5 host 192.0.2.4 (17 matches)
```

Example: Configure IPsec using IKEv2

This section provides a sample configuration for IPSec using Internet Key Exchange Version 2 (IKEv2).

The sample outputs display the IPsec **IKEv2** configuration:

```
topology : [192.0.2.6]DUT — (infra) — PEER[192.0.2.9]
ikev2 config in 192.0.2.6 (peer is 192.0.2.9)
hostname for 192.0.2.9: Edison-M1
hostname for 192.0.2.6: prsna-nyquist-192.0.2.6
ip access-list extended ikev2acl
 permit ip host 192.0.2.6 host 192.0.2.9
crypto ikev2 proposal PH1PROPOSAL
 encryption aes-cbc-256
 integrity sha256
 group 14
!
crypto ikev2 policy PH1POLICY
 proposal PH1PROPOSAL
crypto ikev2 keyring PH1KEY
  peer Edison-M1
  address 192.0.2.9
  pre-shared-key Cisco!123Cisco!123Cisco!123
crypto ikev2 profile PH1PROFILE
 match identity remote address 192.0.2.9 255.255.255.255
 authentication remote pre-share
 authentication local pre-share
 keyring local PH1KEY
crypto ipsec transform-set aes256-sha1 esp-aes 256 esp-sha-hmac
 mode tunnel
crypto map ikev2-cryptomap 1 ipsec-isakmp
 set peer 192.0.2.9
 set transform-set aes256-sha1
 set ikev2-profile PH1PROFILE
 match address ikev2acl
interface Vlan15
 ip address 192.0.2.6 255.255.255.0
 crypto map ikev2-cryptomap
```

Verify IPSec with IKEv2 traffic

Verify IPSec with Internet Key Exchange Version 2 traffic.

The following example shows how to verify the IPSec traffic configuration in IKEv2 configuration:

```
Device# show ip access-lists
                Extended IP access list ikev2acl
                    10 permit ip host 192.0.2.6 host 192.0.2.9 (80 matches)
                Device#show crypto map
                Crypto Map IPv4 "ikev2-cryptomap" 1 ipsec-isakmp
                        Peer = 192.0.2.9
```

```
                        IKEv2 Profile: PH1PROFILE
                        Extended IP access list ikev2acl
                           access-list ikev2acl permit ip host 192.0.2.6 host
 192.0.2.9
                        Current peer: 192.0.2.9
                        Security association lifetime: 4608000 kilobytes/3600
 seconds
                        Responder-Only (Y/N): N
                        PFS (Y/N): N
                        Mixed-mode : Disabled
                        Transform sets={
                                aes256-sha1:  { esp-256-aes esp-sha-hmac  }
,
                        }
                        Interfaces using crypto map ikev2-cryptomap:
                                Vlan15
                Device# show crypto ikev2 sa detailed
                 IPv4 Crypto IKEv2  SA
               Tunnel-id Local                Remote                fvrf/ivrf
           Status
                1         192.0.2.6/500         192.0.2.9/500
none/none            READY
                 Encr: AES-CBC, keysize: 256, PRF: SHA256, Hash: SHA256,
 DH Grp:14, Auth sign: PSK, Auth verify: PSK
                 Life/Active Time: 86400/1002 sec
                 CE id: 1089, Session-id: 2
                 Status Description: Negotiation done
                 Local spi: 271D20169FE91074       Remote spi:
13895472E3B910AF
                 Local id: 192.0.2.6
                 Remote id: 192.0.2.9
                 Local req msg id:  2              Remote req msg id:
0
                 Local next msg id: 2              Remote next msg id:
0
                 Local req queued:  2              Remote req queued:
0
                 Local window:      5              Remote window:
5
                 DPD configured for 0 seconds, retry 0
                 Fragmentation not  configured.
                 Dynamic Route Update: disabled
                 Extended Authentication not configured.
                 NAT-T is not detected
                 Cisco Trust Security SGT is disabled
                 Initiator of SA : Yes
                Device# show crypto ipsec sa detail
                interface: Vlan15
                    Crypto map tag: ikev2-cryptomap, local addr 192.0.2.6
                   protected vrf: (none)
                   local  ident (addr/mask/prot/port):
(192.0.2.6/255.255.255.255/0/0)
                   remote ident (addr/mask/prot/port):
(192.0.2.9/255.255.255.255/0/0)
                   current_peer 192.0.2.9 port 500
                     PERMIT, flags={origin_is_acl,}
                    #pkts encaps: 80, #pkts encrypt:80, #pkts digest: 80
                    #pkts decaps: 80, #pkts decrypt: 80, #pkts verify: 80
                    #pkts compressed: 0, #pkts decompressed: 0
                    #pkts not compressed: 0, #pkts compr. failed: 0
                    #pkts not decompressed: 0, #pkts decompress failed: 0
                    #pkts no sa (send) 0, #pkts invalid sa (rcv) 0
```

```
                    #pkts encaps failed (send) 0, #pkts decaps failed (rcv) 0

                    #pkts invalid prot (recv) 0, #pkts verify failed: 0
                    #pkts invalid identity (recv) 0, #pkts invalid len (rcv)
0
                    #pkts replay rollover (send): 0, #pkts replay rollover
(rcv) 0
                    ##pkts replay failed (rcv): 0
                    #pkts tagged (send): 0, #pkts untagged (rcv): 0
                    #pkts not tagged (send): 0, #pkts not untagged (rcv): 0
                    #pkts internal err (send): 0, #pkts internal err (recv) 0

                     local crypto endpt.: 192.0.2.6, remote crypto endpt.:
192.0.2.9
                     plaintext mtu 1438, path mtu 1500, ip mtu 1500, ip mtu
idb Vlan15
                     current outbound spi: 0xB546157A(3041269114)
                     PFS (Y/N): N, DH group: none
                     inbound esp sas:
                      spi: 0x350925BC(889791932)
                        transform: esp-256-aes esp-sha-hmac ,
                        in use settings ={Tunnel, }
                        conn id: 838, flow_id: 838, sibling_flags
FFFFFFFF80000040, crypto map: ikev2-cryptomap
                        sa timing: remaining key lifetime (k/sec):
(4287660676/2560)
                        IV size: 16 bytes
                        replay detection support: Y
                        Status: ACTIVE(ACTIVE)
                     inbound ah sas:
                     inbound pcp sas:
                     outbound esp sas:
                      spi: 0xB546157A(3041269114)
                        transform: esp-256-aes esp-sha-hmac ,
                        in use settings ={Tunnel, }
                        conn id: 837, flow_id: 837, sibling_flags
FFFFFFFF80000040, crypto map: ikev2-cryptomap
                        sa timing: remaining key lifetime (k/sec):
(4287660672/2560)
                        IV size: 16 bytes
                        replay detection support: Y
                        Status: ACTIVE(ACTIVE)
                     outbound ah sas:
                     outbound pcp sas:
```

159 Transport Layer Security Tunnel Support

Topics:

- Transport layer security tunnels
- Configure a transport layer security tunnel
- Transport layer security tunnel verification

Transport layer security tunnels

Explains how Transport Layer Security (TLS) tunnel support enables the controller to reach a public cloud automatically and securely manage wireless solutions.

A transport layer security tunnel is a secure communication protocol that

- enables the controller to reach a public cloud automatically
- encrypts all data sent over the TCP connection, and
- ensures reliable and secure communication between the controller and Digital Network Architecture (DNA) on Cloud.

Cisco plug and play solution

Supported controllers

The Cisco Catalyst 9800 Series Wireless Controller requires direct access to a public cloud to implement the teleworker solution using Cisco OfficeExtend Access Points (OEAPs). With the introduction of Transport Layer Security (TLS) tunnel support from Cisco IOS XE Amsterdam 17.3.2 onwards, the controller can now reach a public cloud automatically. This helps Cisco Catalyst Center on Cloud to establish TLS communication channels with the controller to monitor and manage wireless solutions.

The TLS connection ensures that the configuration and telemetry are reliably and securely communicated between the controller and the Digital Network Architecture (DNA) on Cloud. The TLS tunnel encrypts all the data that is sent over the TCP connection. The TLS tunnel provides a more secure protocol across the internet. After the controller discovery, the Cisco Catalyst Center on Cloud uses Cisco DNA Assurance and Automation features to manage the controller centrally.

The Cisco Plug and Play solution is a converged solution that provides a highly secure, scalable, seamless, and unified zero-touch deployment experience.

Plug-n-Play Agent

The Cisco Plug and Play (PnP) agent is an embedded software component that is present in all the Cisco network devices that support simplified deployment architecture. The PnP agent understands and interacts only with a PnP server. The PnP agent, using DHCP, DNS, or other such methods, tries to acquire the IP address of the PnP server with which it wants to communicate. After a server is found and a connection is established, the agent communicates with the PnP server to perform deployment-related activities.

For more information on Cisco Plug and Play, see the Cisco Plug and Play Feature Guide.

The Transport Layer Security Tunnel (TLS) over PnP feature is supported on the following controllers:

- Cisco Catalyst 9800-80 Wireless Controller
- Cisco Catalyst 9800-40 Wireless Controller
- Cisco Catalyst 9800-L Wireless Controller

Configure a transport layer security tunnel

Configure a transport layer security tunnel to enable secure communication between devices.

Establish a secure TLS tunnel to protect data exchanged between network devices.

Use this procedure when you need to secure traffic between devices using a TLS tunnel, typically in environments requiring encrypted communication for compliance or security reasons.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a crypto TLS tunnel channel.

   Example

   ```
   Device(config)# crypto tls-tunnel TLS-tunnel-name
   ```

   Example

   ```
   Device(config)# crypto tls-tunnel cloud-primary
   ```

3. Specify the server IPv4 address, IPv6 address, or URL name and the port number.

   Example

   ```
   Device(config-crypto-tls-tunnel)# server ipv4 server-ipv4-address port
   port-number
   ```

   Example

   ```
   Device(config-crypto-tls-tunnel)# server ipv4 172.31.255.255 port 4043
   ```

   The server can be specified as an IPv4 address, IPv6 address, or URL. The port number must be in the range 1025-65535.

4. Specify the overlay interface and interface number.

   Example

   ```
   Device(config-crypto-tls-tunnel)# overlay interface interface-name
   ```

   Example

   ```
   Device(config-crypto-tls-tunnel)# overlay interface Loopback0
   ```

   An overlay interface is a logical, multiaccess, multicast-capable interface. It encapsulates Layer 2 frames in IP unicast or multicast headers.

5. Specify the LAN interface type, number, and the priority rank.

   Example

   ```
   Device(config-crypto-tls-tunnel)# local-interface interface-name priority
   priority-rank
   ```

Example

```
Device(config-crypto-tls-tunnel)# local-interface vlan 1 priority 1
```

Currently, the tunnel supports only one WAN interface with priority 1 and does not support the list of WAN interfaces with multiple priorities.

6. Specify a preshared key and password options.

Example

```
Device(config-crypto-tls-tunnel)# psk id identity key psk-key
```

Example

```
Device(config-crypto-tls-tunnel)# psk id test key
```

7. Specify the trustpoints for use with the RSA signature authentication method.

Example

```
Device(config-crypto-tls-tunnel)# pki trustpoint trustpoint-label
sign-or-verify
```

Example

```
Device(config-crypto-tls-tunnel)# pki trustpoint tsp1 sign
```

Use the certificate from the trustpoint for signing or verifying. If the **sign** or **verify** keyword is not specified, the trustpoint is used for both signing and verification. Authentication can be done using either pre-shared key (PSK) or PKI (certificate based).

8. Optional: Enable common criteria mode (FIPS mode).

Example

```
Device(config-crypto-tls-tunnel)# cc-mode
```

Indicates a common criteria mode, which is a Federal Information Processing Standards (FIPS) mode.

9. Enable the TLS tunnel.

Example

```
Device(config-crypto-tls-tunnel)# no shutdown
```

10. Return to privileged EXEC mode.

Example

```
Device(config-crypto-tls-tunnel)# end
```

The TLS tunnel is now configured and enabled, providing secure encrypted communication between devices.

Transport layer security tunnel verification

Provides steps and commands to verify a transport layer security (TLS) tunnel, including client summary, session details, statistics, and session summary information.

This reference describes how to verify a transport layer security (TLS) tunnel by using various show commands to display client summary, session details, statistics, and session summary information.

The TLS client support includes BinOS processes using Linux Tun/Tap Interface. To verify the TLS client summary details, use the following command:

```
Device# show platform software tlsc client summary
TLS Client - Config Summary

Name         ID     Gateway    Port   Auth     Trustpoint  DPD Time    Rekey
Time  Retry Time
-------------------------------------------------------------------------------------
fqdn         0                 8443   PSK      N/A            60          300
         20
```

To verify the TLS client session detail, session statistics, tunnel statistics, and DNS counters, use the following command:

```
Device# show platform software client detail <tls-name>

Session Name     : fqdn
FQDN resolved IP : 10.255.255.255
ID               : 0
Created          : 04/20/21 00:36:42
Updated          : 04/22/21 05:56:03
State            : Up (Rekey)
Up Time          : 04/21/21 20:30:21 (9 hours 25 minutes 45 seconds)
Down Time        : 04/21/21 20:30:01
Rekey Time       : 04/22/21 05:55:51 (15 seconds)

TLS Session Statistics
Up Notifications    : 3
Down Notifications  : 2
Rekey Notifications : 636
DP State Updates    : 0
DPD Cleanups        : 0

Packets From         Packets To    Packet Errors To     Bytes From
Bytes To
-------------------------------------------------------------------------------
BinOS                  80              0                    0
   0
IOSd                    0              0                    0
   0
TLS Client              0              0                    0
   0

TLS Tunnel Statistics

Type       Tx Packets           Rx Packets
-----------------------------------------
Total        0                    80
CSTP Ctrl   3836                 3836
CSTP Data    80                   0

Type       Requests             Responses
```

```
----------------------------------------
CSTP Cfg   639                  639
CSTP DPD   3197                 3197

Invalid CSTP Rx         : 0
Injected Packet Success : 0
Injected Packet Failed  : 0
Consumed Packets        : 0

TLS Tunnel DNS Counters

DNS Resolve Request Success Count : 641
DNS Resolve Request Failure Count : 0
DNS Resolve Success Count         : 639
DNS Resolve Failure Count         : 2
```

To verify the TLS client global statistics, use the following command.

```
Device# show platform software tlsc statistics
TLS Client: Global Statistics

Session Statistics
Up / Down     : 5 / 2
Rekeys        : 636
DP Updates    : 0
DPD Cleanups  : 0

            Packets From   Packets To     Packet Errors To     Bytes From
  Bytes To
-------------------------------------------------------------------------------------
BinOS        85              0
    0
IOSd         0               0                0                    0
    0
TLS Client   0               0                0                    0
    0

Tunnel Statistics

SSL Handshake Init / Done : 641 / 641
TCP Connection Req / Done : 641 / 641
Tunnel Packets
Rx / Tx           : 85 / 0
Injected / Failed : 0 / 0
Consumed          : 0

CSTP Packets
Control Rx / Tx   : 3839 / 3839
Data    Rx / Tx   : 0 / 85
Config Req / Resp : 641 / 641
DPD    Req / Resp : 3198 / 3198
Invalid Rx        : 0

FQDN Counters
Req / Resp / Success : 0 / 0 / 0

NAT Counters
Transalte In / Out : 0 / 0
Ignore    In / Out : 0 / 0
Failed             : 0
Invalid            : 0
No Entry           : 0
```

```
Unsupported        : 0

Internal Counters

Type     Allocated    Freed
----------------------------
EV       1299         1295
Tunnel    5            4
Conn     643           642
Sess      3            2

Config Message Related Counters
Type     Success       Failed
-----------------------------
Create     3             0
Delete     2             0
```

To view the TLS client-session summary, use the following command.

```
Device# show platform software tlsc session summary

TLS Client - Session Summary

Name     ID     Created            State       Since               Elapsed
-------------------------------------------------------------------------------
fqdn     0      04/20/21 00:36:42  Up          04/21/21 20:30:21  9 hours 26
minutes 44 seconds
```

160 Configuring RFC 5580 Location Attributes

Topics:

- RFC 5580 location attributes

This chapter shows how to enable and manage civic and geo profiles, operator names, AP/NAS associations, out-of-band delivery, and verification commands on Cisco controllers to send location data in RADIUS exchanges.

RFC 5580 location attributes

Introduces RFC 5580 related location attributes, covering the location-capable attribute, configuration restrictions, delivery of location information using out-of-band agreements, methods for defining civic and geo profiles, operator name setup, associating location attributes with user and NAS locations, and verification steps for attribute deployment.

An RFC 5580 location attribute is a RADIUS protocol attribute that

- conveys location related information for authentication and accounting exchanges,
- supports both user and NAS locations using Civic profiles and Geo profiles, and
- enables services such as location-aware authorization, billing, and enhanced user privacy.

Feature history for RFC 5580 location attributes

This table lists the release and related information for this feature.

This feature is available in all subsequent releases unless stated otherwise.

Table 142: Feature history for RFC 5580 location attributes

Release	Feature	Feature information
Cisco IOS XE 17.9.1	Support for RFC 5580 location attributes in the controller	This feature uses RFC 5580 location attributes to convey location related information for authentication and accounting exchanges. The controller supports these RFC 5580-related attributes: • Location-Information • Location-Data CIVIC Profile: Country • Location-Data CIVIC Profile: CAtype 1 (state) • Location-Data CIVIC Profile: CAtype 3 (city) • Location-Data CIVIC Profile: CAtype 23 (venue name) • Location-Data CIVIC Profile: CAtype 24 (zip code) • Location-Data GEO Profile (longitude, latitude, and altitude) • Operator Name

Various operators use RFC 5580 location attributes in wireless networks deployed in public places such as shopping malls, airports, hotels, and coffee shops.

To enable location-aware authorization, billing, or services, the network may require the user location.

You must protect the location information against unauthorized access and distribution.

- **User location** specifies a user-specific location, typically configured at the access point.
- **NAS location** refers to the common host location for all users connected to a specific network access server, configurable in AAA (Authentication, Authorization, and Accounting).

Each location can have two profiles: a Civic profile and a Geo profile.

- **Civic Profile** describes a location using civic attributes such as country, state, city, area, and postal code.
- **Geo Profile** describes a location using geographic attributes such as latitude, longitude, and altitude.

Location Attributes and Profiles

If a user has both user location and NAS location, you can configure each location using Civic profiles and Geo profiles.

You can configure these locations:

- Civic user location
- Civic NAS location
- Geo user location
- Geo NAS location

Each location information, for example, the civic user location, is sent using the attributes that include:

- Location-Information
- Location-Data

The controller supports these RFC 5580-related attributes:

- Location-Information
- Location-Data CIVIC Profile: Country
- Location-Data CIVIC Profile: CAtype 1 (state)
- Location-Data CIVIC Profile: CAtype 3 (city)
- Location-Data CIVIC Profile: CAtype 23 (venue name)
- Location-Data CIVIC Profile: CAtype 24 (zip code)
- Location-Data GEO Profile (longitude, latitude, and altitude)
- Operator Name

You can configure four locations and one operator name for each user.

The Out-of-Band Agreement delivery method (Flow one) mentioned in RFC 5580 is supported for transferring location information if the feature is enabled and location information is configured.

Location-capable attribute

Information about the location-capable attribute.

A location-capable attribute is a Remote Authentication Dial-In User Service (RADIUS) attribute that

- signals to a RADIUS server that a device is capable of providing location information,
- is included in network access authentication or authorization requests for wireless clients, and
- enables support for location-based services as specified in RFC 5580.

Additional reference information

Cisco IOS XE 17.11.1 supports the location-capable attribute defined by RFC 5580. You can enable this attribute by using the following command: **radius-server attribute wireless location delivery out-of-band include-location-capable** .

According to RFC 5580, the attribute is typically sent in Flow two, which refers to location delivery based on the Initial Request. This configuration also enables sending the attribute in Flow one, which refers to an out of band agreement.

The location-capable attribute appears in RADIUS requests for wireless clients. The server may use this data to provide or restrict network access based on device location capability.

Restriction for configuring RFC 5580 location attributes

Restriction for configuring RFC 5580 location attributes.

This feature is supported only for 802.11ax users.

Configure location delivery based on out-of-band agreement (CLI)

Configure location delivery based on out-of-band agreement using commands.

Configure RFC 5580 out-of-band location delivery on network devices through CLI.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure RFC 5580 Out-of-Band location support.

   Example

   ```
   Device(config)# radius-server attribute wireless location delivery out-of-band
   ```

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The device is configured to support out-of-band location delivery as per RFC 5580.

Configure the location-capable attribute (CLI)

Configure location-capable attribute using commands.

Enable and include the location-capable attribute in RADIUS access requests for wireless location delivery features according to RFC 5580.

Before you begin

Use the **radius-server attribute wireless location delivery out-of-band** command to enable the feature globally.

You can use the **radius-server attribute wireless location delivery out-of-band include-location-capable** command to include the location-capable attribute along with other location attributes.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure RFC 5580 out-of-band location attributes along with enabling the location-capable attribute to be part of the access request.

   Example

   ```
   Device(config)# radius-server attribute wireless location delivery out-of-band
    include-location-capable
   ```

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The device includes the location-capable attribute in RADIUS access requests, enabling the server to support advanced wireless location tracking and services.

Configure a civic profile (CLI)

Use commands to configure the civic profile.

Configure civic profiles to specify user location information for Remote Authentication Dial-In User Service (RADIUS) requests and network services.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the civic profile for the user location.

   Example

   ```
   Device(config)# location civic-location identifier civic_identifier
   ```

   Here, *civic_identifier* refers to the civic location identifier string. The identifier can be up to 215 characters long. You can allocate up to 250 bytes for civic address attributes. Of these, Cisco reserves 50 bytes for internal information, leaving 200 bytes available for user-configured civic location information.

> **Note**
>
> You can configure these types of civic attributes and add them to RADIUS requests:
>
> - Country,
> - City,
> - State,
> - ZIP code, and
> - Name

3. Set the country ID.

   Example

   ```
   Device(config-civic)# country country_ID
   ```

   > **Note**
   >
   > Only two-letter ISO 3166 country codes are accepted.

4. Set the city name.

   Example

   ```
   Device(config-civic)# city city_name
   ```

5. Set the state name.

   Example

   ```
   Device(config-civic)# state state_name
   ```

6. Set the ZIP code.

   Example

   ```
   Device(config-civic)# postal-code postal_code
   ```

7. Set the residence name.

   Example

   ```
   Device(config-civic)# name residence_name
   ```

8. Return to privileged EXEC mode.

   Example

   ```
   Device(config-civic)# end
   ```

The civic profile is configured, and the device now uses the specified location details in RADIUS requests.

Configure a geo profile (CLI)

Use CLI commands to set location parameters—identifier, latitude, longitude, altitude, and resolution—for your network device.

Set precise geographic information—identifier, latitude, longitude, altitude, and resolution—for the user's location on your device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a geo profile for the user location.

   Example

   ```
   Device(config)# location geo-location identifier geo_identifier
   ```

   Here, the geographic location identifier string is referenced. It can contain up to 215 characters.

3. Set latitude information.

   Example

   ```
   Device(config-geo)# latitude latitude_in_degrees resolution resolution_value
   ```

   Optional parameters appear in square brackets.

   If you do not specify a resolution, the system uses a default value of 10 meters (32.8 feet).

4. Set longitude information.

   Example

   ```
   Device(config-geo)# longitude longitude_in_degrees resolution resolution_value
   ```

   Optional parameters appear in square brackets.

   If you do not specify a resolution, the system uses a default value of 10 meters (32.8 feet).

5. Configure altitude for the geographic location.

Example

```
Device(config-geo)# altitude altitude_value feet resolution resolution_value
 floor meters resolution resolution_value
```

Example

```
Device(config-geo)# altitude 10 meters (32.8 feet) resolution 10
```

Optional parameters appear in square brackets.

- *altitude_value* : refers to the altitude, in feet, floors, or meters.
- *resolution_value* : refers to the resolution, in feet or meters.

> **Note**
>
> Altitude and altitude resolution values must use the same unit.

6. Specify a resolution for both latitude and longitude.

Example

```
Device(config-geo)# resolution resolution_value
```

7. Return to privileged EXEC mode.

Example

```
Device(config-geo)# end
```

The geo profile is now configured on your device. It is associated with the identifier, latitude, longitude, altitude, and resolution you specified.

Configure an operator name (CLI)

Use commands to configure the operator name.

Establish a unique operator-name identifier for your device user or Network Access Server (NAS) location so you can reference it for administrative or tracking purposes.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure an operator name for the user's location.

   Example

   ```
   Device(config)# location operator identifier identifier_name
   ```

   In this context, *identifier_name* supports strings with a maximum length of 215 characters (215 bytes).

3. Configure the operator name for the location.

   Example

   ```
   Device(config-operator)# name operator-name
   ```

   In this context, *operator-name* supports strings with a maximum length of 248 characters (248 bytes).

4. Configure the namespace for the location where the operator name applies.

   Example

   ```
   Device(config-operator)# namespace-id E212 ICC REALM TADIG
   ```

   You can use these namespace options.

   - **E212** refers to the Mobile Country Code (MCC) and Mobile Network Code (MNC).
   - **ICC** refers to the International Telecommunication Union Carrier Codes (ICC).
   - **REALM** refers to any registered domain name.
   - **TADIG** refers to the Transferred Account Data Interchange Group (TADIG) code.

   Note

   - If you have not configured any namespace, **REALM** is used as the default.
   - You can associate the operator name with both NAS-Location and USER-Location. If you configure an operator name at both locations, the operator name you configure in USER-Location takes precedence.

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config-operator)# end
   ```

The operator name is now configured for the location you specified. The device uses this operator name to identify the appropriate context.

Associate a location attribute with a user location (CLI)

Use this procedure to associate civic, geographic, and operator location attributes with a user location for access points using CLI commands.

Assign location metadata to user-defined locations for APs for improved location-based services.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a location name for an AP.

   Example

   ```
   Device(config)# ap location name location_name
   ```

3. Add the AP to the location.

   Example

   ```
   Device(config-ap-location)# ap-eth-mac AP_Ethernet_MAC
   ```

   Here, *AP_Ethernet_MAC* refers to the AP Ethernet MAC address.

4. Associate the civic location attribute with the user location.

   Example

   ```
   Device(config-ap-location)# location civic-location-id identifier_name
   ```

5. Associate the geographic location attribute with the user location.

   Example

   ```
   Device(config-ap-location)# location geo-location-id identifier_name
   ```

6. Associate the operator location attribute with the user location.

   Example

   ```
   Device(config-ap-location)# location operator-id identifier_name
   ```

7. Return to privileged EXEC mode.

Example

```
Device(config-ap-location)# end
```

The selected AP has its civic, geographic, and operator attributes associated with the specified user location.

Associate the NAS location with location attributes (CLI)

Use this procedure to associate location attributes (civic, geographic, operator) with the NAS location using CLI commands.

Apply location attributes to a NAS device for wireless location tracking and compliance.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Associate the civic location attribute with the NAS location.

   Example

   ```
   Device(config)# radius-server attribute wireless location civic-location-id
   identifier_name
   ```

   Here, *identifier_name* supports strings up to 215 characters in length.

3. Associate the geographic location attribute with the NAS location.

   Example

   ```
   Device(config)# radius-server attribute wireless location geo-location-id
   identifier_name
   ```

   Here, *identifier_name* supports strings up to 215 characters in length. Enter a valid or existing identifier name.

4. Associate the operator location attribute with the NAS location.

   Example

   ```
   Device(config)# radius-server attribute wireless location operator-id
   identifier_name
   ```

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The specified location attributes are now associated with the NAS location.

Verify RFC 5580 location attribute configuration

Verify RFC 5580 location attribute configuration.

To verify the location attributes associated with a given location, use this command:

```
Device# show ap location details AAA_location
Location Name.....................: AAA_location
Location description..............:
Policy tag........................: default-policy-tag
Site tag..........................: default-site-tag
RF tag............................: default-rf-tag
AAA Location Status ..............: Enabled
Civic Location Identifier : NAS_C_1
Geo Location Identifier   : NAS_G_1
Operator Name Identifier  : NAS_O_1

Configured list of APs
38ed.18ca.5a20
```

To verify the Cisco AP location, use this command:

```
Device# show ap name AP38ED.18CA.5A20 config general
Cisco AP Name   : AP38ED.18CA.5A20
================================================

Cisco AP Identifier                              : 38ed.18cb.cf00
Country Code                                     : Multiple Countries :
Regulatory Domain Allowed by Country           : 802.11bg:   802.11a:   802.11
 6GHz:
AP Country Code                                  : US  -
AP Regulatory Domain
  802.11bg                                       : -A
  802.11a                                        : -A
MAC Address                                      : 38ed.18ca.5a20
IP Address Configuration                         : Static IP assigned
IP Address                                       : 192.0.2.254
IP Netmask                                       : 255.255.255.0
Gateway IP Address                               : 9.4.172.1
Fallback IP Address Being Used                   :
Domain                                           :
Name Server                                      :
CAPWAP Path MTU                                  : 1485
Capwap Active Window Size                        : 1
Telnet State                                     : Disabled
CPU Type                                        :  ARMv7 Processor rev 0 (v7l)
Memory Type                                      : DDR3
Memory Size                                      : 995328 KB
SSH State                                        : Disabled
Cisco AP Location                                : AAA_location
-
-
-
```

To verify the location attributes associated with a given MAC address, use this command:

```
Device# show wireless client mac 0080.5222.545c detail

Client MAC Address : 0080.5222.545c
```

```
Client MAC Type : Universally Administered Address
Client DUID: NA
Client IPv4 Address :
AP MAC Address : 38ed.18cb.cf00
AP Name: AP38ED.18CA.5A20
AP slot : 1
Client State : Associated
Policy Profile : default-policy-profile
Flex Profile : N/A
…
Civic Location Identifier : NAS_C_1
Geo Location Identifier   : NAS_G_1
Operator Name Identifier  : NAS_O_1
```

Note

You will be able to view this output only if the RFC 5580 feature is enabled.

To verify the Civic location details, use this command:

```
Device# show location civic-location identifier TEST1
Civic location information
--------------------------
Identifier              : TEST1
Name                    : home
City                    : Morges
State                   : Vaud
Postal code             : 1110
Country                 : CH
```

To verify the Geo location details, use this command:

```
Device# show location geo-location identifier TEST4
Geo location information
------------------------
Identifier  : TEST4
Latitude    : 46.5112700
Longitude   : 6.4985400
Altitude    : 380 meters          Resolution : 10
Resolution  : 100
```

To verify the Operator location details, use this command:

```
Device# show location operator-location identifier myoperator
Operator location information
-----------------------
Operator Identifier     : myoperator
Operator Name           : myoperator
Operator Namespace      : REALM
-----------------------
```

161 IP MAC Binding

Topics:

- IP MAC binding

Explains the IP-MAC binding feature and provides guidance on configuring and managing MAC address to IP address associations for network security.

IP MAC binding

Explains the IP MAC address binding configuration that enables wireless device tracking features, such as theft detection, proxy, DHCP relay, gleaning, and suppression.

IP MAC binding is a wireless configuration feature that

- enables wireless device tracking features such as theft detection, proxy, DHCP relay, gleaning, and suppression
- tracks wireless clients' IPv4 addresses by binding them to their MAC addresses, and
- is enabled by default in the policy profile.

No IP MAC binding

No IP MAC binding disables all the wireless device tracking features for wireless clients' IPv4 address.

> **Note**
>
> It is not normally necessary to disable IP MAC binding, except for the following scenarios:
>
> - When you have a single wireless station with multiple IP addresses.
> - When you intentionally have duplicate IP addresses across clients.
> - When you are using ARP-spoofing Network Access Control (NAC) devices.

> **Note**
>
> The IP MAC address binding is enabled by default in the policy profile.

> **Note**
>
> IP MAC binding is required for DHCP Option 82 to work in some situations.

Use cases for no IP MAC binding

Lists use cases for no IP MAC binding.

The following are the use cases for No IP MAC binding:

- Disabling IP Learning in FlexConnect Mode
- Disabling Device Tracking to Support NAC Devices
- Disabling IP Learning in FlexConnect Mode

Disable IP MAC binding (CLI)

Configure wireless policy settings to disable IP MAC binding using command line interface.

Disable IP MAC binding on a wireless policy profile to remove the restriction that binds client IP addresses to their MAC addresses.

IP MAC binding prevents clients from using IP addresses that are not associated with their MAC addresses. Disabling this feature allows more flexible IP address assignment for wireless clients.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the wireless profile policy.

   Example

   ```
   Device(config)# wireless profile policy profile-policy-name
   ```

   Example

   ```
   Device(config)# wireless profile policy test-profile-policy
   ```

3. Disable the wireless policy profile.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

   Disabling policy profile results in associated AP and client to rejoin.

4. Disable IP MAC binding.

   Example

   ```
   Device(config-wireless-policy)# no ip mac-binding
   ```

5. Enable the wireless policy profile.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

6. Return to privileged EXEC mode.

Example

```
Device(config-wireless-policy)# exit
```

IP MAC binding is disabled on the wireless policy profile, allowing clients to use IP addresses without MAC address restrictions.

Verify IP MAC binding

Verify IP MAC binding status to confirm whether it is enabled or disabled.

Procedure

Use the following command to verify if IP MAC binding is disabled or enabled:

Example

```
Device# show run | b wireless profile policy test-profile-policy
wireless profile policy test-profile-policy
 no ip mac-binding
 vlan VLAN0169
```

162 Disabling IP Learning in FlexConnect Mode

Topics:

- Disabling IP learning in FlexConnect mode

Guides you through the process of disabling Internet Protocol learning functionality when operating network devices in FlexConnect mode.

Disabling IP learning in FlexConnect mode

Explains how disabling IP learning in FlexConnect mode prevents IP theft errors and client exclusion by avoiding device tracking for clients with overlapping IP addresses.

A disabling IP learning in FlexConnect mode feature is a WLAN configuration option that

- prevents device tracking for clients in FlexConnect local switching scenarios
- avoids IP theft errors by ensuring no device tracking is done for clients, and
- supports overlapping IP address allocation across different sites.

IP address handling in FlexConnect local switching

In FlexConnect local switching scenarios, clients from the same sites may share the same address range, which can result in multiple clients being allocated or registered with the same IP address. The controller receives IP address information from the AP, and if more than one client attempts to use the same IP address, the controller discards the last device trying to register an already-used address as an IP theft event, potentially resulting in client exclusion.

The **no ip mac-binding** command ensures that no device tracking is done for clients, thus preventing the IP theft error.

Note

- This feature is applicable only for IPv4 addresses.
- Configuring **ip overlap** in FlexConnect Profile assists overlapping IP address support for clients across different sites in FlexConnect local switching.

Restrictions for disabling IP learning in FlexConnect mode

Outlines restrictions for disabling IP learning in FlexConnect mode.

- The **wireless client ip deauthenticate** command works by referring to the IP table binding entries directly. It does not work for client whose IPs are not learnt.
- Overlapping IP addresses within a single site tag and across different site tags require different settings. Furthermore, if a single site tag contains overlapping IP addresses, L3 web authentication is necessary. However, L3 web authentication relies on IP addresses, and ensuring the uniqueness of IP addresses cannot be guaranteed, making this combination incorrect.
- When IP Source Guard (IPSG) is enabled and multiple binding information is sent with the same IP and preference level (such as DHCP, ARP, and so on) to CPP, the CPP starts to ignore the later bindings after the first binding creation. Hence, you should not configure IPSG and disable IP MAC binding together. If IPSG and **no ip mac-binding** are configured together then IPSG does not work.

Disable IP learning in FlexConnect mode

Configure IP learning in FlexConnect mode using CLI commands.

Configure the wireless policy profile to disable IP learning in FlexConnect mode for improved client management.

Disabling IP learning in FlexConnect mode prevents the controller from learning client IP addresses through MAC binding, which can be useful in certain deployment scenarios where IP-MAC binding is not required or causes issues.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the wireless profile policy.

   Example

   ```
   Device(config)# wireless profile policy profile-policy-name
   ```

   Example

   ```
   Device(config)# wireless profile policy test-profile-policy
   ```

3. Disable the wireless policy profile.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

   Disabling the policy profile results in associated AP and client rejoining.

4. Disable IP learning in FlexConnect mode.

   Example

   ```
   Device(config-wireless-policy)# no ip mac-binding
   ```

5. Enable the wireless policy profile.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

6. Exit wireless policy configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# exit
   ```

IP learning is disabled in FlexConnect mode, and the wireless policy profile is active.

MAC entry verification from database

Provides instructions to verify MAC entries from the database using the appropriate command.

This reference explains how to verify MAC address details from the database using the relevant CLI command.

To verify the MAC details from database, use the following command:

```
Device# show wireless device-tracking database mac
MAC VLAN IF-HDL IP
-----------------------------------------------------------------------------------------------
6c96.cff2.889a 64 0x90000008 9.9.64.175
```

163 Disabling Device Tracking to Support NAC Devices

Topics:

- Disabling device tracking to support NAC devices

Outlines the considerations and procedures for disabling device tracking to ensure compatibility with Network Access Control devices.

Disabling device tracking to support NAC devices

Explains how disabling device tracking helps control the flow of traffic between wireless clients using a network access control (NAC) device.

A device tracking disablement is a network configuration feature that

- controls the flow of traffic between wireless clients using a network access control (NAC) device
- blocks direct traffic between wireless clients through ARP spoofing, and
- disables wireless client device tracking to support NAC devices.

Command usage for disabling device tracking

Use the **no ip mac-binding** command to enable ARP spoofing from the NAC. This command also disables wireless client device tracking.

> **Note**
>
> You can use this feature only with IPv4 addresses.

Feature history for disable device tracking support for NAC devices

Provides the feature history for disable device tracking support for NAC devices.

This reference provides the release and related information for the feature that disables device tracking to support NAC devices.

Table 143: Feature history for disabling device-tracking to support NAC devices

Release	Feature	Feature Information
Cisco IOS XE Cupertino 17.8.1	Disabling Device Tracking to Support NAC Devices	This feature helps to control the flow of traffic between wireless clients using network access control (NAC) device.

Restrictions for disabling device tracking to support NAC devices

Outlines restrictions for disabling device tracking to support NAC devices, including command limitations and unsupported Layer 3 policies. Review these constraints before configuration changes.

Restrictions for disabling device tracking to support NAC devices

- The **wireless client ip deauthenticate** command refers directly to the IP table binding entries. The command does not work if the client's IP address is not learned.
- Layer 3 web authentication and other L3 policies are not supported.
- When IP Source Guard (IPSG) is enabled and multiple binding information with the same address and preference level (such as DHCP or ARP) is sent to Cisco Packet Processor (CPP), the CPP ignores later bindings after creating the first binding. Do not configure IPSG and **no ip mac-binding** together. If you configure IPSG and **no ip mac-binding** together, IPSG does not work.

Disable device tracking for wireless clients (CLI)

Configure the system to disable device tracking for wireless clients using the CLI.

Disable device tracking for wireless clients to prevent the system from monitoring their IP and MAC address bindings.

Device tracking is used to monitor and bind IP and MAC addresses of wireless clients for security and management. Disabling this feature may be required in environments where such tracking is not needed or interferes with network operations.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the wireless profile policy.

   Example

   ```
   Device(config)# wireless profile policy profile-policy-name
   ```

   Example

   ```
   Device(config)# wireless profile policy test-profile-policy
   ```

3. Disable the wireless policy profile.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

   Disabling the policy profile results in associated AP and client to rejoin.

4. Disable the IP-MAC address binding.

   Example

   ```
   Device(config-wireless-policy)# no ip mac-binding
   ```

5. Enable the wireless policy profile.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

6. Exit wireless policy configuration mode.

   Example

   ```
   Device(config-wireless-policy)# exit
   ```

7. Configure a VLAN and enter VLAN configuration mode.

   Example

   ```
   Device(config)# vlan configuration vlan-id
   ```

   Example

   ```
   Device(config)# vlan configuration 20
   ```

8. Enable ARP broadcast on the VLAN.

   Example

   ```
   Device(config-vlan-config)# arp broadcast
   ```

9. Return to privileged EXEC mode.

   Example

   ```
   Device(config-vlan-config)# end
   ```

Device tracking is disabled for wireless clients, and ARP broadcast is enabled on the specified VLAN.

Verify ARP broadcast

Verify ARP broadcast to ensure that the feature is enabled and functioning as expected on the device. Use this procedure to confirm ARP broadcast status for troubleshooting or validation purposes.

Use this task to confirm whether ARP broadcast is enabled on your device. Confirming ARP broadcast helps ensure proper network communication and assists in troubleshooting connectivity issues.

Before you begin

Procedure

Use the **show platform software arp broadcast** command.

Example

```
Device# show platform software arp broadcast
Arp broadcast is enabled on vlans: 20,50
```

This command displays the ARP broadcast status and lists the VLANs where ARP broadcast is enabled.

View the ARP broadcast status and the VLANs on which it is enabled. If troubleshooting is required, proceed to the relevant troubleshooting task.

164 Disabling IP Learning in Local Mode

Topics:

- Disabling IP learning in local mode

Outlines the process and considerations for disabling Internet Protocol learning in local mode, highlighting its impact on network behavior and configuration.

Disabling IP learning in local mode

Explains how disabling IP learning in local mode prevents IP Theft events and client exclusion by stopping device tracking for clients.

A disabling IP learning in local mode feature is a WLAN security mechanism that

- uses the **no ip mac-binding** command to prevent device tracking for clients
- prevents IP Theft errors by ensuring that multiple clients do not register the same IP address, and
- allows downstream broadcast ARP traffic to reach wireless clients in the VLAN when ARP broadcast is enabled and IP MAC binding is disabled.

IP learning prevention in local mode scenarios

Describes how the controller handles IP address conflicts and ARP broadcast traffic in local mode.

Key points about IP learning prevention:

- In local mode central switching, multiple clients may have an allocated or registered IP address.
- If the controller detects more than one client using the same IP address, it discards one client as an IP Theft event, which may result in client exclusion.
- To allow downstream broadcast ARP traffic to reach wireless clients in the VLAN, enable ARP broadcast and disable IP MAC binding.
- The controller replicates broadcast ARP packets to all APs belonging to the controller when Multicast over Multicast (MOM) is disabled.
- To avoid replication, enable MOM.

> **Note**
>
> This feature is applicable only for IPv4 addresses.

Restrictions for disabling IP learning in local mode

Outlines key restrictions you must consider when disabling IP learning in local mode, including command limitations and policy incompatibilities.

Restrictions for disabling IP learning in local mode

- The **wireless client ip deauthenticate** command works by referring to the IP table binding entries directly. It does not work for clients whose IPs are not learnt.
- The L3 web authentication and other L3 policies are not supported.
- When IP Source Guard (IPSG) is enabled and multiple binding information is sent with the same IP and preference level (such as DHCP, ARP, and so on) to CPP, the CPP starts to ignore the later bindings after the first binding creation. Hence, you should not configure IPSG and disable IP MAC binding together. If IPSG and **no ip mac-binding** are configured together then IPSG does not work.

Disable IP learning in local mode (CLI)

Configure IP learning in local mode using CLI commands to control client IP address assignment and ARP broadcast behavior.

Configure the wireless policy profile to disable IP learning in local mode, ensuring proper client IP address handling and ARP broadcast support.

This task is performed when you need to prevent IP address learning for wireless clients in local mode, typically to support downstream ARP broadcast traffic and avoid MAC binding issues.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the wireless profile policy.

   Example

   ```
   Device(config)# wireless profile policy profile-policy-name
   ```

3. Disable the wireless policy profile.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

   Disabling the policy profile results in associated AP and client rejoining.

4. Disable IP learning in local mode.

   Example

   ```
   Device(config-wireless-policy)# no ip mac-binding
   ```

5. Enable the wireless policy profile.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

6. Exit wireless policy configuration mode.

   Example

   ```
   Device(config-wireless-policy)# exit
   ```

7. Configure a VLAN and enter VLAN configuration mode.

Example

```
Device(config-vlan-config)# vlan configuration vlan-id
```

To allow downstream broadcast ARP traffic to reach the wireless client in the VLAN, enable ARP broadcast and disable IP MAC binding.

8. Enable ARP broadcast on the VLAN.

Example

```
Device(config-vlan-config)# arp broadcast
```

9. Return to privileged EXEC mode.

Example

```
Device(config-vlan-config)# end
```

IP learning is disabled in local mode, allowing ARP broadcast traffic to reach wireless clients and preventing MAC binding issues.

Verify MAC entries database

Provides steps to verify MAC entries from the database.

Use the **show wireless device-tracking database mac** command to verify MAC address details stored in the device database.

To verify the MAC details from database, use the following command:

```
Device# show wireless device-tracking database mac
  MAC             VLAN  IF-HDL      IP

------------------------------------------------------------------------------------------

  6c96.cff2.889a  64    0x90000008  9.9.64.175
```

ARP broadcast verification

Provides information about how to verify ARP broadcast.

To verify the ARP broadcast, use the following command.

```
Device# show platform software arp broadcast
Arp broadcast is enabled on vlans:
20,50
```

165 Security-Enhanced Linux

Topics:

- Security-Enhanced Linux

Introduces Security-Enhanced Linux fundamentals and provides essential information about implementing mandatory access controls in Linux environments.

Security-Enhanced Linux

Describes the SELinux enhancement for Cisco IOS XE platforms.

Security-Enhanced Linux (SELinux) is a Linux kernel security solution that

- incorporates a strong, flexible Mandatory Access Control (MAC) architecture into Cisco IOS XE platforms
- provides enhanced mechanisms to enforce information separation based on confidentiality and integrity requirements, and
- confines user programs and system services to minimum privilege required to perform their assigned functionality.

SELinux functionality and modes

SELinux addresses threats of tampering and bypassing of application security mechanisms and enables the confinement of damage that malicious or flawed applications can cause.

SELinux enforces mandatory access control policies that confine user programs and system services to the minimum privilege required to perform their assigned functionality. This reduces or eliminates the ability of these programs and daemons to cause harm when compromised (for example, through buffer overflows or misconfigurations). This is a practical implementation of principle of least privilege by enforcing MAC on Cisco IOS XE platforms. This confinement mechanism works independently of the traditional Linux access control mechanisms. SELinux allows you to define policies to control the access from an application process to any resource object, thereby allowing for the clear definition and confinement of process behavior.

SELinux can operate in two modes when enabled on a system:

- **Permissive Mode**: In Permissive mode, SELinux does not enforce the policy, and only generates system logs for any denials caused by violation of the resource access policy. The operation is not denied, but only logged for resource access policy violation.
- **Enforcing Mode**: In Enforcing mode, the SELinux policy is enabled and enforced. The Enforcing mode denies resource access based on the access policy rules, and generates system logs.

SELinux is enabled in the Enforcing mode by default on supported Cisco IOS XE platforms. In the Enforcing mode, any system resource access that does not have the necessary allow policy is treated as a violation, and the operation is denied. The violating operation fails when a denial occurs, and system logs are generated. In Enforcing mode, the solution works in access-violation prevention mode.

> **Note**
>
> By default, SELinux is in the Enforcing mode.

Configure SELinux in the EXEC mode

Configure SELinux in the EXEC mode using CLI commands.

Enable and configure SELinux security policies in EXEC mode to enhance system security through access controls.

SELinux (Security-Enhanced Linux) provides mandatory access control security mechanisms. This configuration allows you to set SELinux policies in different operational modes including enforcing, permissive, or default settings.

Procedure

1. Enable privileged EXEC mode.

Example

```
Device> enable
```

Enter the password if prompted.

2. Enter global configuration mode.

Example

```
Device# configure terminal
```

3. Enable the internal commands of the network-based services.

Example

```
Device(config)# service internal
```

4. Exit from the global configuration mode.

Example

```
Device(config)# exit
```

5. Configure SELinux in the EXEC mode.

Example

```
Device# set platform software selinux {default | enforcing | permissive}
```

Example

```
Device# set platform software selinux enforcing
```

Available modes are default, enforcing, or permissive.

SELinux is now configured in the specified mode and will apply the appropriate security policies to system operations.

Configure SELinux in the global configuration mode

Configure SELinux policy in the global configuration mode using CLI commands.

This task enables you to set up SELinux security policies on the device to enhance system security through access controls.

SELinux (Security-Enhanced Linux) provides mandatory access controls that can be configured in either enforcing or permissive mode. Use this procedure when you need to establish or modify SELinux security policies on your device.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enable the internal commands of the network-based services.

Example

```
Device(config)# service internal
```

3. Configure SELinux policy.

Example

```
Device(config)# platform security selinux {enforcing | permissive}
```

```
Device(config)# platform security selinux enforcing
```

SELinux is now configured with the specified policy mode. The system will apply the selected security enforcement level to control access to system resources.

Examples for SELinux

Lists the examples for SELinux.

Provides examples of system log output when changing SELinux modes between Enforcing and Permissive states.

This example shows the output for changing the mode from Enforcing to Permissive:

```
"*Oct 20 21:44:03.609: %IOSXE-1-PLATFORM: R0/0: SELINUX_MODE_PROG: Platform
Selinux confinement mode downgraded to permissive!"
```

This example shows the output for changing the mode from Permissive to Enforcing:

```
"*Oct 20 21:44:34.160: %IOSXE-1-PLATFORM: R0/0: SELINUX_MODE_PROG: Platform
Selinux confinement mode upgraded to enforcing!"
```

Note

If you change the SELinux mode, the change is considered as a system security event, and a system log message is generated.

SELINUX syslog message reference

Lists the syslog message references for SELINUX.

Provides reference information for SELINUX syslog messages including message details, explanations, components, and recommended actions for troubleshooting security policy violations.

Facility-Severity-Mnemonic	**%SELINUX-1-VIOLATION**
Severity-Meaning	Alert Level Log
Message	N/A
Message Explanation	Resource access was made by the process for which a resource access policy does not exist. The operation was flagged, and resource access was denied. A system log was generated with information that process resource access has been denied.
Component	SELINUX
Recommended Action	Contact Cisco TAC with these relevant information as attachments: • The exact message as it appears on the console or in the system. • Output of the **show tech-support** command (text file). • Archive of the Btrace files from the box using the following command: **request platform software trace archive target** *URL* • Output of the **show platform software SELINUX** command. • Output of the **show platform software audit all | section exclude** command. • Output of the **show platform software audit summary** command.

These examples display sample syslog messages:

Example 1:

```
*Nov 14 00:09:04.943: %SELINUX-1-VIOLATION: R0/0: audispd: type=AVC
msg=audit(1699927057.934:129): avc: denied { getattr } for pid=5899 comm="ls"

path="/root/test" dev="rootfs" ino=25839
scontext=system_u:system_r:polaris_iosd_t:s0
tcontext=system_u:object_r:admin_home_t:s0 tclass=file permissive=0
```

Example 2:

```
*Nov 14 00:09:04.947: %SELINUX-1-VIOLATION: R0/0: audispd: t type=AVC
msg=audit(1699927198.486:130): avc: denied { write } for pid=6012 comm="echo"

path="/root/test" dev="rootfs" ino=25839
scontext=system_u:system_r:polaris_iosd_t:s0
tcontext=system_u:object_r:admin_home_t:s0 tclass=file permissive= 0
```

Verify count of denials

Provides command to verify the count of denials.

To verify the count of denials, use this command:

```
Device# show platform software audit summary
==================================
AUDIT LOG ON chassis 1 route-processor 0
----------------------------------
AVC Denial count: 6
```

Verify SELinux enablement

Lists the show command and its output for verifying SELinux enablement.

To verify SELinux enablement, use this command:

```
Device# show platform software selinux
=======================================
IOS-XE SELINUX STATUS
=======================================
SElinux Status : Enabled
Current Mode : Enforcing
Config file Mode : Enforcing
```

Part XI

Mobility

Topics:

- Mobility
- NAT Support on Mobility Groups
- Static IP Client Mobility
- Mobility Domain ID - 802.11i Roaming
- 802.11r Support for Flex Local Authentication
- Opportunistic Key Caching

166 Mobility

Topics:

- Mobility

Introduces the fundamental concepts and principles of network mobility, covering essential components and technologies that enable wireless connectivity and device movement across network infrastructures.

Mobility

Introduces mobility functionality including intracontroller, intercontroller, and intersubnet roaming concepts, SDA roaming capabilities, and mobility group configurations for wireless network management.

Mobility, or roaming, is the ability of a wireless LAN client to maintain a seamless and secure connection while moving from one access point to another with minimal latency.

- enables uninterrupted connectivity as clients move between access points,
- relies on controllers to manage client information, including MAC and IP addresses, security context, quality of service (QoS), wireless LAN (WLAN), and associated access point, and
- ensures secure and efficient data forwarding and traffic management for wireless clients.

Definitions of mobility-related terms

Key mobility terms in wireless LANs include point of attachment, point of presence, and station.

- Point of Attachment: the location in the network where a station data path is first processed when it enters the network.
- Point of Presence: the point in the network where a station is advertised.
- Station: a device that connects to and requests service from a network.

Intracontroller roaming

Intracontroller roaming refers to the process by which a wireless client moves its association from one access point to another within the same controller. The controller updates the client database and establishes new security context and associations when necessary, enabling seamless connectivity.

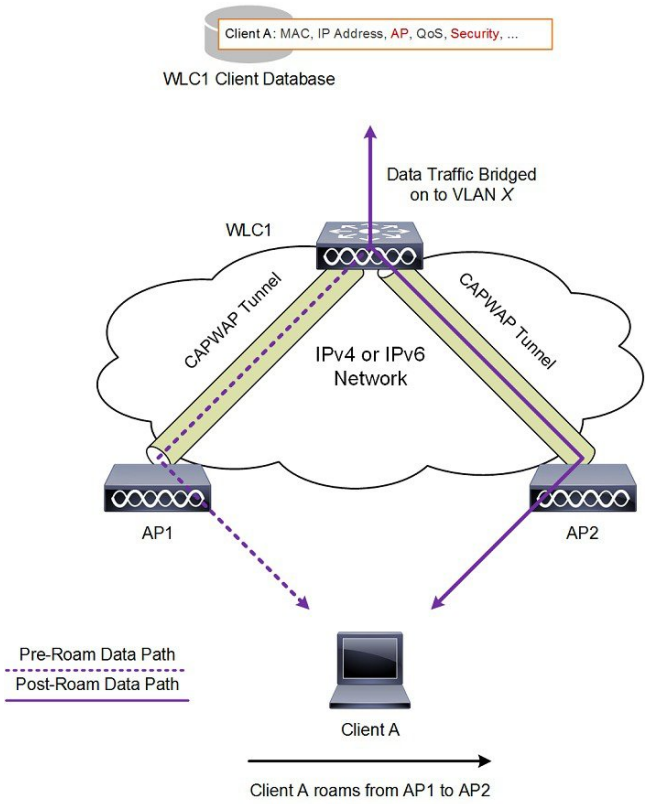
A intracontroller roaming is a wireless network mobility process that

- enables a wireless client to move its association from one access point to another within the same controller
- allows the controller to update the client database with the new access point association, and
- maintains or reestablishes the security context and client associations to support uninterrupted connectivity.

Additional information

When a client roams between access points managed by the same controller, the controller simply updates the client database to reflect the client's new association. If required, the controller creates new security contexts and reestablishes associations to maintain connectivity.

Figure 27: Intracontroller roaming



Intracontroller roaming example

A user walking through a building with multiple access points connected to one wireless LAN controller experiences no interruption in service because intracontroller roaming automatically updates the client's association information as the device connects to new access points.

Intercontroller roaming

This concept explains intercontroller roaming events, including how clients transition between controllers, how mobility messages function, security considerations, and important authentication and SDA limitations.

An intercontroller roaming event is a wireless mobility process that provides the following functions:

- occurs when a client moves from an access point managed by one controller to an access point managed by another controller
- exchanges mobility messages between the new and original controllers to update the client database entry, and
- maintains the user network experience, making the handover transparent to the user.

- Mobility messages: specialized communications between controllers that coordinate the handover and update the client database during a roaming event.
- Client database entry: the information each controller stores about a connected client, such as security context, session details, and access point association.

Expanded explanation

When a client device joins an access point managed by a new controller, the new controller exchanges mobility messages with the previous controller.

These messages transfer the client session information and database entry from the original controller to the new controller. The system establishes new security contexts and associations as needed.

The system updates the client database entry to reflect the new access point and controller location. This process occurs in the background and appears seamless to the user.

Note

Clients configured with 802.1X or Wi-Fi Protected Access (WPA) security complete full authentication to comply with IEEE standards.

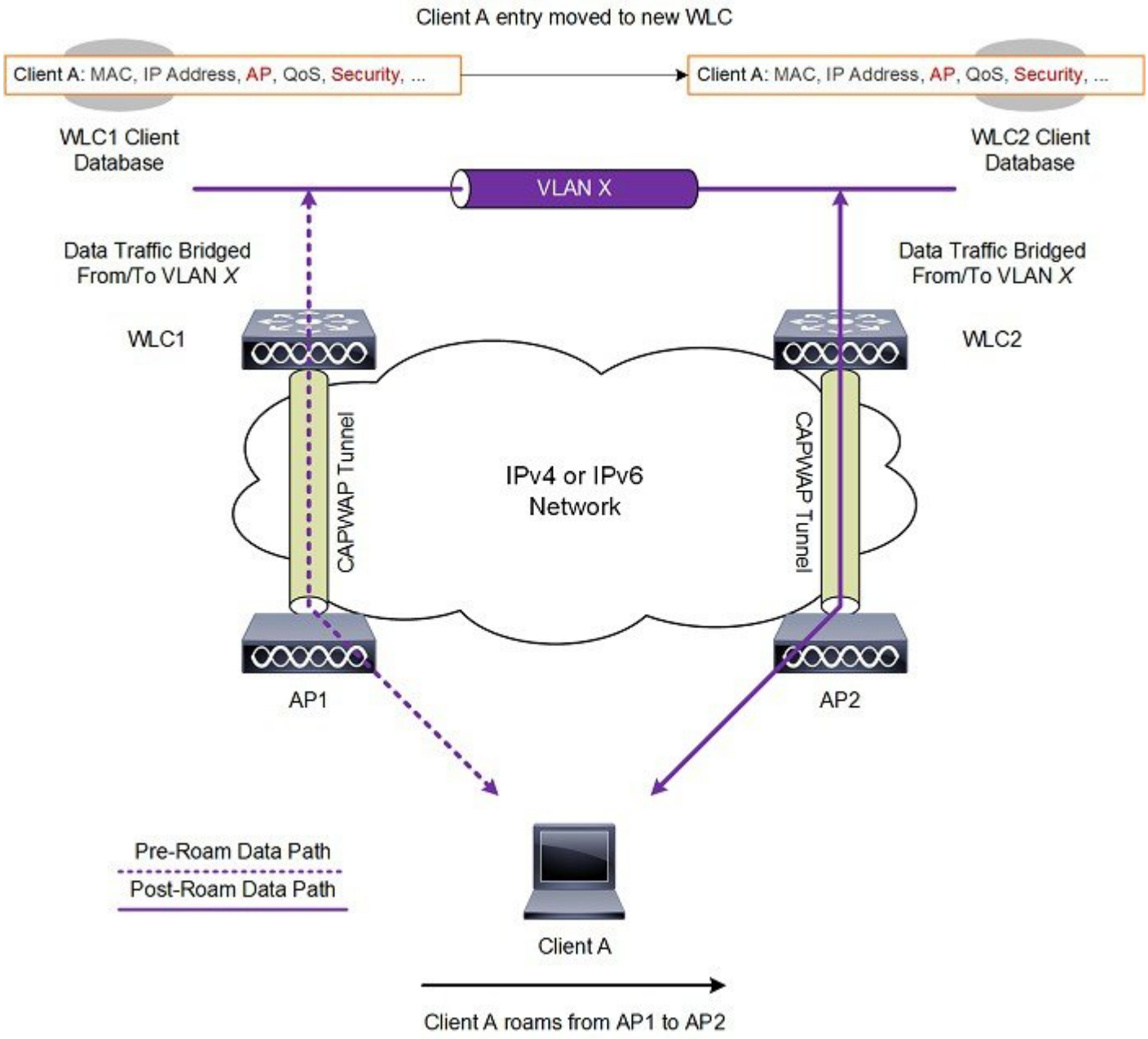
Important

Intersubnet roaming is not supported for Software-Defined Access (SDA).

Intercontroller roaming examples

A user with a mobile device walks from one building to another on a campus network. The device switches from an access point managed by Controller A to one managed by Controller B. The client remains connected and active while the controllers complete the roaming event.

The figure shows intercontroller roaming, which occurs when wireless LAN interfaces on controllers are on the same IP subnet.



Intersubnet roaming

Learn how intersubnet roaming works in wireless LAN environments, including controller roles, client mobility behavior, and required network conditions.

An intersubnet roaming event is a wireless network mobility process that provides the following functions:

- allows wireless clients to move seamlessly between controllers on different IP subnets
- keeps the original client IP address after roaming
- uses anchor and foreign controller roles to maintain and track the client session across controllers
- Anchor controller: the original controller where the client is first authenticated and where session anchoring remains, maintaining the primary client database entry.

- Foreign controller: the new controller to which the client connects after roaming, which receives a copy of the client database entry marked as a foreign entry.

Expanded explanation

Intersubnet roaming is similar to intercontroller roaming because controllers exchange mobility messages when a client roams. Instead of transferring the client database entry to the new controller, the original controller marks the client with an anchor entry in its client database.

The system then copies the database entry to the new controller and marks it as a foreign entry. This process keeps the roam transparent to the wireless client and preserves the original IP address.

WLANs on both the anchor and foreign controllers must provide the same network access privileges.

Ensure that source-based routing and source-based firewalls are not configured. If they are present, clients may experience network connectivity issues after the handoff.

Additional reference information

In static anchor setups that use controllers and a RADIUS server, the following applies:

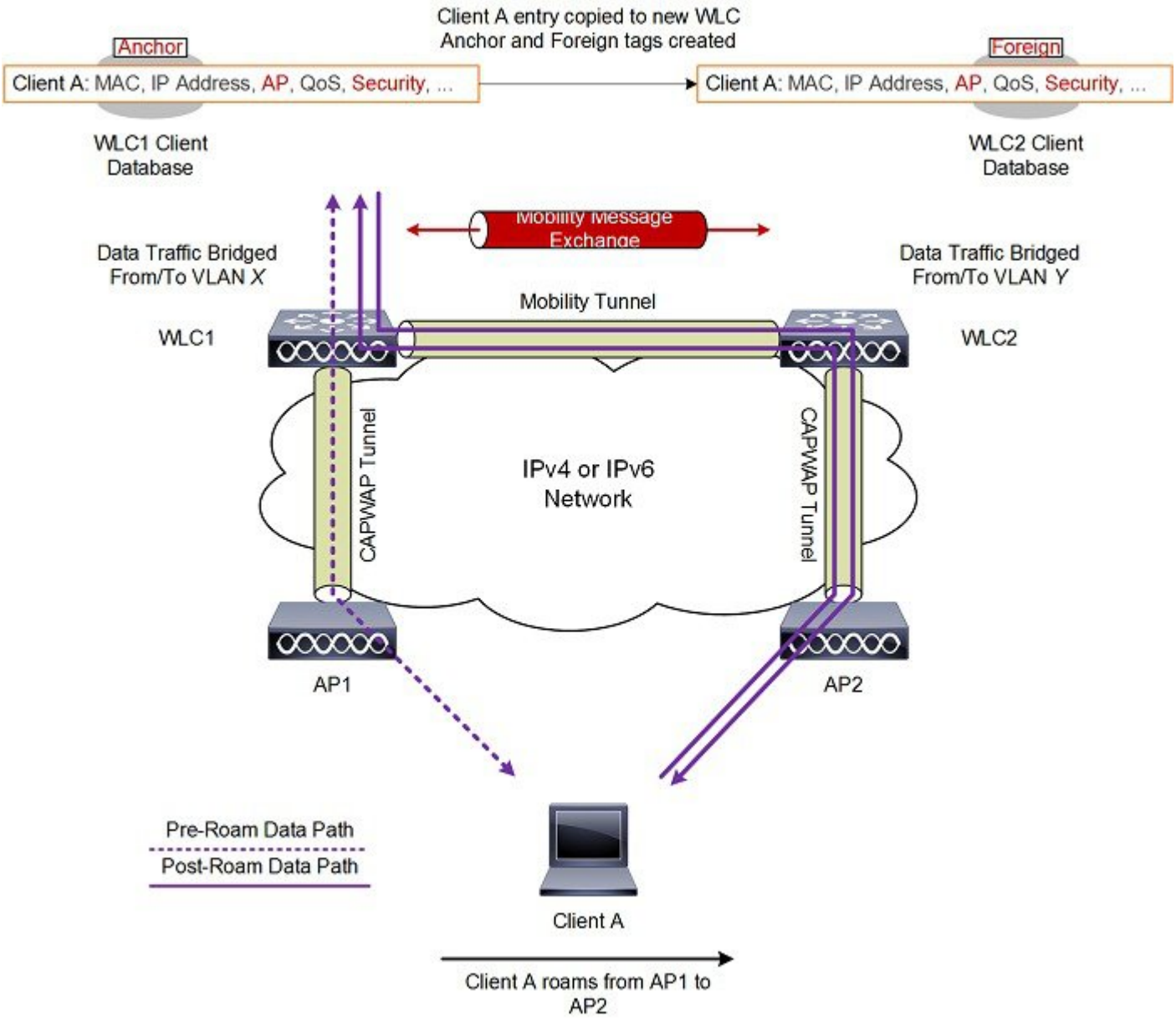
- If AAA override dynamically assigns VLAN and QoS, the foreign controller updates the anchor controller with the correct VLAN after Layer 2 authentication (802.1X).
- For Layer 3 RADIUS authentication, the anchor controller sends authentication requests.

> **Note**
>
> The Cisco Catalyst 9800 Series Wireless Controller mobility tunnel is a CAPWAP tunnel with a control path (UDP 16666) and a data path (UDP 16667). The control path uses DTLS encryption by default. You can enable DTLS on the data path when you add the mobility peer.
>
> The supported DTLS version is 1.2.

Intersubnet roaming example

This figure shows intersubnet roaming, which occurs when wireless LAN interfaces on controllers are located on different IP subnets.

Figure 28: Intersubnet roaming



SDA roaming

Learn how SDA roaming types keep wireless clients connected when they move between access points in a software-defined access (SDA) fabric. This section explains intra-xTR and inter-xTR roaming and how client location tables and data tables are updated during roaming events.

SDA roaming types keep wireless clients connected as they move across access points in a software-defined access (SDA) network. SDA roaming:

- keeps wireless clients connected within the fabric as they move across access points
- supports both intra-xTR events—within the same access switch—and inter-xTR events—across different access switches
- updates key system tables—local client database, client history table, and map server where applicable—for accurate client location tracking and ongoing connectivity
- SDA (software-defined access): An architecture that uses centralized control to automate and secure network operations across wired and wireless endpoints.

- xTR (tunnel router): In SDA, an xTR is an access switch that acts as both an ingress tunnel router and an egress tunnel router. It serves as a fabric edge node responsible for client mobility and traffic tunneling.

Expanded explanation

This section describes the two principal types of SDA roaming events:

- Intra-xTR roaming happens when your client moves between access points attached to the same access switch (xTR) in an SDA fabric-enabled WLAN. The system updates the local client database and client history table with the new access point information.
- Inter-xTR roaming happens when your client moves to an access point attached to another access switch (xTR) within the fabric. The system updates the local client database and the map server with the new location (RLOC) to show which switch the client is now using.

Note

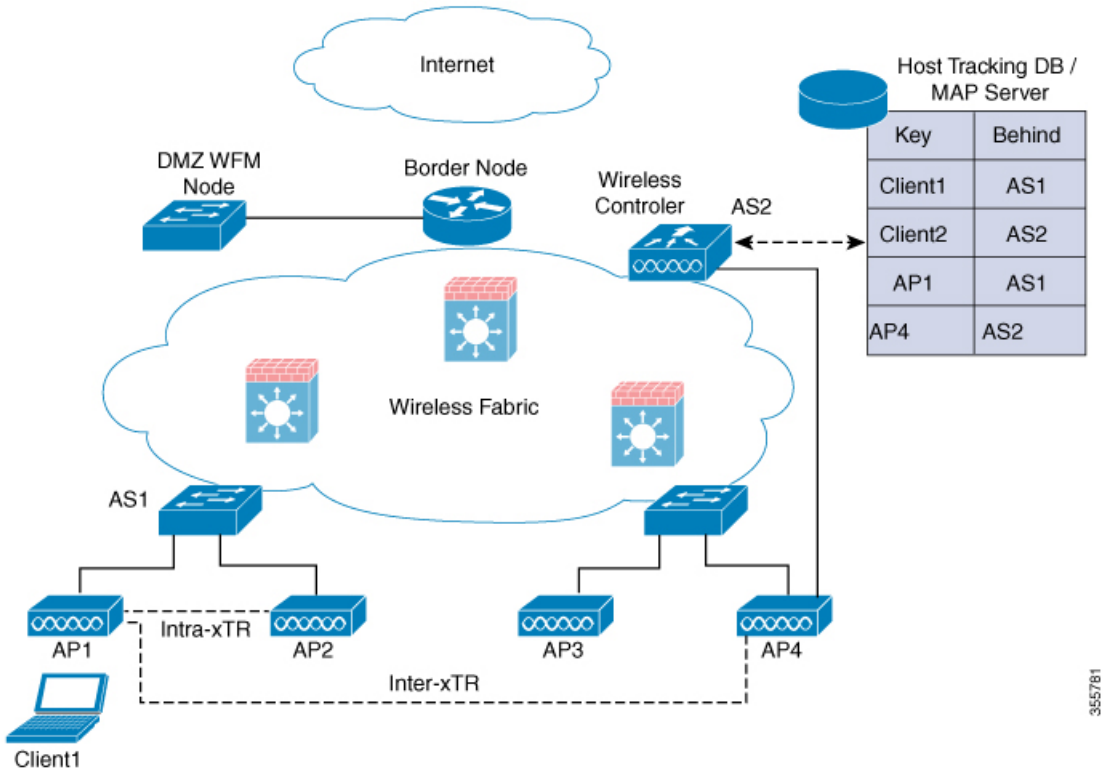
- During intra-xTR events, the access switch updates only its internal tables.
- During inter-xTR events, the switch and the map server update their tables so the client session continues without interruption.

SDA roaming example

When a wireless client roams from Access Point A to Access Point B, both connected to the same access switch, the client experiences an intra-xTR roaming event. Only local tables are updated.

When the client moves from Access Point A to Access Point C on another access switch, the system initiates an inter-xTR roaming event. It updates both the local client database and the central map server with the new location.

The figure shows intra-xTR and inter-xTR roaming. This occurs when the client moves between access points on the same switch or on different switches in a fabric topology.



Mobility groups

This concept explains how mobility groups enable seamless roaming for wireless clients by coordinating multiple controllers within the same network realm.

A mobility group is a wireless controller configuration that

- defines a set of controllers identified by the same mobility group name
- enables seamless roaming and state and context sharing for wireless clients, and
- allows controllers to dynamically forward data traffic and share the list of access points.

By creating a mobility group, multiple controllers in a network dynamically share information and forward data traffic when inter-controller or inter-subnet roaming occurs.

Controllers in the same mobility group share the context and state of client devices and their list of access points. This prevents controllers from identifying access points from other controllers as rogue devices. With this shared information, the network supports inter-controller wireless LAN roaming and controller redundancy.

When an access point (AP) moves from one controller to another, and both controllers are mobility peers, a client associated with controller 1 before the move might remain anchored to controller 1 after the move.

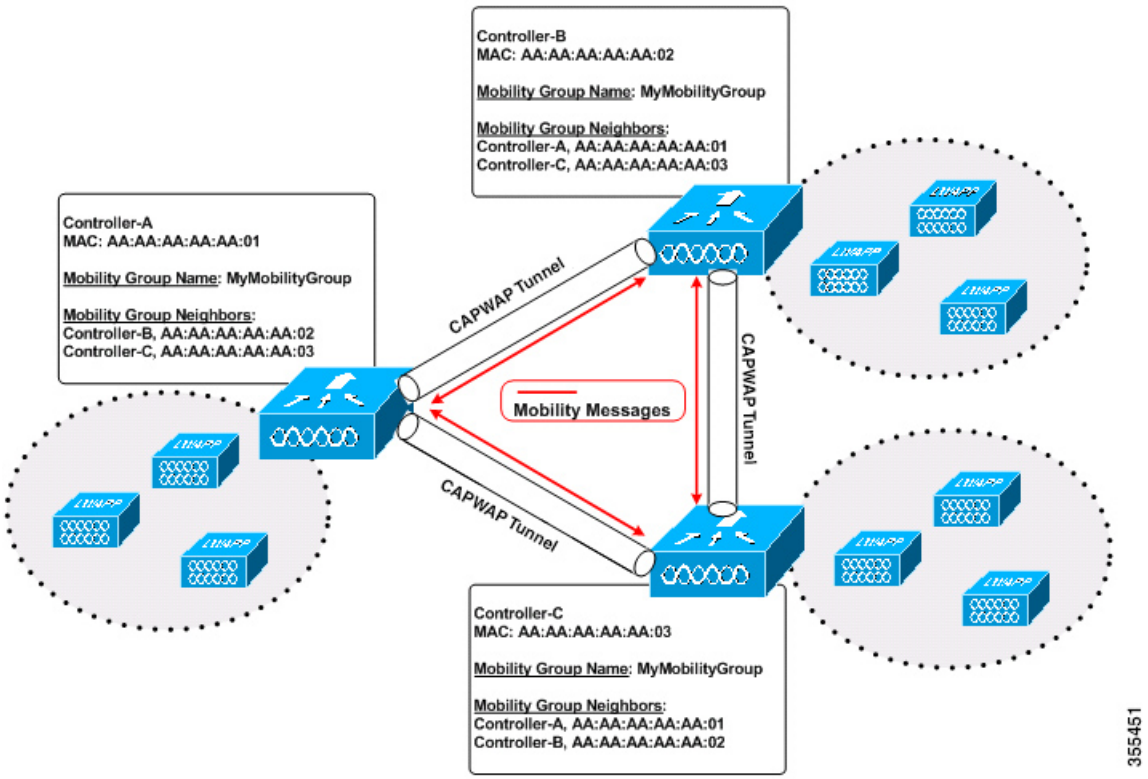
Controller 1 maintains the client entry for a timeout period to support roaming and reassociation scenarios. To prevent a client from remaining anchored to controller 1 after such a move, remove the mobility peer configuration from that controller.

> **Note**
>
> A mobility group can contain up to 24 members, while a mobility list can include up to 72 entries.

Example of a single mobility group

The figure shows that each controller is configured with a list of the other mobility group members.

Whenever a new client joins a controller, the controller sends a unicast message, or a multicast message if mobility multicast is configured, to all controllers in the mobility group. The controller that previously managed the client passes the client status to the new controller.



Supported platforms

Details platform support for mobility features, outlines guidelines and restrictions on mobility implementation, and specifies inter-release controller mobility supported platforms.

Lists the AireOS wireless controllers, Catalyst 9800 wireless controllers, and Catalyst switch platforms supported by the solution.

Review this list to confirm your device models are supported for integration and management.

Supported AireOS wireless controllers

- Cisco 3504 Wireless Controller
- Cisco 5520 Wireless Controller
- Cisco 8540 Wireless Controller

Supported Catalyst 9800 Wireless controllers

- Catalyst 9800-CL (Cloud Wireless Controller)
- Cisco Catalyst 9800-80 Wireless Controller
- Cisco Catalyst 9800-40 Wireless Controller
- Cisco Catalyst 9800-L Wireless Controller

Supported switch platforms

- **Catalyst switches**
  - Cisco Catalyst 9300 Switch
- **Catalyst 9800 controllers**
  - Catalyst 9800-CL (Cloud Wireless Controller)
  - Cisco Catalyst 9800-40 Wireless Controller

Guidelines and restrictions

This topic explains the guidelines and restrictions for mobility, including supported platforms, configuration requirements, and operational considerations for SDA Inter-Controller Mobility.

SDA Inter-Controller Mobility enables seamless client roaming across compatible Cisco wireless controllers and switch platforms. This topic describes supported hardware, operational requirements, and restrictions.

- Data DTLS Configuration: Ensure that the data DTLS configuration on Catalyst 9800 and AireOS controller platforms is identical. If the configurations do not match, the system disables the mobility data path.
- Layer 3 Roaming: In intercontroller roaming scenarios, policy profiles with different VLANs are supported for Layer 3 roaming.
- L3 Override in Guest VLANs: L3 override is not supported in guest VLANs on AireOS controllers. Trigger DHCP Discovery manually on the new VLAN because clients do not trigger it automatically.
- Policy Profile and VLAN Flexibility: Controllers that use the same WLAN profile can have different policy profile names and client VLANs.
- Intracontroller Roaming: Roaming is supported between identical policy profiles with the WLAN mapped within a single controller.

  Starting with Cisco IOS XE Amsterdam 17.3.x, seamless roaming is supported between the same WLAN associated with different policy profiles. For more information, see the Client Roaming Policy Profile feature.
- Web Authentication Roaming: If a client roams while in web authentication state, the system treats the client as a new client on the other controller rather than as a mobile client.
- DHCP Server Consistency: Mobility peer controllers must use the same DHCP server to maintain client mobility move counts on intra-VLAN.
- Data DTLS and Server Security Chip (SSC) Hash Key: Data DTLS and SSC hash key values must be identical for mobility tunnels established between controller members.
- Mobility Move Count: The system updates the move count under client details only during inter-controller roaming. For intra-controller roaming, verify the value under client statistics and mobility history.
- VLAN Mapping Differences: The Anchor VLAN on Catalyst 9800 controllers corresponds to the Access VLAN on Cisco AireOS controllers.

- Mobility Role State for Roaming Clients: Roaming clients may appear with the *Unknown* mobility role because clients in the *IP learn* state can experience frequent additions and deletions.
- WLAN Profile Mismatch: In inter-controller roaming between Catalyst 9800 and Catalyst 9800 or AireOS controllers, client roaming is not supported if there is a WLAN profile mismatch.
- IPv4 Tunnel Only: Only IPv4 tunnels are supported between Catalyst 9800 and Cisco AireOS controllers.
- High Availability Configuration: In High Availability scenarios, configure the mobility MAC address explicitly by using the `wireless mobility mac-address` command to ensure that the mobility tunnel remains up after Stateful Switchover (SSO).
- Certificate and Trustpoint Limitation: Use RSA-based certificates or RSA-based trustpoints for wireless management. Mobility tunnels do not support ECDSA-based certificates or ECDSA-based trustpoints.
- Layer 2 and Layer 3 Topology Warning: Connecting Anchor controllers and Foreign controllers in the same Layer 2 network and over a Layer 3 mobility tunnel creates a loop topology. This configuration may trigger a MAC_CONFLICT warning message each minute. The warning does not impact functionality or performance. Use a VLAN other than the management VLAN as the client VLAN.
- Tunnel Reset and Stateful Switchover (SSO): The mobility tunnel reestablishes if Stateful Switchover (SSO) is triggered due to a gateway check failure.
- AP Slot 2 and Neighbor Reporting: If the current AP has a 5 GHz slot 2 radio operating on Layer 2 or Layer 3 mobility, the WLAN BSSID is added only to the 802.11k or 802.11v neighbor information. When radio property information from APs on other controllers is unavailable, the AP uses its own radio properties.

  If the AP does not have a slot 2 radio, other APs cannot be added as neighbors. If validation fails, the system does not add that radio to the neighbor list.
- Keepalive Values Recommendation: Use the default keepalive count and interval values to reduce convergence time when setting up mobility tunnels between AireOS and Catalyst 9800 controllers.
- Client Join Times: When the mobility tunnel is up and mobility peers are configured, a new client may take up to 3 seconds to join. The system sends 3 mobile messages, 1 second apart, to verify whether the client is already part of the network.

Inter-release controller mobility supported platforms

This section lists the platforms, software versions, feature compatibility, and restrictions for Inter-Release Controller Mobility (IRCM) on Cisco Wireless Controllers.

The Inter-Release Controller Mobility (IRCM) feature is supported on these Cisco Wireless Controllers.

Cisco Catalyst 9800 series

- All Cisco Catalyst 9800 Series Wireless Controller platforms running Cisco IOS XE Software version 16.10.1 or later.
- By design, Cisco Catalyst 9800 Wireless Controllers do not include the Primary Mode configuration in the Discovery Response. The controller always sends the Discovery Response with Primary Mode enabled.

Cisco AireOS wireless controllers

- Supported AireOS Wireless Controllers running the AireOS 8.5.14x.x IRCM image based on the 8.5 maintenance release software include:
  - Cisco 3504 Wireless Controller
  - Cisco 5508 Wireless Controller
  - Cisco 5520 Wireless Controller

  - Cisco 8510 Wireless Controller
  - Cisco 8540 Wireless Controller
- Supported AireOS Wireless Controllers running AireOS 8.8.111.0 and later include:
  - Cisco 3504 Wireless Controller
  - Cisco 5520 Wireless Controller
  - Cisco 8540 Wireless Controller

Unsupported AireOS wireless controllers

You cannot use the IRCM feature on these Cisco AireOS Wireless Controllers:

- Cisco 2504 Wireless Controller
- Cisco Flex 7510 Wireless Controller
- Cisco WiSM 2

Recommended configuration for IRCM deployments

- Configure both Cisco AireOS and Cisco Catalyst 9800 Series Controllers as static RF leaders to avoid RF grouping between them.
- Configure the same RF network name on both controllers.

AireOS 9800 virtual controllers: hash command and data tunnel encryption

- If the peer Cisco Catalyst 9800 Series Wireless Controller is virtual, configure the hash using this command:

```
config mobility group member hash 172.20.227.73
3f93a86cee2039e9c3aada1822ad74b89fea30c1
```

- Enable or disable data tunnel encryption using this command:

```
config mobility group member data-dtls 00:0c:29:a8:d5:77 enable or disable
```

- Run this command on the Cisco Catalyst 9800 Series Wireless Controller to obtain the hash:

```
show wireless management trustpoint
Trustpoint Name : ewlc-tp1
Certificate Info : Available
Certificate Type : SSC
Certificate Hash : 3f93a86cee2039e9c3aada1822ad74b89fea30c1
Private key Info : Available
```

Platform and feature limitations

- IPv6 is not supported for Software Defined Access (SDA) IRCM fabric client roaming.
- IPv6 is supported for nonfabric IRCM client roaming.
- Ensure that AireOS controllers support the Encrypted Mobility feature.
- Application Visibility and Control (AVC) is not supported for IRCM.
- In mixed deployments that include Catalyst 9800 and AireOS controllers, use the same name for the WLAN profile and the policy profile. AireOS does not recognize the policy profile and uses the WLAN name for both profiles.
- Mobility group multicast is not supported because AireOS does not support mobility multicast in encrypted mobility.

- Disable Link Local bridging on the peer AireOS controller.
- IRCM is not supported in FlexConnect or FlexConnect+Bridge modes.
- If the client roaming rate is very high, the displayed client count may exceed the supported roaming scale. This occurs because the system requires time to update records. A client that appears on multiple WNCDs for a short time may be counted more than once. Wait for the system to stabilize before you verify client counts using show CLIs, WebUI, Cisco DNA Center, or SNMP.

IPv6 feature support between controllers

The following features support IPv6 client mobility between AireOS controllers and Catalyst 9800 controllers:

- Accounting
- Layer 3 security (Webauth)
- Policy (ACL and QoS)
- IP address assignment and learning through SLAAC and DHCPv6
- IPv6 Source Guard
- Multiple IPv6 address learning
- IPv6 multicast
- SISF IPv6 features include:
  - RA Guard
  - RA Throttling
  - DHCPv6 Guard
  - ND Suppress

These IPv6 features are not supported on Catalyst 9800 controllers:

- Configurable IPv6 timers
- RA Guard enabled on access points
- Global IPv6 disable
- IPv6 Central Web Authentication (CWA)
- More than eight IPv6 addresses per client

Configure the mobility feature (GUI)

Provides instructions for configuring mobility through GUI and CLI interfaces, including inter-release controller mobility setup and mobility peer configuration procedures.

Configure wireless controller mobility features to support seamless roaming and redundancy.

Procedure

1. Choose **Configuration** > **Wireless** > **Mobility**.
   The **Wireless Mobility** page appears, where you can perform global configuration and peer configuration.
2. In the **Global Configuration** section, perform these tasks:

a) Enter a name for the mobility group.
b) Enter the multicast IP address for the mobility group.
c) In the **Keep Alive Interval** field, specify how many times to send a ping request to a mobility list member before marking the member unreachable. The valid range is 3 to 20 seconds, and the default value is 3 seconds.
d) Specify the time, in seconds, between ping requests to a mobility list member in the **Mobility Keep Alive Count** field. The valid range is 1 to 30 seconds.
e) Enter the Differentiated Services Code Point (DSCP) value for the mobility group.
f) Enter the mobility MAC address.
g) Click **Apply**.

3. In the **Peer Configuration** tab, perform these tasks:
   a) In the **Mobility Peer Configuration** section, click **Add**.
   b) In the **Add Mobility Peer** window that is displayed, enter the MAC address and IP address for the mobility peer. The MAC address can be in *xx:xx:xx:xx:xx:xx*, *xx-xx-xx-xx-xx-xx*, or *xxxx.xxxx.xxxx* format.
   c) If NAT is used, enter the optional public IP address for the NAT-ed address of the mobility peer. If NAT is not used, the device displays the direct IP address of the mobility peer.
   d) Enter the mobility group to which you want to add the mobility peer.
   e) Select the required status for **Data Link Encryption**.
   f) Specify the **SSC Hash** as required.

      Secure Socket Communications (SSC) hash is required if the peer is a Cisco Catalyst 9800-CL Wireless Controller, which uses a self-signed certificate. In this case, the SSC hash provides additional validation. SSC hash is not required if the peer is an appliance because appliances have manufacturing installed certificates (MIC) or device certificates installed in the hardware.

   g) Click **Save & Apply to Device**.
   h) In the **Non-Local Mobility Group Multicast Configuration** section, click **Add**.
   i) Enter the mobility group name.
   j) Enter the multicast IP address for the mobility group.
   k) Click **Save**.

The controller applies the mobility group configuration and peer configuration, enabling mobility features for wireless devices.

Configure the mobility group (CLI)

Learn how to configure a mobility group and related parameters using CLI commands on your device.

Use this task to configure a mobility group and related parameters using CLI commands on a network device.

Procedure

---

1. Create a mobility group named **`Mygroup`**.

   Example

   ```
   Device(config)# wireless mobility group name Mygroup
   ```

2. Configure the MAC address used in mobility messages.

   Example

   ```
   Device(config)# wireless mobility mac-address mac-addr
   ```

3. Optional: Configure the mobility intercontroller DSCP value.

   Example

   ```
   Device(config)# wireless mobility dscp value-0-to-63
   ```

4. Optional: Configure the interval between two keepalives sent to a mobility member.

   Example

   ```
   Device(config)# wireless mobility group keepalive interval time-in-seconds
   ```

   The valid range is from 1 to 30 seconds.

   > **Note**
   >
   > For controllers connected through mobility tunnels, ensure that both controllers use the same keepalive interval value.

5. Optional: Configure the number of keepalive retries before a member status is set to DOWN.

   Example

   ```
   Device(config)# wireless mobility group keepalive count count
   ```

6. Add a peer IPv4 or IPv6 address to a specific group. Use one of the following options to configure IPv4 or IPv6:

   - **wireless mobility mac-address** *mac-address* **ip** *peer-ip-address* **group** *group-name* **data-link-encryption**
   - **wireless mobility mac-address** *mac-address* **ip** *peer-ip-address* **public-ip** *public-ip-address* **group** *group-name*

   Example

   ```
   Device(config)# wireless mobility mac-address 001E.BD0C.5AFF ip 9.12.32.10
   group test-group data-link-encryption
   ```

   ```
   Device(config)# wireless mobility mac-address 001E.BD0C.5AFF ip fd09:9:2:49::55
    public-ip fd09:9:2:49::55 group scalemobility
   ```

   Use the **no** form of this command to remove the peer from the local group.

7. Configure a multicast IPv4 or IPv6 address for a local or nonlocal mobility group.

   - **wireless mobility multicast** { **ipv4** | **ipv6** } *ip-address*
   - **wireless mobility group multicast-address** *group-name* { **ipv4** | **ipv6** } *ip-address*

> **Note**
>
> **Mobility Multicast** – The controller sends a multicast message instead of a unicast message to all members in the local or nonlocal mobility group when a client joins or roams.

Example

```
Device(config)# wireless mobility multicast ipv4 224.0.0.4
```

This command sets the multicast IPv4 address to 224.0.0.4 for a local mobility group.

Example

```
Device(config)# wireless mobility group multicast-address Mygroup ipv4
224.0.0.5
```

This command sets the multicast IPv4 address to 224.0.0.5 for a nonlocal mobility group.

The mobility group and related parameters are configured. Mobility services are enabled across controllers in the network.

Configure inter-release controller mobility (GUI)

Learn how to configure inter-release controller mobility using the graphical user interface (GUI) to set mobility group parameters and apply the configuration.

Configure inter-release controller mobility so devices can move between controllers running different software releases.

Configure this feature when you want to support mobility between controllers that run different software versions in your wireless infrastructure.

Procedure

---

1. Choose **Configuration** > **Wireless** > **Mobility** > **Global Configuration**.
2. Enter values for the following fields: **Mobility Group Name**, **Multicast IPv4 Address**, **Multicast IPv6 Address**, **Keep Alive Interval (sec)**, **Mobility Keep Alive Count**, **Mobility DSCP Value**, and **Mobility MAC Address**.
3. Click **Apply**.

Inter-release controller mobility is now enabled.

Configure inter-release controller mobility (CLI)

Learn how to configure inter-release controller mobility (IRCM) to enable seamless mobility and wireless services between controllers running different software releases, such as Cisco AireOS and Cisco IOS XE.

Inter-Release Controller Mobility (IRCM) provides features that allow controllers running different software releases to work together.

IRCM enables seamless mobility and wireless services between controllers running Cisco AireOS and Cisco IOS XE. For example, you can configure mobility between a Cisco 8540 WLC and Cisco Catalyst 9800 Series Wireless Controller. Supported features include Layer 2 roaming, Layer 3 roaming, guest access, and guest termination.

> **Note**
>
> To configure IRCM for different combinations of AireOS and Catalyst 9800 controllers, see the Cisco Catalyst 9800 Wireless Controller-AireOS IRCM Deployment Guide.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Add a peer IPv4 or IPv6 address to a specific group. You can configure IPv4 or IPv6 using one of the following options:

   - **wireless mobility group member mac-address** *mac-address* **ip** *peer-ip* **group** *group-name* **data-link-encryption**
   - **wireless mobility group member mac-address** *mac-address* **ip** *peer-ip-address* **public-ip** *public-ip-address* **group** *group-name*

   Example

   ```
   Device(config#) wireless mobility group member mac-address 001E.BD0C.5AFF ip
    9.12.32.10 group test-group data-link-encryption
   ```

   ```
   Device(config#) wireless mobility group member mac-address 001E.BD0C.5AFF ip
    fd09:9:2:49::55 public-ip fd09:9:2:49::55 group scalemobility
   ```

   Use the **no** form of this command to remove the peer from the local group.

3. Add a name for the local group.

   Example

   ```
   Device(config#) wireless mobility group name group-name
   ```

   The default local group name is "default".

4. Optional: Configure the MAC address used in mobility messages.

   Example

   ```
   Device(config#) wireless mobility mac-address mac-address
   ```

5. Add a peer to the local group.

Example

```
Device(config#) wireless mobility group member ip peer-ip
```

Use the **no** form of this command to remove the peer from the local group.

6. Optional: Configure the Differentiated Services Code Point (DSCP) value.

Example

```
Device(config#) wireless mobility dscp dscp-value
```

The default value is 48.

7. Configure the mobility control and data path keepalive count.

Example

```
Device(config#) wireless mobility group keepalive count count
```

The default value is 3.

8. Configure the mobility control and data path keepalive interval.

Example

```
Device(config#) wireless mobility group keepalive interval interval
```

The default value is 10.

> **Note**
>
> Ensure that both controllers use the same keepalive interval value when they are connected through mobility tunnels.

After you configure IRCM, wireless client mobility and service continuity are enabled across Cisco AireOS and Cisco IOS XE controllers in the specified mobility group.

Verify mobility

Explains monitoring procedures and commands for tracking mobility operations and performance across wireless network infrastructure.

To display the summary of the mobility manager, use this command:

```
Device# show wireless mobility summary
```

To display mobility peer information, use this command:

```
Device# show wireless mobility peer ip 10.0.0.8
```

To display the list of access points known to the mobility group, use this command:

```
Device# show wireless mobility ap-list
```

To display statistics for the mobility manager, use this command:

```
Device# show wireless stats mobility
Mobility event statistics:
     Joined as
         Local                          : 0
         Foreign                        : 0
         Export foreign                 : 2793
         Export anchor                  : 0
     Delete
         Local                          : 2802
         Remote                         : 0
     Role changes
         Local to anchor                : 0
         Anchor to local                : 0
     Roam stats
         L2 roam count                  : 0
         L3 roam count                  : 0
         Flex client roam count         : 0
         Inter-WNCd roam count          : 0
         Intra-WNCd roam count          : 0
         Remote inter-cntrl roam count  : 0
         Remote WebAuth pending roams   : 0
     Anchor Request
         Sent                           : 0
           Grant received               : 0
           Deny received                : 0
         Received                       : 0
           Grant sent                   : 0
           Deny  sent                   : 0
     Handoff Status Received
         Success                        : 0
         Group mismatch                 : 0
         Client unknown                 : 0
         Client blacklisted             : 14
         SSID mismatch                  : 0
         Denied                         : 0
     Handoff Status Sent
         Success                        : 0
         Group mismatch                 : 0
         Client unknown                 : 0
         Client blacklisted             : 0
         SSID mismatch                  : 0
         Denied                         : 0
     Export Anchor
         Request Sent                   : 2812
         Response Received              :
             Ok                         : 2793
             Deny - generic             : 19
             Client blacklisted         : 0
             Client limit reached       : 0
             Profile mismatch           : 0
             Deny - unknown reason      : 0
         Request Received               : 0
         Response Sent                  :
             Ok                         : 0
             Deny - generic             : 0
```

```
             Client blacklisted        : 0
             Client limit reached      : 0
             Profile mismatch          : 0
MM mobility event statistics:
     Event data allocs                 : 17083
     Event data frees                  : 17083
     FSM set allocs                    : 2826
     FSM set frees                     : 2816
     Timer allocs                      : 8421
     Timer frees                       : 8421
     Timer starts                      : 14045
     Timer stops                       : 14045
     Invalid events                    : 0
     Internal errors                   : 0
     Delete internal errors            : 0
     Roam internal errors              : 0


MMIF mobility event statistics:
     Event data allocs                 : 17088
     Event data frees                  : 17088
     Invalid events                    : 0
     Event schedule errors             : 0
     MMIF internal errors:
       IPC failure                     : 0
       Database failure                : 0
       Invalid parameters              : 0
       Mobility message decode failure : 0
       FSM failure                     : 0
       Client handoff success          : 0
       Client handoff failure          : 14
       Anchor Deny                     : 0
       Remote delete                   : 0
       Tunnel down delete              : 0
       MBSSID down                     : 0
       Unknown failure                 : 0
```

To display counters for all messages in mobility, use this command:

```
Device# show wireless stats mobility messages
MM datagram message statistics:
  Message Type               Built  Tx    Rx    Processed  Tx Error  Rx Error
  Forwarded  Retry  Drops  Allocs  Frees

_____________________________________________________________________________


  Mobile Announce             0      0     0     0          0         0
  25350      5624   0      2826    2826
  Mobile Announce Nak         0      0     0     0          0         0
  0          0      0      0       0
  Static IP Mobile Annc       0      0     0     0          0         0
  0          0      0      0       0
  Static IP Mobile Annc Rsp   0      0     0     0          0         0
  0          0      0      0       0
  Handoff                     0      0     14    14         0         0
  0          0      0      42      42
  Handoff End                 0      0     0     0          0         0
  2783       0      0      2783    2783
  Handoff End Ack             0      0     2783  2783       0         0
  0          0      0      8349    8349
  Anchor Req                  0      0     0     0          0         0
  0          0      0      0       0
```

```
Anchor Grant                0      0     0     0          0          0
0          0      0      0        0
Anchor Xfer                 0      0     0     0          0          0
0          0      0      0        0
Anchor Xfer Ack             0      0     0     0          0          0
0          0      0      0        0
Export Anchor Req           0      0     0     0          0          0
2812       0      0      2812     2812
Export Anchor Rsp           0      0     2812  2812       0          0
0          0      0      8436     8436
AAA Handoff                 0      0     0     0          0          0
0          0      0      0        0
AAA Handoff Ack             0      0     0     0          0          0
0          0      0      0        0
IPv4 Addr Update            0      0     2792  0          0          0
0          0      0      2792     2792
IPv4 Addr Update Ack        2792   2792  0     0          0          0
0          0      0      2792     2792
IPv6 ND Packet              0      0     0     0          0          0
0          0      0      0        0
IPv6 Addr Update            0      0     5587  0          0          0
0          0      0      5587     5587
IPv6 Addr Update Ack        5587   5587  0     0          0          0
0          0      0      5587     5587
Client Add                  0      0     0     0          0          0
0          0      0      0        0
Client Delete               0      0     0     0          0          0
0          0      0      0        0
AP List Update              25585  25585 8512  8512       2          1
0          0      0      34098    34098
Client Device Profile Info  0      0     0     0          0          0
0          0      0      0        0
PMK Update                  0      0     0     0          0          0
0          0      0      0        0
PMK Delete                  0      0     0     0          0          0
0          0      0      0        0
PMK 11r Nonce Update        0      0     0     0          0          0
0          0      0      0        0
Device cache Update         0      0     0     0          0          0
0          0      0      0        0
HA SSO Announce             0      0     0     0          0          0
0          0      0      0        0
HA SSO Announce Resp        0      0     0     0          0          0
0          0      0      0        0
Mesh Roam Request           0      0     0     0          0          0
0          0      0      0        0
Mesh Roam Response          0      0     0     0          0          0
0          0      0      0        0
Mesh AP PMK Time Upd        0      0     0     0          0          0
0          0      0      0        0
Mesh AP PMK Time Upd Ack    0      0     0     0          0          0
0          0      0      0        0
Mesh AP Channel List        0      3     1     0          0          1
0          0      0      2        2
Mesh AP Channel List Resp   0      0     0     0          0          0
0          0      0      0        0
AP upgrade                  0      0     0     0          0          0
0          0      0      0        0
Keepalive Ctrl Req          34080  34080 17031 17031      0          0
0          0      0      51111    51111
Keepalive Ctrl Resp         17031  17031 34067 34067      0          0
0          0      0      51098    51098
```

```
  Keepalive Data Req/Resp  238527  238527  221451  221451  0         0
  0          0      0      459978  459978
```

To display mobility information of the client, use this command:

```
Device# show wireless client mac-address 00:0d:ed:dd:35:80 detail
```

To display roaming history of the active client in the subdomain, use this command:

```
Device# show wireless client mac-address 00:0d:ed:dd:35:80 mobility history
```

To display client-specific statistics for the mobility manager, use this command:

```
Device# show wireless client mac-address 00:0d:ed:dd:35:80 stats mobility
```

To verify whether intercontroller roam is successful, use these commands:

- **show wireless client mac** *mac-address* **detail**: (on the roamed-to Controller) Displays the roam type as L2 and the roam count is incremented by 1.
- **show wireless client summary** : (on the roamed-from controller) The client entry will not be there in the output.

Verify SDA Mobility

To verify whether intracontroller, intra-xTR roam is successful, use these commands:

- **show wireless client summary**: Displays the new AP if the client has roamed across the APs on the same xTR.
- **show wireless client mac** *mac-address* **detail**: Displays the same RLOC as before the roam.

To verify whether intracontroller, inter-xTR roam is successful, use these commands:

- **show wireless fabric client summary**: Displays the new AP if the client has roamed across the APs on a different xTR.
- **show wireless client mac** *mac-address* **detail**: Displays the RLOC of the new xTR to which the client has roamed to.

To check client status before and after intracontroller roaming, perform these steps:

1. Check if client is on the old AP, using **show wireless client summary** command on the controller.
2. Check whether the client MAC is listed against the old AP, using **show mac addr dyn** command on the xTR1.
3. Check whether the client IP is registered from current xTR1, and client MAC is registered from both current xTR1, and WLC1, using **show lisp site detail** command on the MAP server.
4. After the intra-WLC roam, check whether the client is on the new AP, using the **show wireless client summary** and **show mac addr dyn** commands on the WLC1 and xTR1.
5. After the Inter-xTR Roam (old and new APs on different xTRs), check whether the client is on the new AP (connected to the new xTR2), using the **show wireless client summary** and **show mac addr dyn** commands on the WLC1 and xTR2.
6. Check whether the client is registered from the new xTR2, using the **show lisp site detail** command on the MAP server.

Verify roaming on MAP werver for SDA

Run this command on the MAP server, before and after the roam, to check whether the client IP is registered from current xTR, and client MAC is registered from both current xTR, and WLC.

```
Device# show lisp site detail
```

167 NAT Support on Mobility Groups

Topics:

- NAT support for mobility groups

Explains how to configure Network Address Translation support for wireless mobility groups to enable seamless client roaming across different network segments.

NAT support for mobility groups

This concept explains how Network Address Translation (NAT) supports mobility tunnels between peer controllers in a mobility group, including scenarios in which one or both controllers are located behind a NAT device.

NAT support for mobility groups is a mobility architecture enhancement that

- enables mobility tunnels between peer controllers, even when one or both peers operate behind a NAT device,
- translates public and private IP addresses to maintain connectivity between controllers, and
- requires proper configuration of both addresses for NAT peers to ensure seamless mobility communication.

Additional information

Depending on the placement and number of NAT devices, translation might be required at one or both ends of the tunnel.

When you configure a NAT mobility peer, configure both the private IP address (the address used inside the network before the NAT device) and the public IP address (the address visible on the external network).

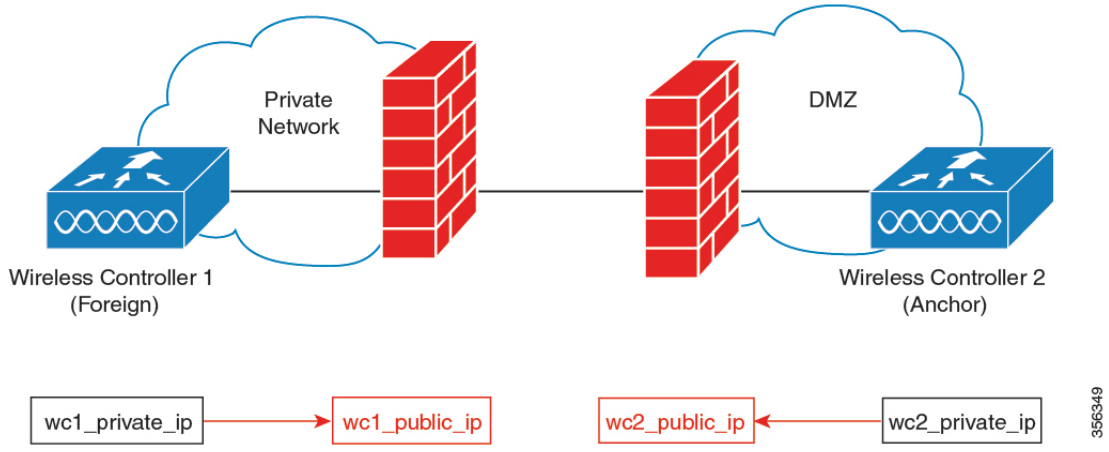
If a firewall is present between mobility group controllers, ensure that the following ports are accessible:

- Port 16666 for mobility control messages
- Port 16667 for mobility data messages

> **Note**
>
> Ensure that ports 16666 and 16667 are accessible through any firewalls between peer controllers.

This figure illustrates how public and private IP addresses are translated to establish a mobility tunnel between controllers behind NAT devices.

Figure 29: Mobility NAT



Restrictions for NAT support on mobility groups

Adhere to these restrictions when configuring NAT support on mobility groups to ensure proper controller peer connectivity and client roaming functionality.

- Only one-to-one (static) NAT entries entries can exist for the controller peers that form the mobility tunnels.
- Configuring multiple peers with the same public IP address is not supported.
- Private IP addresses of the configured peers must be unique.
- Port Address Translation (PAT) is not supported.
- If peer controllers of different types, for example, Cisco AireOS and Cisco Catalyst 9800 Series, are placed behind NAT, Inter-Release Controller Mobility (IRCM) is not supported for client roaming.
- IPv6 address translation is not supported.

Functionalities supported on mobility NAT

This topic explains the functionalities supported on mobility NAT, including supported controllers, features, and application compatibility.

This topic lists the functionalities supported on mobility NAT.

Two controllers, with the foreign controller behind a NAT device (1to1 NAT only)	Yes
Two controllers, with the anchor controller behind a NAT device (1to1 NAT only)	Yes
Two controllers, with the anchor and foreign controller behind a NAT device (1to1 NAT only)	Yes
Multiple foreign and anchor controllers behind NATs (1to1 NAT only)	Yes
Supported Cisco Catalyst 9800 Series Wireless Controllers	• Cisco Catalyst 9800-40 Wireless Controller • Cisco Catalyst 9800-80 Wireless Controller • Catalyst 9800 Wireless Controller for Cloud • Cisco Catalyst 9800-L Wireless Controller
Number of peers supported	72
Manageability using SNMP, Yang, and web UI	Yes
IRCM support for mobility	Yes
SSO	Yes
Client roaming (Layer 2 and Layer 3) between Cisco Catalyst 9800 Series Wireless Controllers	Yes
Client roaming (Layer 2 and Layer 3) between Cisco Catalyst 9800 Series Wireless Controller and AireOS controller	No

Supported applications on the mobility tunnel	• Native profiling • AP list • PMK cache • Mesh AP

Configure a mobility peer (CLI)

Configure a mobility peer by using the CLI to add a peer with a required private IP address and an optional public IP address.

Add a mobility peer to your wireless controller by specifying a required private IP address and, if needed, an optional public IP address using the CLI.

Before you begin

Ensure that both the private and public IP addresses of a mobility peer use the same type: IPv4 or IPv6.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Add a mobility peer to the list by specifying the required private IP address. Optionally, include a public IP address.

   Example

   ```
   Device(config)# wireless mobility group member mac-address peer_mac ip
   peer_private_ip [ public-ip peer_public_ip ] group group_name
   ```

   Example

   ```
   Device(config)# wireless mobility group member mac-address 001e.494b.04ff ip
    11.0.0.2 public-ip 4.0.0.112 group dom1
   ```

   Note

   You cannot configure multiple peers with the same private or public IP address.

3. Return to privileged EXEC mode.

Example

```
Device(config)# exit
```

The controller saves the mobility peer with its private IP address and, if provided, its public IP address. You can use this peer for wireless mobility group operations.

Verify NAT support on mobility groups

Verify NAT support on mobility groups.

To display the mobility information of a client, use this command:

```
Device# show wireless client mac-address 000a.bd15.0010 detail

Client MAC Address : 000a.bd15.0010
Client IPv4 Address : 100.100.0.2
Client Username: N/A
AP MAC Address : 000a.ad00.0800
AP Name: SIM-AP-7
AP slot : 1
.
.
.
```

To display mobility peer information using a private peer IP address, use this command:

```
Device# show wireless mobility peer ip 21.0.0.2

Mobility Peer Info
==================
Ip Address : 21.0.0.2
Public Ip Address : 3.0.0.22
MAC Address : cc70.ed02.c3b0
Group Name : dom1
.
.
.
```

168 Static IP Client Mobility

Topics:

- Static IP client mobility

Explains how static IP clients maintain network connectivity while moving between different network locations and access points.

Static IP client mobility

This concept explains how static IP client mobility enables seamless connectivity for wireless clients that use static IP addresses as they roam across controllers within the same mobility group.

Static IP client mobility is a wireless network capability that

- allows clients with static IP addresses to maintain connectivity when roaming between controllers
- tunnels client traffic to controllers that support the client's original subnet, and
- ensures reliable service across controllers within the same mobility group.

Behavior and configuration details for static IP client mobility

Wireless networks sometimes require clients to use static IP addresses. When these clients move within the network, they may attempt to associate with different controllers.

If a client tries to connect to a controller that does not support the subnet that matches the static IP address, the connection fails. The controller inspects ARP requests from the client to determine whether the IP address was assigned by DHCP or was statically configured.

If the ARP request contains an IP address that is not present on any of the controller's Switched Virtual Interfaces (SVIs), the client is disconnected with a VLAN_FAIL error. In this case, client traffic is backhauled without explicit disconnection. The disconnection behavior that results from a VLAN mismatch was introduced in the 17.9.1 release.

Example of static IP client mobility in action

- If a wireless client with a static IP address moves to a controller that does not support the client's subnet, the controller analyzes the ARP request. If the subnet is not recognized, the controller disconnects the client, and the client traffic is backhauled.
- When the client moves to a controller within the same mobility group that supports the correct subnet, the traffic is tunneled to the appropriate controller. As a result, the client remains connected to the network even when subnet support varies across controllers.

Restrictions for static IP client mobility

Adhere to these restrictions when configuring static IP client mobility to ensure compatibility and feature support.

- This feature is not supported on the Fabric and Cisco Catalyst 9800 Wireless Controller for Switch platforms.
- IPv6 is not supported.
- FlexConnect mode is not supported.
- WebAuth (LWA and CWA) is not supported.
- Only Open, 802.11ax, and PSK authentication mechanisms are supported.
- This feature is supported only on WLANs that do not have a mobility anchor configured. If a mobility anchor is configured on a WLAN and static IP mobility is enabled, the feature is not supported.
- This feature is supported only when all peers are configured for static IP mobility.
- IRCM is not supported.

Configure a static IP client mobility profile (GUI)

Learn how to configure static IP client mobility using the graphical user interface (GUI) by updating the policy profile and enabling the Static IP Mobility option.

Provide guidance for users to draft a clear and concise purpose and results statement, outlining the objectives and expected outcomes for their task or project.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy** .
2. On the **Policy** page, click the policy profile name or click **Add** to create a new one.
3. Click the **Mobility** tab.
4. Set the **Static IP Mobility** field to **Enabled** state.
5. Click **Update & Apply to Device** .

Static IP clients are able to roam seamlessly between access points, maintaining their network sessions and improving overall connectivity.

Configure a static IP client mobility (CLI)

Learn how to configure static IP client mobility using CLI commands on your device.

Enable wireless client devices with static IP addresses to seamlessly roam between controllers in the network while maintaining their IP configuration.

Before you begin

- Configure the SVI interface (L3 VLAN interface) to service the static IP client on at least one of the peer controllers in the network.
- For clients to join a controller, the VLAN (based on the VLAN number in the policy profile configuration) should be configured on the device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a WLAN policy profile and enter wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy profile-policy-name
   ```

3. Enable static IP mobility.

   Example

   ```
   Device(config-wireless-policy)# static-ip-mobility
   ```

Static IP client mobility is successfully configured. Wireless clients with static IP addresses can roam across controllers without losing connectivity or their IP settings.

Verify static IP client mobility

Verify static IP client mobility.

Use these commands to verify the static IP client mobility configuration:

```
Device# show wireless profile policy detailed static-ip-policy


Policy Profile Name           : static-ip-policy
Description                   :
Status                        : DISABLED
VLAN                          : 1
Wireless management interface VLAN        : 34
Passive Client                : DISABLED
ET-Analytics                  : DISABLED
StaticIP Mobility             : DISABLED
WLAN Switching Policy
  Central Switching           : ENABLED
  Central Authentication      : ENABLED
  Central DHCP                : DISABLED
  Flex NAT PAT            : DISABLED
  Central Assoc               : DISABLED
WLAN Flex Policy
  VLAN based Central Switching           : DISABLED
WLAN ACL
  IPv4 ACL                    : Not Configured
  IPv6 ACL                    : Not Configured
  Layer2 ACL                  : Not Configured
  Preauth urlfilter list      : Not Configured
  Postauth urlfilter list     : Not Configured
WLAN Timeout
  Session Timeout             : 1800
  Idle Timeout                : 300
  Idle Threshold              : 0
WLAN Local Profiling
  Subscriber Policy Name      : Not Configured
  RADIUS Profiling            : DISABLED
  HTTP TLV caching            : DISABLED
  DHCP TLV caching            : DISABLED
WLAN Mobility
  Anchor                      : DISABLED
AVC VISIBILITY                : Disabled
Flow Monitor IPv4
  Flow Monitor Ingress Name   : Not Configured
  Flow Monitor Egress Name    : Not Configured
Flow Monitor IPv6
  Flow Monitor Ingress Name   : Not Configured
  Flow Monitor Egress Name    : Not Configured
NBAR Protocol Discovery       : Disabled
Reanchoring                   : Disabled
Classmap name for Reanchoring
  Reanchoring Classmap Name   : Not Configured
QOS per SSID
  Ingress Service Name        : Not Configured
  Egress Service Name         : Not Configured
QOS per Client
  Ingress Service Name        : Not Configured
  Egress Service Name         : Not Configured
```

```
Umbrella information
  Cisco Umbrella Parameter Map : Not Configured
Autoqos Mode                   : None
Call Snooping                  : Disabled
Fabric Profile
  Profile Name                 : Not Configured
Accounting list
  Accounting List                : Not Configured
DHCP
  required                     : DISABLED
  server address               : 0.0.0.0
 Opt82
  DhcpOpt82Enable              : DISABLED
  DhcpOpt82Ascii               : DISABLED
  DhcpOpt82Rid                 : DISABLED
  APMAC                        : DISABLED
  SSID                         : DISABLED
  AP_ETHMAC                    : DISABLED
  APNAME                       : DISABLED
  POLICY TAG                   : DISABLED
  AP_LOCATION                  : DISABLED
  VLAN_ID                      : DISABLED
Exclusionlist Params
  Exclusionlist                : ENABLED
  Exclusion Timeout            : 60
AAA Policy Params
  AAA Override                 : DISABLED
  NAC                          : DISABLED
  AAA Policy name              : default-aaa-policy
WGB Policy Params
  Broadcast Tagging            : DISABLED
  Client VLAN                  : DISABLED
Mobility Anchor List
  IP Address                                  Priority
  -------------------------------------------------------
```

```
Device# show run | section profile policy

wireless profile policy default-policy-profile
 central switching
 description "default policy profile"
 static-ip-mobility
 vlan 50
 no shutdown
```

169 Mobility Domain ID - 802.11i Roaming

Topics:

- Mobility domain ID—802.11i roaming

Explains the use of Mobility Domain Identifiers in 802.11i wireless networks to enable secure and seamless client roaming between access points.

Mobility domain ID—802.11i roaming

A mobility domain is a group of access points (APs) that forms a continuous radio frequency space. This grouping allows synchronization of the Pairwise Master Key (PMK) to support fast roaming for 802.11i (WPA) and 802.11r (Fast Transition). The Mobility Domain ID (MDID) manages PMK sharing within defined AP clusters to enable efficient roaming.

A mobility domain ID is a wireless network grouping mechanism that

- enables APs to form a continuous radio frequency space for seamless client roaming,
- facilitates synchronization and sharing of Pairwise Master Keys (PMKs) across APs, and
- manages roaming boundaries by linking APs under a common mobility domain for 802.11i and WPA, as well as 802.11r (Fast Transition).

A mobility domain can be defined in two ways:

- Static configuration: APs are grouped under a common mobility domain ID (MDID).
- Dynamic computation: Spatial clustering algorithms group APs based on neighbor associations.

The MDID is used by 802.11r to define the network boundary where fast roaming is supported.

Within a mobility domain, APs share PMKs. This capability allows clients to roam quickly without fully reauthenticating.

If an MDID is configured, it overrides the site tag for PMK sharing control.

Key characteristics and functionality

- PMK sharing: When an MDID is configured for a group of APs, all APs in that group share PMK cache keys, even if they belong to different site tags. Share PMKs only within the same mobility domain to ensure fast and secure client roaming.
- Precedence: If a mobility domain is defined, the MDID takes precedence over the AP site tag for PMK caching.
- Flexibility: The MDID method supports PMK cache sharing for both central and local authentication modes.

Implementation information

- Configuration method: You can configure MDID only through the open configuration model. There is no CLI or GUI support for this feature.
- Scale: Each mobility domain (site tag or MDID) supports up to 100 APs in Cisco IOS XE 17.2.1. Each AP supports up to 1,000 PMK cache entries in this release.
- Restriction: The Mobility Domain ID–802.11i Roaming feature does not function when Flex APs operate in standalone mode because controller coordination is required for key sharing.

Historical context and release notes

- Before Cisco IOS XE 17.2.1, the PMK cache was shared across FlexConnect APs using the AP site tag. All APs in the same site tag could share the PMK cache, but this support applied only to central authentication.
- Since Cisco IOS XE 17.2.1, you can assign a unique MDID to each AP. All APs with the same MDID share PMK cache keys regardless of site tag assignment. MDID-based PMK cache sharing now works for both central and local authentication. It also ensures that APs outside an MDID do not participate in the same PMK cache, even if they are under the same site tag.

Notes

Note

- The Mobility Domain ID–802.11i Roaming feature does not function when Flex APs operate in standalone mode because controller coordination is required for key sharing.
- You can set MDID only through the open configuration model.
- In Cisco IOS XE 17.2.1, each site tag or MDID supports up to 100 APs, and each AP supports up to 1,000 PMK entries.

You can configure MDID only through open configuration models. For more information, see the Cisco IOS XE Programmability Configuration Guide at https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/prog/configuration/172/b_172_programmability_cg.html.

Verify mobility domain ID - 802.11i roaming

Verify mobility domain ID - 802.11i roaming.

This examples shows how to view and verify the 802.11i roaming configuration:

```
Device#  show running-config | section specific-config
ap specific-config 58ac.70dc.xxxx hostname AP58AC.70DC.XXXX
   roaming-domain roaming_domain_2
ap specific-config 78xc.f09d.xxxx hostname AP78XC.F09D.XXXX
   roaming-domain roaming_domain_3
```

170 802.11r Support for Flex Local Authentication

Topics:

- 802.11r support for FlexConnect local authentication

802.11r support for FlexConnect local authentication

Describes the 802.11r fast transition capability for locally authenticated FlexConnect clients.

802.11r support for FlexConnect local authentication is a fast transition capability that

- enables fast roaming for locally authenticated FlexConnect clients by sharing Pairwise Master Key (PMK) cache entries
- uses site tag or Mobility Domain ID (MDID) grouping to distribute PMK cache entries across APs, and
- supports up to 100 APs per group by default and 300 with high scale mode, with a maximum of 1000 PMK entries per AP.

Implementation details

In releases prior to Cisco IOS XE Amsterdam 17.2.1, the FlexConnect mode fast transition was supported only in centrally authenticated clients. This was achieved by sharing the Pairwise Master Key (PMK) to all the FlexConnect APs in the same site tag. From Cisco IOS XE Amsterdam 17.2.1, fast transition is supported even for locally authenticated clients.

The client PMK cache entries are shared and distributed to all the APs in the same site tag. From Cisco IOS XE Amsterdam 17.2.1, another grouping called Mobility Domain ID (MDID) is introduced, for sharing the PMK cache entries. MDID can be configured for APs using the open configuration model only. There is no CLI or GUI support.

The PMK cache distribution in a FlexConnect local site (using either the site tag or MDID) is restricted to 100 APs per group by default and 300 with high scale mode, with a maximum support for 1000 PMK entries per AP.

The following are the 802.11r support guidelines:

- Supports 802.11r on FlexConnect local authentication only with Over-the-Air method of roaming. Over-the-DS (Distribution System) is not supported.
- Supports adaptive 11r for Apple clients.
- Supports both Fast Transition + 802.1x and Fast Transition + PSK.

> **Note**
>
> This is supported only when clients join the standalone mode AP.

802.11r support verification for flex local authentication

Provides commands to verify 802.11r support for flex local authentication.

Use the following commands to verify 802.11r support for flex local authentication by checking PMK caches and 802.11r flex roam attempts.

To verify the number of PMK caches, use the **show wireless pmk-cache** command:

```
Device# show wireless pmk-cache
Number of PMK caches in total : 1

Type      Station            Entry Lifetime  VLAN Override        IP Override
         Audit-Session-Id              Username
-----------------------------------------------------------------------------------------
DOT11R    74xx.bx5a.07xx      87              NA
           000000000000000FF3562B5D      jey
```

To verify the 802.11r flex roam attempts, use the **show wireless client mac-address 74xx.bx5a.07xx mobility history** command:

```
Device# show wireless client mac-address 74xx.bx5a.07xx mobility history
Recent association history (most recent on top):

AP Name                                      BSSID           AP Slot    Assoc
 Time               Instance   Mobility Role   Run Latency (ms)     Dot11 Roam
 Type

-------------------------------------------------------------------------------------------------------------------------


APM-9120-1-GCP                               d4xx.80xx.8fxx  1
12/11/2019 18:44:37      1          Local           2                   802.11R


APM-4800-3                                   f4xx.e6xx.08xx  1
12/11/2019 18:43:02      1           Local           17547                N/A


show wireless stats client detail | sec roam
Total 11r flex roam attempts                    : 1
```

171 Opportunistic Key Caching

Topics:

- Opportunistic key caching

Explains Opportunistic Key Caching technology and provides step-by-step guidance for enabling and verifying the feature through both command-line and graphical interfaces.

Opportunistic key caching

Explains the OKC technique that allows wireless clients and the WLAN infrastructure to cache only one PMK for client association with a WLAN.

Opportunistic Key Caching (OKC) is an enhancement of the WPA2 Pairwise Master Key ID (PMKID) caching method that

- allows wireless clients and the WLAN infrastructure to cache only one PMK for client association with a WLAN, even when roaming between multiple APs
- enables APs to share the original PMK that is used for the WPA2 4-way handshake, and
- works with WPA2-EAP as a fast roaming technique.

Key characteristics

OKC shares several characteristics with PMKID caching:

- The initial association to an AP is a regular first-time authentication to the corresponding WLAN, where you must complete the entire 802.1X/EAP authentication for the authentication server, and the 4-way handshake for key generation, before sending data frames.
- All APs share the original PMK from a client session under a centralized device that caches and distributes the original PMK to all the APs.
- New encryption keys are generated every time a client reassociates with APs.

OKC support varies by client type:

- Microsoft and some Android clients support OKC.
- Apple and few Android clients support 802.11r as an alternative fast roaming method.

OKC configuration characteristics:

- OKC is enabled by default on a WLAN.
- Disabling OKC on a WLAN disables the OKC even for the OKC-supported clients.
- A new configuration is introduced for each WLAN in the controller in Cisco IOS XE Amsterdam 17.2.1, to disable or enable fast and secure roaming with OKC at the corresponding AP.

Enable opportunistic key caching

Configure opportunistic key caching to improve wireless client roaming performance.

Opportunistic Key Caching (OKC) improves wireless client roaming performance by allowing clients to use cached encryption keys when moving between access points within the same mobility domain.

Use this procedure when you need to enable OKC on a WLAN to enhance client roaming experience. OKC is enabled by default on most configurations.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enter WLAN configuration submode.

   Example

   ```
   Device(config)# wlan wlan-profile-name wlan-identifier <1-4096>
   ssid-network-name
   ```

   ```
   Device(config)# wlan wlan-profile-name 18 san-ssid
   ```

3. Enable Opportunistic Key Caching, if not enabled. By default, the OKC feature is enabled.

   Example

   ```
   Device(config-wlan)# okc
   ```

   Use the **no** form of this command to disable the OKC feature.

Opportunistic Key Caching is now enabled on the specified WLAN, allowing wireless clients to experience improved roaming performance when moving between access points.

Enable opportunistic key caching (GUI)

Configure opportunistic key caching using GUI.

Enable opportunistic key caching to improve wireless network performance by allowing devices to cache encryption keys for faster roaming between access points.

Opportunistic key caching (OKC) is configured through the WLAN settings in the Advanced tab. This feature is enabled by default but can be managed based on your network security requirements.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.

   The **Add WLAN** dialog box is displayed.

3. In the **Add WLAN** dialog box, click the **Advanced** tab and complete the following procedure:
   a) In the **11ax** section, check the **OKC** check box to disable or enable the feature. By default this feature is enabled.
   b) Click **Update & Apply to Device**.

Opportunistic key caching is configured according to your selection. The WLAN settings are updated and applied to the device.

Verify opportunistic key caching

Provides commands to verify opportunistic key caching configuration.

Verify whether opportunistic key caching (OKC) is disabled for a WLAN profile using show commands.

These example shows how to verify whether OKC is disabled for a WLAN profile.

```
Device# show wlan id 18
WLAN Profile Name     : 18%wlanprofile
================================================
Identifier                                       : 18
Description                                      :
Network Name (SSID)                              : san-ssid
Status                                           : Disabled
Broadcast SSID                                   : Enabled
Advertise-Apname                                 : Disabled
Universal AP Admin                               : Disabled
Max Associated Clients per WLAN                  : 0
Max Associated Clients per AP per WLAN           : 0
Max Associated Clients per AP Radio per WLAN     : 200
OKC                                              : Disabled
Number of Active Clients                         : 0
CHD per WLAN                                     : Enabled
WMM                                              : Allowed
Channel Scan Defer Priority:
  Priority (default)                             : 5
  Priority (default)                             : 6
Scan Defer Time (msecs)                          : 100
Media Stream Multicast-direct                    : Disabled
CCX - AironetIe Support                          : Disabled
Peer-to-Peer Blocking Action                     : Disabled
Radio Policy                                     : All
```

```
Device# show run wlan
wlan name 2 ssid-name
wlan test 24 test
wlan test2 15 test2
wlan test4 12 testssid
 radio dot11a
wlan wlan1 234 wlan1
wlan wlan2 14 wlan-aaa
 security dot1x authentication-list realm
wlan wlan7 27 wlan7
wlan test23 17 test23
wlan wlan_1 4 ssid_name
 security dot1x authentication-list authenticate_list_name
wlan wlan_3 5 ssid_3
 security wpa wpa1
 security wpa wpa1 ciphers aes
wlan wlan_8 9 ssid_name
 no security wpa
 no security wpa wpa2
 no security wpa wpa2 ciphers aes
 no security wpa akm dot1x
 security web-auth
wlan test-wlan 23 test-wlan
wlan wlan-test 1 wlan2
 mac-filtering default
wlan 18%wlanprofile 18 san-ssid
 no okc
```

Part XII

High Availability

Topics:

- High Availability
- Redundancy Management Interface

This guide provides technical procedures for configuring and managing High Availability on Cisco Catalyst 9800 Series Wireless Controllers. It enables administrators to implement robust failover mechanisms, monitor system health, and troubleshoot redundancy configurations to ensure continuous wireless network operations.

172 High Availability

Topics:

- High Availability
- Auto-upgrade feature
- Link layer discovery protocol use cases
- Reload reason history

The Cisco Catalyst 9800 Series Wireless Controller High Availability (HA) feature reduces network downtime by enabling seamless failover between active and standby controllers, preserving AP and client connectivity through stateful switchover, and mirroring AP and client databases to avoid disconnections during failover events.

High Availability

This topic explains High Availability (HA) and how it reduces wireless network downtime by enabling seamless failover between controllers, ensuring continuous service for APs and clients.

A high availability feature is a wireless controller capability that

- reduces network downtime by enabling seamless failover between active and standby controllers,
- preserves AP and client connectivity through stateful switchover by maintaining CAPWAP tunnels and client sessions during failover, and
- mirrors AP and client databases from the active controller to the standby controller to prevent APs from entering discovery state and avoid client disconnections.

Feature history for High Availability

This table provides release and related information for the features explained in this module.

These features are available in all the releases subsequent to the one they were introduced in, unless noted otherwise.

Table 144: Feature history

Release	Feature	Feature information
Cisco IOS XE 17.18.1	Enhanced Gateway Reachability Statistics	Improves visibility into gateway reachability and provides detailed statistics for ICMP, ARP, and ND probes. This feature enables simplified troubleshooting, greater transparency, and more reliable diagnostics for HA and RMI functionality. This command is introduced: • **show platform software rif-mgr chassis { active | standby} r0 gateway-statistics** This command is modified: • **show platform software rif-mgr chassis { active | standby} r0 resource-status**

Release	Feature	Feature information
Cisco IOS XE 17.9.1	High Availability Deployment for Application Centric Infrastructure (ACI) Network	This feature avoids interleaving traffic between the old and new active controller using these functionalities: • Bringing down Wireless Management Interface (WMI) faster. • Disabling fast switchover notification.
	Link Layer Discovery Protocol (LLDP) Support in the Standby Controller	From this release, the Link Layer Discovery Protocol (LLDP) process will be up and running in both active and standby controllers.
Cisco IOS XE 17.6.1	Standby Interface Status using Active SNMP	This feature allows the standby controller interface status to be queried at the active using SNMP.
Cisco IOS XE 17.5.1	Auto-Upgrade	The auto-upgrade feature enables the standby controller to upgrade to active controller's software image, so that both controllers can form an high availability (HA) pair.
Cisco IOS XE 17.5.1	Standby Monitoring Enhancements	The Standby Monitoring Enhancements feature monitors the standby CPU or memory information from the active controller. Also, this feature independently monitors the standby controller using SNMP for the interface MIB. The cLHaPeerHotStandbyEvent and cLHaPeerHotStandbyEvent MIB objects in CISCO-HA-MIB are used to monitor the standby HA status.
Cisco IOS XE 17.4.1	Gateway Reachability Detection	Gateway reachability feature minimizes the downtime on APs and clients when the gateway reachability is lost on the active controller.
Cisco IOS XE 17.1.1s	Redundant Management Interface	The Redundancy Management Interface (RMI) is used as a secondary link between the active and standby controllers. This interface is the same as the Wireless Management Interface and the IP address on this interface is configured in the same subnet as the Wireless Management Interface.

Additional reference information

High availability enables seamless controller failover, ensuring that APs and clients remain connected during controller outages. These notes and recommendations apply to HA deployments:

- Do not shut or unshut the RP port during controller bootup in HA mode.
- If RP communication is lost between the active and standby controllers during HA synchronization, the standby controller intentionally crashes when IPC communication fails.

  If the RP link is restored, the standby controller restarts gracefully and forms an HA pair.
- When the controller operates as a spanning tree host, configure portfast trunk on the uplink switch to ensure faster convergence. Use **spanning-tree port type edge trunk** or **spanning-tree portfast trunk**.
- You can configure FIPS in an HA setup. For information, see the Configuring FIPS in HA Setup.
- Do not configure the secondary IPv4 address. The controller uses the IPv4 secondary address internally for RMI purposes.

  Configure only one management IPv6 address on the Wireless Management Interface (WMI). The controller uses any secondary address for RMI-IPv6.

  Configuring more than one management IPv4 address or more than one management IPv6 address on the WMI may cause unpredictable behavior.

During a failover event, only one CAPWAP tunnel is maintained between APs and the active controller. This ensures zero downtime for client services and no SSID outages. Database mirroring prevents APs from entering the discovery state and ensures that clients remain connected without interruption.

Prerequisites for High Availability

This topic explains the prerequisites for high availability, including interface configuration, HA port options, and latency, bandwidth, and MTU requirements.

To ensure high availability, configure interfaces properly, select the appropriate HA port, and meet latency, bandwidth, and MTU requirements for RP links.

External interfaces and IPs

All interfaces are configured on the Active box and synchronized with the Standby box. Therefore, the same set of interfaces is present on both controller s.

External nodes connect to the same IP addresses regardless of which controller they are connected to.

APs, clients, DHCP servers, Cisco Prime Infrastructure, Cisco Catalyst Center, and Cisco Identity Services Engine (ISE) servers, as well as other controller members in the mobility group, always connect to the same IP address.

The SSO switchover is transparent to these devices. However, if TCP connections exist from external nodes to the controller , reset and reestablish those connections.

HA interfaces

The HA interface provides:

- Provides connectivity between the controller pair before an IOSd comes up,
- provides IPC transport across the controller pair, and
- enables redundancy across control messages exchanged between the controller pair. The control messages include HA role resolution, keepalive messages, notifications, HA statistics, and similar messages.

You can select an SFP or RJ-45 connection for the HA port. Supported Cisco SFPs are:

- GLC-SX-MMD
- GLC-LH-SMD

> **Note**
>
> Connect either the SFP port or the RJ-45 port to the peer. Do not connect both ports at the same time.

HA operates when either an SFP or RJ-45 connection is present between the two controllers. If you connect an SFP link while RJ-45 HA is active, the HA pair restarts. The restart occurs even if the SFP link is not connected.

> **Note**
>
> - Use a dedicated physical network interface card (NIC) and switch for the redundancy port (RP) when deploying the HA pair across two hosts. This prevents keepalive losses and false HA switchovers or alarms.
> - Disable security scans on VMware virtual instances to prevent HA issues.

> **Note**
>
> Connect RP links using switches to enable controller HA. Keep the round-trip time between the two controllers under 80 milliseconds.

Latency, bandwidth, and MTU

These are the latency, bandwidth, and MTU prerequisites for the RP link:

- The maximum supported latency for the RP link is 80 milliseconds round-trip time (RTT).
- The RP link must support a minimum bandwidth of 60 megabits per second (Mbps).
- The RP link must support a minimum maximum transmission unit (MTU) of 1500 bytes.

High Availability restrictions

Outlines restrictions and important guidelines for configuring High Availability, including synchronization requirements, interface pairing, and switchover scenarios on wireless controllers.

- Wait until configuration synchronization completes on the standby controller. Before initiating a fail-safe Stateful Switchover (SSO), ensure that the standby controller has been powered on for sufficient time (up to 24 minutes [up to 1,440 seconds] on some platforms) to achieve readiness. Use the **show wireless stats redundancy config database** command to view database statistics.
- During a switchover in local mode, NBAR engine flow states are lost. As a result, classification restarts and may lead to incorrect packet classification.
- You can use HA connections only with IPv4.

- When you perform a switchover or an active reload, the high-availability link goes down on the new primary controller.
- Do not enable hyper-threading in HA systems. If enabled, HA keepalives are lost, and a stack merge may occur.
- You cannot access the web UI from the standby RMI interface.
-
-
- Configure two HA interfaces, RMI and RP, on the same subnet. Do not share this subnet with any other interfaces on the device.
- After a switchover, you must re-establish any TCP session because synchronization is not possible.
- Client SSO does not address clients that have not reached the RUN state. These clients are removed after a switchover.
- Statistics tables are not synchronized from the active controller to the standby controller.
- Creating a machine snapshot of a VM hosting controller HA interfaces is not supported. This action may lead to a crash in the HA controller.
- Clients that are not in RUN state are reauthenticated after a switchover.
- Application classification may not be retained after SSO:
  - AVC limitation–After a switchover, context transfer or synchronization to the standby controller does not occur. The new active flow must be relearned. AVC QoS does not take effect during classification failure.
  - A voice call cannot be recognized after a switchover because a voice policy is based on RTP or RTCP protocol.
  - Auto QoS does not work due to AVC limitation.
- For virtual platforms, pair the active controller and the standby controller with the same interface. For hardware appliances, use a dedicated HA port.
- You can synchronize static IP addressing to the standby controller, but you cannot use the IP address from the standby controller.
- You can map a dedicated HA port to a 1-gigabit (1,000 Mbps) interface only.
- To use EtherChannels in HA mode in releases up to Cisco IOS XE Gibraltar 16.12.x, ensure that the channel mode is set to On.
- EtherChannel Auto-mode is not available in HA mode in releases up to Cisco IOS XE Gibraltar 16.12.x.
- LACP and PAGP protocols cannot be used in HA mode in releases up to Cisco IOS XE Gibraltar 16.12.x.
- When the controller operates as a host for spanning tree, configure portfast trunk on the uplink switch using **spanning-tree port type edge trunk** or **spanning-tree portfast trunk** command to ensure faster convergence.
- The **clear chassis redundancy** and **write erase** commands do not reset the chassis priority to the default value.
- While configuring devices in HA, ensure that members do not have wireless trustpoints with the same name but different keys. If you form an HA pair between two standalone controllers with mismatched trustpoints, the wireless trustpoint does not come up after SSO. The *rsa keypair* file exists but is incorrect because the *nvram:private-config* file is not synchronized with the actual *WLC_WLC_TP* key pair.
- Before forming HA, delete existing certificates and keys from each controller that was previously deployed as standalone. This is a best practice.
- Do not configure the WLAN or WLAN policy after a switchover while recovery is in progress. Doing so may cause the controller to crash.

- After a switchover, clients that are not in RUN state and not connected to an AP are removed after 300 seconds (5 minutes).

Best practices for RP port configuration

Use these guidelines when configuring RP ports to ensure proper redundancy and select the correct subnet for Local and Remote IP addresses.

When you configure RP ports, use these best practices:

- Ensure that the Local and Remote IP addresses are in the same subnet.
- Use the 169.254.X.X/16 subnet, deriving the last two octets from the management interface.
- Do not use the 10.10.10.x/24 subnet for the RP port.
- For more information about RMI+RP chosen as the redundancy method, see Information About Redundancy Management Interface .

Configure High Availability (CLI)

Learn how to configure high availability using CLI commands to set up device priorities, high-availability interfaces, and keepalive parameters for redundancy pairs.

Set up high availability for network redundancy and automatic failover between devices using the CLI.

Before you begin

Ensure that the active and standby controllers use the same mode—either Install mode or Bundle mode—and the same image version. Use Install mode.

Procedure

1. Optional: Configure the priority of the device.

Example

```
Device# chassis chassis-num priority chassis-priority
```

Note

From Cisco IOS XE 16.12.x onward, a device reload is not required for the chassis priority to take effect.

- *chassis-num* —Enter the chassis number (range: one to two).
- *chassis-priority* —Enter the chassis priority (range: one to two; default: one).

> **Note**
>
> If both devices boot up simultaneously, the device with the higher priority (2) becomes active. The other device becomes standby. If both devices have the same priority, the device with the smaller MAC address becomes active, and the peer device becomes standby.

2. Set the chassis high-availability parameters.

   Example

   ```
   Device# chassis redundancy ha-interface GigabitEthernet num local-ip
   local-chassis-ip-addr network-mask remote-ip remote-chassis-ip-addr
   ```

   Example

   ```
   Device# chassis redundancy ha-interface
                          GigabitEthernet 2 local-ip 4.4.4.1 /24 remote-ip
   4.4.4.2
   ```

   - *num* –GigabitEthernet interface number (range: zero to 32).
   - *local-chassis-ip-addr* –Enter the IP address of the local chassis high-availability interface.
   - *network-mask* –Enter the network mask or prefix length in slash nn or dotted decimal format.
   - *remote-chassis-ip-addr* –Enter the remote chassis IP address.

   This command is not supported on these models:

   - Cisco Catalyst CW9800H1 Wireless Controller.
   - Cisco Catalyst CW9800H2 Wireless Controller.
   - Cisco Catalyst CW9800M Wireless Controller.

   RMI-based high availability is mandatory in the Cisco Catalyst CW9800H1 Wireless Controller, Cisco Catalyst CW9800H2 Wireless Controller, and Cisco Catalyst CW9800M Wireless Controller.

3. Configure the peer keepalive timeout value.

   Example

   ```
   Device# chassis redundancy keep-alive timer timer
   ```

   Set the time interval in multiples of 100 milliseconds (ms). Enter one for the default value.

4. Set the peer keepalive retry value that determines when the system considers the peer down.

   Example

   ```
   Device# chassis redundancy keep-alive retries retry-value
   ```

The default value is five.

After you complete these steps, high availability is configured between two devices. The system uses device priorities, high-availability interfaces, and keepalive parameters to ensure redundancy and seamless failover if a device fails.

Disable High Availability

This topic explains how to disable high availability on controllers using the RP or RMI method. It includes the required commands and important considerations for supported models.

When you disable high availability, all HA-related parameters are removed and the controller returns to stand-alone mode.

- Use **clear chassis redundancy** with the RP method to clear the local IP, remote IP, HA interface, mask, timeout, and priority.
- Use **no redun-management interface vlan chassis** with the RMI method.
- After you unpair the controllers, the startup and HA configuration of the standby controller are cleared, and it enters Day zero state.

If you configure the controller using the RP method for SSO, use this command to clear all HA-related parameters: local IP, remote IP, HA interface, mask, timeout, and priority:

- **clear chassis redundancy**

If you configure the controller using the RMI method, use this command:

- **no redun-management interface vlan chassis**

Note

This command is not supported on these models:

- Cisco Catalyst CW9800H1 Wireless Controller .
- Cisco Catalyst CW9800H2 Wireless Controller .
- Cisco Catalyst CW9800M Wireless Controller .

Note

Reload your devices to apply the changes.

Before you execute the command, you see this warning on the active controller:

```
Device# clear chassis redundancy
WARNING: Clearing the chassis HA configuration will result in both the chassis
 move into
Stand Alone mode. This involves reloading the standby chassis after clearing
its HA
```

```
configuration and startup configuration which results in standby chassis coming
 up as a totally
clean after reboot. Do you wish to continue? [y/n]? [yes]:
*Apr 3 23:42:22.985: received clear chassis.. ha_supported:1yes
WLC#
*Apr 3 23:42:25.042: clearing peer startup config
*Apr 3 23:42:25.042: chkpt send: sent msg type 2 to peer..
*Apr 3 23:42:25.043: chkpt send: sent msg type 1 to peer..
*Apr 3 23:42:25.043: Clearing HA configurations
*Apr 3 23:42:26.183: Successfully sent Set chassis mode msg for chassis 1.chasfs
 file updated
*Apr 3 23:42:26.359: %IOSXE_REDUNDANCY-6-PEER_LOST: Active detected chassis 2
 is no
longer standby
```

On the standby controller, these messages indicate that the configuration is being cleared:

```
Device-stby#
*Apr 3 23:40:40.537: mcprp_handle_spa_oir_tsm_event: subslot 0/0 event=2
*Apr 3 23:40:40.537: spa_oir_tsm subslot 0/0 TSM: during state ready, got event
 3(ready)
*Apr 3 23:40:40.537: @@@ spa_oir_tsm subslot 0/0 TSM: ready -> ready
*Apr 3 23:42:25.041: Removing the startup config file on standby
!Standby controller is reloaded after clearing the chassis.
```

Copy a WebAuth tar bundle to the standby controller (GUI)

Copy a WebAuth tar bundle to the standby controller in a high-availability configuration.

Ensure the standby controller receives the WebAuth tar bundle to maintain seamless high-availability functionality.

Copy a WebAuth tar bundle to the standby controller in a high-availability configuration.

Procedure

1. Go to **Administration** , select **Management** , and then select **Backup and Restore**.
2. Select **To Device** from the **Copy** drop-down list.
3. Select **WebAuth Bundle** from the **File Type** drop-down list.
4. Select **TFTP** , **SFTP** , **FTP** , or **HTTP** from the **Transfer Mode** drop-down list.

   The required values for the **Server IP Address** and **File Path** fields depend on the selected transfer mode.

   - **TFTP**
     - **IP Address (IPv4/IPv6)** : Enter the server IP address (IPv4 or IPv6) of the TFTP server you want to use.
     - **File Path**: Enter the file path. Start the file path with a slash, such as */path*.
     - **File Name**: Enter a file name.

       Do not use spaces in the file name. Use underscores and hyphens. Make sure the file name ends with .tar, for example, webauthbundle.tar.
   - **SFTP**
     - **IP Address (IPv4/IPv6)**: Enter the server IP address (IPv4 or IPv6) of the SFTP server that you want to use.
     - **File Path**: Enter the file path. Start the file path with a slash, such as */path*.
     - **File Name**: Enter a file name.

Do not use spaces in the file name. Only underscores and hyphens are allowed. Make sure the file name ends with .tar, for example, webauthbundle.tar.

- **Server Login Username**: Enter the SFTP server login user name.
- **Server Login Password**: Enter the SFTP server login passphrase.

- **FTP**
  - **IP Address (IPv4/IPv6)**: Enter the server IP address (IPv4 or IPv6) of the FTP server that you want to use.
  - **File Path**: Enter the file path. Start the file path with a slash, such as */path* .
  - **File Name**: Enter a file name.

    Do not use spaces in the file name. Only underscores and hyphens are allowed. Make sure the file name ends with .tar, for example, webauthbundle.tar.
  - **Logon Type**: Choose **Anonymous** or **Authenticated** . If you choose **Authenticated**, these fields are activated:
    - **Server Login Username**: Enter the FTP server login user name.
    - **Server Login Password**: Enter the FTP server login passphrase.

- **HTTP**
  - **Source File Path**: Click **Select File** to select the configuration file, and click **Open**.

5. Click the **Yes** or **No** radio button to back up the existing startup configuration to Flash.

   Save the configuration to Flash to propagate the WebAuth bundle to other members, including the standby controller.
6. Click **Download File**.

The WebAuth tar bundle is successfully copied to the standby controller, ensuring high-availability readiness.

System and network fault handling

This topic explains system and network fault handling, including controller switchover conditions and recovery scenarios for both system and network faults.

If the standby controller crashes, it reboots and comes up as the standby controller. Bulk sync follows causing the standby to become hot. If the active controller crashes, the standby becomes active. The new active controller assumes the role of primary and tries to detect a dual active.

These matrices provide a clear picture of the conditions the controller switchover would trigger:

Table 145: System and Network Fault Handling after 17.5

Number	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
1	Up	P-Reachable	G-Reachable	G-Reachable	No SSO	No Action	

Note	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
2	Up	P-Reachable	G-Reachable	G-Unreachable	No SSO	No action is required. The standby unit is not ready for SSO in this state because it does not have gateway reachability. In this scenario, the standby unit appears in standby-recovery mode. **Spring Back:** If the gateway reachability is restored (G_Reachable), the controller returns to Standby state (no reboot is necessary). **Note** RP resources and gateway resources each trigger distinct actions.	**Spring Back:** If the gateway reachability is restored (G_Reachable), the controller transitions to Standby state. A reboot is not required.
3	Up	P-Reachable	G-Unreachable	G-Reachable	SSO	The system exchanges gateway reachability messages over the RMI and RP links. When the active controller reboots, the standby controller takes over as the active controller. The RP goes down during the reboot process.	

Node	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
							The Stack Manager sends a message to the standby controller to initiate a role change. The standby controller consults the active controller. • If the active controller responds, the standby controller determines that the active controller does not have all the required resources and allows the role change. • If the active controller does not respond, or if the RMI link is down, the standby controller proceeds with the role change because it has all the resources required to become active.
4	Up	P-Reachable	G-Unreachable	G-Unreachable	No SSO	The standby controller is not ready for SSO in this state because it does not have gateway reachability. The standby controller appears in Standby-Recovery mode.	**SpringBack:** If the gateway reachability is restored on the Standby-Recovery controller (G_Reachable), the controller transitions to the standby state.

Number	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
5	Up	P-Unreachable	G-Reachable	G-Reachable	No SSO	No action taken when RMI goes DOWN. There will be no DAD when the RMI link is DOWN.	If gateway reachability (G_Reachable) is lost, the controller transitions to Standby. This situation is managed as case (3) The active controller maintains its state when gateway reachability is lost. No action is taken when the RMI link goes down. Dual-Active Detection (DAD) does not occur when the RMI link is down.
6	Up	P-Unreachable	G-Reachable	G-Unreachable	No SSO	No Action. Standby is not ready for SSO in this state as it does not have gateway reachability. The standby shall be shown to be in standby-recovery mode.	**Spring Back:** If the gateway reachability is restored (G_Reachable), the controller shall go to Standby mode without a reload. There shall be no action if the RMI comes UP.

Number	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
7	Up	P-Unreachable	G-Unreachable	G-Reachable	SSO	A gateway reachability message is also exchanged over the RP link. The Active device reboots so that the Standby device becomes the new Active. The RP link goes down when the Active device reboots. The Stack Manager sends a message over the RP and RMI links to the Standby Controller to initiate the role change. The Standby controller consults the Active device. • If the Active responds, the Standby determines that the Active does not have all resources and allows the role change. • If the Active does not respond–possibly because the RP link is already down–the Standby allows the role change regardless of resource status.	When the active controller reboots, the RP goes down. The Stack Manager sends a message over the RP and RMI link to the standby controller to initiate a role change. The standby controller consults the active controller.

Node	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
8	Up	P-Unreachable	G-Unreachable	G-Unreachable	No SSO	The standby controller is not ready for SSO in this state because it does not have gateway reachability. The standby controller appears in standby-recovery mode. **Spring Back:** If gateway reachability is restored on the standby-recovery controller (G_Reachable), the controller transitions to standby. Refer to step 7 for more details. The active controller does not change its state when gateway reachability is lost. No action occurs if the RMI comes up.	When the Active device reboots, the RP goes down. The Stack Manager sends a message over the RP and RMI links to the Standby Controller to initiate a role change. The Standby Controller consults the Active Controller. • If the Active responds, the Standby deduces that the Active does not have all the required resources and proceeds with the role change. • If the Active does not respond (for example, if the RP is already down), the Standby allows the role change regardless of the resource status.

Number	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
9	Down	P-Reachable	G-Reachable	G-Reachable	No SSO	When the RP is not available, the standby transitions to Standby-Recovery mode. The stack manager requests a role change when the RP goes down. If the RMI is up, the RIF manager sends a message to the active unit to check its status. If a response is received, the standby does not allow the role change and transitions to Standby-Recovery. If there is no response, such as when the active unit is down due to a crash, the role change is allowed. This scenario works differently if the RP goes down before the standby reaches Standby-Hot state. If the RP link goes down before the standby becomes Standby-Hot, the RIF sends a positive response to the stack manager, resulting in a controller reload.	**Spring Back:** If gateway reachability is restored on the Standby-Recovery (G_Reachable), the controller transitions to Standby. In this case, refer to state (7). The Active controller does not change its state when gateway reachability is lost. No action is taken if the RMI comes up.

Number	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
10	Down	P-Reachable	G-Reachable	G-Unreachable	No SSO		When the RP goes down, the stack manager requests a role change. While the RMI is operational, the RIF manager sends a message to the active controller to verify its status. If a response is received, the standby controller prevents the role change and transitions to standby-recovery. If there is no response, such as when the active controller is down due to a crash, the role change is permitted. However, if the RP goes down before the standby controller reaches the standby-hot state, the RIF manager sends a positive response to the stack manager, which results in a controller reload.

Node	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
						The standby is not ready for SSO in this state because it does not have gateway reachability. The standby will appear in standby-recovery mode. There are two possible scenarios: • The RP goes down first, followed by the standby gateway. • The standby gateway goes down first, followed by the RP. Consider the case where the RP goes down first. In this situation, the stack manager requests a role change. However, because the standby does not have gateway reachability, it cannot allow the role change. The system starts a 30-minute timer when the RMI goes down (meaning both the RP and RMI are down). If the RP goes down before the standby is in standby-hot state, the system reloads. There are several sub-cases: • If the active unit crashes and returns within 30 minutes, the timer stops. The standby remains	

Node	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
						in recovery and reboots when the RP is up. • If the RP stays down, no action is taken when the timer expires, provided the RMI is up. • If the active unit continuously crashes, the timer expires with the RMI down, and the standby-recovery unit reboots as the active unit. Spring Back: • If the gateway returns first, the standby-recovery unit remains in recovery. • If the RP returns first, the system reboots to standby-recovery or standby, depending on whether the gateway is reachable.	

Number	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
11	Down	P-Reachable	G-Unreachable	G-Unreachable	No SSO		

Node	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
						Standby transitions to Standby-Recovery. Assume both controllers lose gateway, then RP goes down. Stack manager requests a role change. Because Standby lacks resources, it starts a 30-minute timer when RMI goes down (that is, RP and RMI are both down). There are three possible outcomes: • If Active recovers within 30 minutes, the timer stops. Standby remains in recovery and may reboot when RP returns. • If RP stays down, no action occurs when the timer expires, provided RMI is up. • If Active never recovers, the timer expires with RMI down, and Standby-Recovery reboots as Active. • If Active never recovers, the timer expires with RMI down, and Standby-Recovery reboots as Active. **Note**	Let us assume that both the controllers lost their GW and then the RP went DOWN. The stack manager will request for a role change when the RP goes DOWN. The standby anyway does not have all resources (Gateway Reachability at present) and hence it shall not allow role change to happen. It will start the 30 min timer when RMI goes DOWN(timer starts when RP+RMI are DOWN).There are now two possibilities: • The active suffered a software glitch (For example: a crash) in which case, it would come up within 30 minutes and the timer would be stopped. The standby will continue to be in standby-recovery. If the RP comes UP when the timer is running, the Standby-Recovery would reboot and might come up as Standby or Standby-Recovery. • Physical RP connection went down and it remains down. When the timer expires, if the RMI is UP, no

Node	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
						If gateway reachability was not enabled, SSO is not allowed when Active is up. If Active is down and Standby is standby-hot, SSO is allowed. If RP returns before standby-hot, it reloads. Note: Recovery to Standby without reload is possible only if recovery was due solely to gateway. **Spring Back:** • If gateway returns first, the system remains in Standby-Recovery. • If RP returns first, the system reboots to Standby-Recovery and then to Standby if gateway is up.	action shall be taken. • The active continuously crashes, that is, it does not come up after 30 minutes. In this case, when the timer expires,the RMI will be DOWN. The standby-recovery shall reboot when the timer expires (and might come UP as Active.) When RP DOWN event is received, if the Gateway Reachability is not enabled, Gateway will not be considered as a resource. In this case, SSO shall not be allowed if the Active is UP. SSO shall be allowed if Active is DOWN, provided Standby is in Standby-Hot state. If the RP link goes down before the standby becomes standby-hot, it shall reload.

Node	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information

Note

The Standby-Recovery that has lost RP is no more Standby Hot. This implies that the recovery from Standby-Recovery to Standby without a reboot (as was the case earlier in 17.2) is not possible for RP events. It is however possible for Gateway events.

Spring Back:

- When the Standby-Recovery findsGateway is UP it continues to be in Standby-Recovery if RP is still DOWN.
- When the Standby-Recovery finds that its RP is UP, it will reboot and come up as Standby-Recovery

Node	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
13	Down	P-Unreachable	G-Reachable	G-Reachable	SSO		

Node	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
						A double fault may result in two active controllers. When this occurs, the Standby controller becomes active, but the original Active controller may still exist. Once connectivity is restored, role negotiation ensures that the most recent Active controller is retained. In the event that RMI goes down and then RP also goes down, the stack manager requests a role change. If RMI is unavailable, Standby grants the role change only if it is in standby-hot mode; otherwise, it denies the request. If RP returns before standby-hot mode is reached, it reloads. **Spring Back:** If RMI returns, the previous Active controller enters Active-Recovery mode. When RP returns, the controller reboots and transitions to Standby. If RP goes down, RMI goes down, and the timer expires, Standby reboots as Active. The timer may be skipped in cases of a pure double fault. **Note**	Let us assume that the RMI goes DOWN first and then the RP goes DOWN. When the RP goes DOWN, the stack manager requests a role change. Since the RMI is DOWN, the standby cannot consult with the Active. The standby allows a role change to become Active, regardless of its resource state, provided the standby is in Standby-Hot. If the standby is not in Standby-Hot, a role change is not allowed. If the RP link goes down before the standby becomes Standby-Hot, the standby reloads **Spring Back:** If the RMI comes UP at any time, Old Active transitions to Active-Recovery. Active-Recovery reboots when the RP comes up, after which it will become Standby. If the RP goes DOWN first, refer to case (9). If RP_DOWN and RMI_DOWN occur in that sequence and the 30-minute timer expires, the standby shall reboot. It will come up as Active if RP and RMI continue to be DOWN. Alternatively, the 30-minute timer may not be started in this case.

Node	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
						You may skip the timer for pure double-fault cases.	The timer can be used when the standby does not have all required resources, such as gateway reachability at present or port status and gateway reachability in the future, to take over as Active. **Note** Another option is to not start the 30-minute timer in this situation. Use the timer only if the standby does not have all the required resources to take over as active. Currently, this refers to gateway reachability; in the future, it may also include port status and gateway reachability.
14	Down	P-Unreachable	G-Reachable	G-Unreachable	No SSO		

Node	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
						Double fault – two active controllers possible. Old Active stays Active; Standby may become Active if connectivity is not restored within a set time. If Standby is in standby-recovery due to GW loss, then RMI goes down, then RP goes down. Stack manager requests role change; no RMI means no consult, so Standby allows change. If Active crashed, it restarts as Standby; if both come up, split-brain conflict may occur.	Let us assume that the Standby is in Standby-Recovery mode as it loses GW. Let us assume that the RMI goes DOWN first and then the RP goes DOWN. The stack manager shall request role change when the RP goes DOWN. Since the RMI isDOWN, the standby cannot consult with the Active. The standby shall allow role change. **Spring Back:** If RMI returns, Old Active enters Active-Recovery and reboots on RP return to become Standby

Number	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
15	Down	P-Unreachable	G-Unreachable	G-Reachable	SSO	Double fault – two active controllers possible. Standby becomes active; old Active may still exist. Role negotiation occurs once connectivity is restored. Assume GW loss on Active, then RMI down then RP down. Stack manager requests role change; no RMI means standby allows change if in standby-hot, else reloads. If RP returns before standby-hot, it reloads. **Spring Back:** If RMI returns, old Active goes to Active-Recovery and reboots on RP return to become Standby.	Suppose the Standby is in Standby-Recovery mode after losing GW. Assume the RMI goes down first, then the RP goes down. The stack manager requests a role change when the RP goes down. Because the RMI is down, the Standby cannot consult with the Active, so it allows the role change. If the Active went down due to a software glitch, it will come up and become Standby. If no communication is established between the two controllers, both may become active, causing a network conflict **Spring Back:** If the RMI comes UP at some point of time,Old Active will go to Active-Recovery. Active-Recovery shall reboot when the RP comes up and will become Standby.

Node	RP Link	Reachability Through RMI	GW From Active	GW From Standby	SSO	Result	Additional Information
16	Down	P-Unreachable	G-Unreachable	G-Unreachable	No SSO	A double fault can result in two active controllers. The old Active remains Active, and the Standby may become Active if connectivity is not restored within a stipulated time. If both controllers lose GW and the Standby is in standby-recovery, then RMI goes down, followed by RP going down. The stack manager requests a role change. If there is no RMI, the Standby allows the change, which can cause a conflict. **Spring Back:** If RMI returns, the old Active enters Active-Recovery and, when RP returns, reboots to become Standby.	Assume that both Active and Standby lose GW, and Standby enters Standby-Recovery. If RMI goes DOWN first, followed by RP going DOWN, the stack manager requests a role change when RP goes DOWN. Since RMI is DOWN, Standby cannot consult with Active and allows the role change. This situation can cause a network conflict. **Spring Back:** If RMI comes UP at any point, the old Active transitions to Active-Recovery. Active-Recovery reboots when RP comes UP and then becomes Standby.

Handling recovery mechanism

This topic explains the Active-to-Active and Standby-to-Standby high availability (HA) recovery mechanisms and describes the system behaviors for each scenario.

This topic provides details about the Active-to-Active and Standby-to-Standby High Availability (HA) recovery mechanisms and the system behaviors for each scenario.

Active-to-Active recovery

- Active recovery occurs if the Route Processor (RP) is down but the Redundancy Management Interface (RMI) is up during boot. The system triggers active recovery after startup.
- In a stable active-standby HA state, if RMI goes down followed by RP, and RMI recovers before RP, the system enters Active-to-Active recovery. After RP is restored, the system reloads active recovery and re-establishes HA.

Standby-to-Standby recovery

- If the Gateway fails, the standby node enters standby recovery mode. In this mode, the standby node stays synchronized with the active unit but cannot become active without the Gateway. When in a hot state, the standby node is ready to take over when Gateway connectivity resumes.

- If only the Gateway is unavailable, restore the Gateway to allow HA to recover without rebooting.
- If a Route Processor (RP) failure triggers standby recovery, wait for the RP to return online. The standby node then reboots automatically, and the system restores HA.

Verify high availability configurations

Verify high availability configurations.

To view the HA configuration details, use this command:

```
Device# show romvar
ROMMON variables:
 LICENSE_BOOT_LEVEL =
 MCP_STARTUP_TRACEFLAGS = 00000000:00000000
 BOOTLDR =
 CRASHINFO = bootflash:crashinfo_RP_00_00_20180202-034353-UTC
 STACK_1_1 = 0_0
 CONFIG_FILE =
 BOOT =
bootflash:boot_image_test,1;bootflash:boot_image_good,1;bootflash:rp_super_universalk9.vwlc.bin,1;

 RET_2_RTS =
 SWITCH_NUMBER = 1
 CHASSIS_HA_REMOTE_IP = 10.0.1.9
 CHASSIS_HA_LOCAL_IP = 10.0.1.10
 CHASSIS_HA_LOCAL_MASK = 255.255.255.0
 CHASSIS_HA_IFNAME = GigabitEthernet2
 CHASSIS_HA_IFMAC = 00:0C:29:C9:12:0B
 RET_2_RCALTS =
 BSI = 0
 RANDOM_NUM = 647419395
```

Verify AP or client SSO statistics

Verify AP or slient SSO statistics.

To view the AP SSO statistics, use this command:

```
Device# show wireless stat redundancy statistics ap-recovery wnc all
AP SSO Statistics

Inst    Timestamp     Dura(ms)   #APs  #Succ  #Fail  Avg(ms)  Min(ms)  Max(ms)
-------------------------------------------------------------------------------

   0    00:06:29.042        98     34     34      0        2        1
35
   1    00:06:29.057        56     33     30      3        1        1
15
   2    00:06:29.070        82     33     33      0        2        1
13


Statistics:

WNCD Instance   : 0
No. of AP radio recovery failures          : 0
No. of AP BSSID recovery failures          : 0
No. of CAPWAP recovery failures            : 0
No. of DTLS recovery failures              : 0
```

```
No. of reconcile message send failed       : 0
No. of reconcile message successfully sent : 34
No. of Mesh BSSID recovery failures: 0
No. of Partial delete cleanup done : 0
.
.
.
```

To view the Client SSO statistics, use this command:

```
Device# show wireless stat redundancy client-recovery wncd all
Client SSO statistics
---------------------

WNCD instance  : 1
Reconcile messages received from AP                    : 1
Reconcile clients received from AP                     : 1
Recreate attempted post switchover                     : 1
Recreate attempted by SANET Lib                        : 0
Recreate attempted by DOT1x Lib                        : 0
Recreate attempted by SISF Lib                         : 0
Recreate attempted by SVC CO Lib                       : 1
Recreate attempted by Unknown Lib                      : 0
Recreate succeeded post switchover                     : 1
Recreate Failed post switchover                        : 0
Stale client entries purged post switchover            : 0

Partial delete during heap recreate                    : 0
Partial delete during force purge                      : 0
Partial delete post restart                            : 0
Partial delete due to AP recovery failure              : 0
Partial delete during reconcilation                    : 0

Client entries in shadow list during SSO               : 0
Client entries in shadow default state during SSO      : 0
Client entries in poison list during SSO               : 0

Invalid bssid during heap recreate                     : 0
Invalid bssid during force purge                       : 0
BSSID mismatch with shadow rec during reconcilation    : 0
BSSID mismatch with shadow rec reconcilation(WGB client): 0
BSSID mismatch with dot11 rec during heap recreate     : 0

AID mismatch with dot11 rec during force purge         : 0
AP slotid mismatch during reconcilation                : 0
Zero aid during heap recreate                          : 0
AID mismatch with shadow rec during reconcilation      : 0
AP slotid mismatch shadow rec during reconcilation     : 0
Client shadow record not present                       : 0
```

To view the mobility details, use this command:

```
Device# show wireless stat redundancy client-recovery mobilityd
Mobility Client Deletion Reason Statistics
------------------------------------------
Mobility Incomplete State         : 0
Inconsistency in WNCD & Mobility  : 0
Partial Delete                    : 0

General statistics
--------------------
```

```
Cleanup sent to WNCD, Missing Delete case   : 0
```

To view the Client SSO statistics for SISF, use this command:

```
Device# show wireless stat redundancy client-recovery sisf
Client SSO statistics for SISF
------------------------------
Number of recreate attempted post switchover    : 1
Number of recreate succeeded post switchover    : 1
Number of recreate failed because of no mac     : 0
Number of recreate failed because of no ip      : 0
Number of ipv4 entry recreate success           : 1
Number of ipv4 entry recreate failed            : 0
Number of ipv6 entry recreate success           : 0
Number of ipv6 entry recreate failed            : 0
Number of partial delete received               : 0
Number of client purge attempted                : 0
Number of heap and db entry purge success       : 0
Number of purge success for db entry only       : 0
Number of client purge failed                   : 0
Number of garp sent                             : 1
Number of garp failed                           : 0
Number of IP entries validated in cleanup       : 0
Number of IP entry address errors in cleanup    : 0
Number of IP entry deleted in cleanup           : 0
Number of IP entry delete failed in cleanup     : 0
Number of IP table create callbacks on standby  : 0
Number of IP table modify callbacks on standby  : 0
Number of IP table delete callbacks on standby  : 0
Number of MAC table create callbacks on standby : 1
Number of MAC table modify callbacks on standby : 0
Number of MAC table delete callbacks on standby : 0
```

To view the HA redundancy summary, use this command:

```
Device# show wireless stat redundancy summary
HA redundancy summary
---------------------

AP recovery duration (ms)        : 264
SSO HA sync timer expired        : No
```

Verify high availability

Verify high availability.

Table 146: Commands for monitoring chassis and redundancy

Command Name	Description
show chassis	Displays the chassis information. **Note** When the peer timeout and retries are configured, the **show chassis ha-status** command output may show incorrect values. To check the peer keep-alive timer and retries, use these commands: • **show platform software stack-mgr chassis active r0 peer-timeout** • **show platform software stack-mgr chassis standby r0 peer-timeout**
show redundancy	Displays details about Active box and Standby box.
show redundancy switchover history	Displays the switchover counts, switchover reason, and the switchover time.

To start the packet capture in the redundancy HA port (RP), use these commands:

- test wireless redundancy packet dump start
- test wireless redundancy packet dump stop
- test wireless redundancy packet dump start filter port 2300

```
Device# test wireless redundancy packetdump start
Redundancy Port PacketDump Start
Packet capture started on RP port.

Device# test wireless redundancy packetdump stop
Redundancy Port PacketDump Start
Packet capture started on RP port.
Redundancy Port PacketDump Stop
Packet capture stopped on RP port.
Device# dir bootflash:
Directory of bootflash:/
1062881  drwx           151552  Oct 20 2020 23:15:25 +00:00  tracelogs
47       -rw-          20480  Oct 20 2020 23:15:24 +00:00  haIntCaptureLo.pcap
1177345  drwx             4096  Oct 20 2020 19:56:14 +00:00  certs
294337   drwx             8192  Oct 20 2020 19:56:05 +00:00  license_evlog
15       -rw-              676  Oct 20 2020 19:56:01 +00:00  vlan.dat
14       -rw-               30  Oct 20 2020 19:55:16 +00:00
throughput_monitor_params
13       -rw-           134808  Oct 20 2020 19:54:57 +00:00  memleak.tcl
1586145  drwx             4096  Oct 20 2020 19:54:45 +00:00  .inv
1103761  drwx             4096  Oct 20 2020 19:54:39 +00:00  dc_profile_dir
17       -r--              114  Oct 20 2020 19:54:17 +00:00  debug.conf
```

```
1389921  drwx             4096  Oct 20 2020 19:54:17 +00:00  .installer
46       -rw-       1104760207  Oct 20 2020 19:26:41 +00:00
leela_katar_rping_test.SSA.bin
49057    drwx             4096  Oct 20 2020 16:11:21 +00:00  .prst_sync
45       -rw-       1104803200  Oct 20 2020 15:39:19 +00:00
C9800-L-universalk9_wlc.2020-10-20_14.57_yavadhan.SSA.bin
269809   drwx             4096  Oct 19 2020 23:41:49 +00:00  core
44       -rw-       1104751981  Oct 19 2020 17:42:12 +00:00
C9800-L-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20201018_053825_2.SSA.bin
43       -rw-       1104286975  Oct 16 2020 12:05:47 +00:00
C9800-L-universalk9_wlc.BLD_POLARIS_DEV_LATEST_20201010_001654_2.SSA.bin

Device# test wireless redundancy packetdump start filter port 2300
Redundancy Port PacketDump Start
Packet capture started on RP port with port filter 2300.
```

To check connection between the two HA Ports (RP) and check if there are any drops, delays, or jitter in the connection, use this command:

```
Device# test wireless redundancy rping
Redundancy Port ping
PING 169.254.64.60 (169.254.64.60) 56(84) bytes of data.
64 bytes from 169.254.64.60: icmp_seq=1 ttl=64 time=0.083 ms
64 bytes from 169.254.64.60: icmp_seq=2 ttl=64 time=0.091 ms
64 bytes from 169.254.64.60: icmp_seq=3 ttl=64 time=0.074 ms

--- 169.254.64.60 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2041ms
rtt min/avg/max/mdev = 0.074/0.082/0.091/0.007 ms
test wireless redundancy
```

To see the HA port interface setting status, use the **show platform hardware slot R0 ha_port interface stats** command.

```
Device# show platform hardware slot R0 ha_port interface stats
HA Port
ha_port   Link encap:Ethernet  HWaddr 70:18:a7:c8:80:70
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
          Memory:e0900000-e0920000

Settings for ha_port:
        Supported ports:            [ TP ]
        Supported link modes:       10baseT/Half 10baseT/Full
                                    100baseT/Half 100baseT/Full
                                    1000baseT/Full
        Supported pause frame use:   Symmetric
        Supports auto-negotiation:   Yes
        Supported FEC modes:         Not reported
        Advertised link modes:       10baseT/Half 10baseT/Full
                                     100baseT/Half 100baseT/Full
                                     1000baseT/Full
        Advertised pause frame use:  Symmetric
        Advertised auto-negotiation: Yes
        Advertised FEC modes:        Not reported
        Speed:                       Unknown!
        Duplex:                      Unknown! (255)
        Port:                        Twisted Pair
```

```
        PHYAD:                          1
        Transceiver:                    internal
        Auto-negotiation:               on
        MDI-X:                          off (auto)
        Supports Wake-on:               pumbg
        Wake-on:                        g
        Current message level:          0x00000007 (7)
                                        drv probe link
        Link detected:                  no

NIC statistics:
     rx_packets:             0
     tx_packets:             0
     rx_bytes:               0
     tx_bytes:               0
     rx_broadcast:           0
     tx_broadcast:           0
     rx_multicast:           0
     tx_multicast:           0
     multicast:              0
     collisions:             0
     rx_crc_errors:          0
     rx_no_buffer_count:     0
     rx_missed_errors:       0
     tx_aborted_errors:      0
     tx_carrier_errors:      0
     tx_window_errors:       0
     tx_abort_late_coll:     0
     tx_deferred_ok:         0
     tx_single_coll_ok:      0
     tx_multi_coll_ok:       0
     tx_timeout_count:       0
     rx_long_length_errors:  0
     rx_short_length_errors: 0
     rx_align_errors:        0
     tx_tcp_seg_good:        0
     tx_tcp_seg_failed:      0
     rx_flow_control_xon:    0
     rx_flow_control_xoff:   0
     tx_flow_control_xon:    0
     tx_flow_control_xoff:   0
     rx_long_byte_count:     0
     tx_dma_out_of_sync:     0
     tx_smbus:               0
     rx_smbus:               0
     dropped_smbus:          0
     os2bmc_rx_by_bmc:       0
     os2bmc_tx_by_bmc:       0
     os2bmc_tx_by_host:      0
     os2bmc_rx_by_host:      0
     tx_hwtstamp_timeouts:   0
     rx_hwtstamp_cleared:    0
     rx_errors:              0
     tx_errors:              0
     tx_dropped:             0
     rx_length_errors:       0
     rx_over_errors:         0
     rx_frame_errors:        0
     rx_fifo_errors:         0
     tx_fifo_errors:         0
     tx_heartbeat_errors:    0
     tx_queue_0_packets:     0
```

```
      tx_queue_0_bytes:       0
      tx_queue_0_restart:     0
      tx_queue_1_packets:     0
      tx_queue_1_bytes:       0
      tx_queue_1_restart:     0
      rx_queue_0_packets:     0
      rx_queue_0_bytes:       0
      rx_queue_0_drops:       0
      rx_queue_0_csum_err:    0
      rx_queue_0_alloc_failed:0
      rx_queue_1_packets:     0
      rx_queue_1_bytes:       0
      rx_queue_1_drops:       0
      rx_queue_1_csum_err:    0
      rx_queue_1_alloc_failed:0
```

ACI network controllers

Explains Cisco Application Centric Infrastructure (ACI) network controllers, their architecture, deployment topologies, and integrated mechanisms for ensuring reliable high-availability failover.

A Cisco ACI network controller is a data center infrastructure platform that provides the following capabilities:

- integrates virtual and physical workloads across a programmable, multi-hypervisor fabric,
- supports multiservice and cloud data center environments, and
- uses a spine-and-leaf switch topology that the controller provisions and manages as a unified entity.

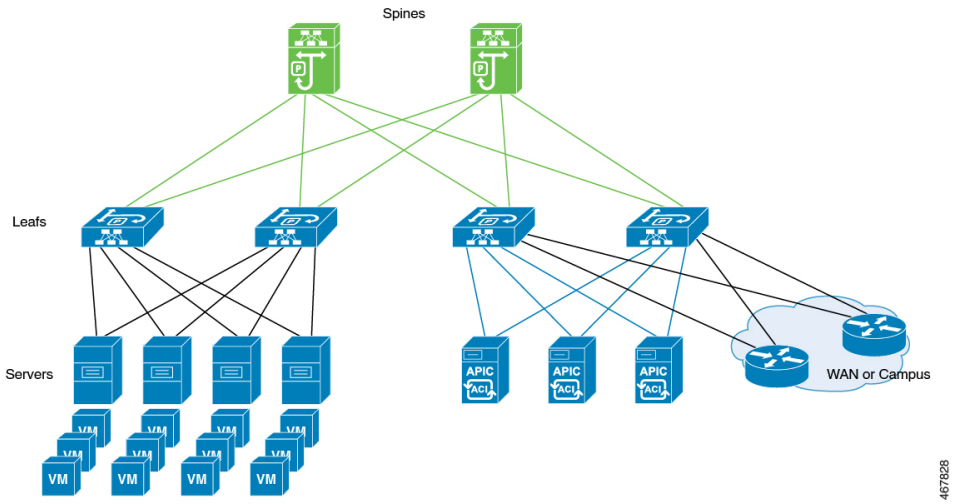
Additional reference information

Use Cisco Application Centric Infrastructure (ACI) technology to combine physical and virtualized environments so you can deploy programmable networking in your data center. ACI controllers help you automate deployment, streamline management, and achieve high availability and scalability.

You can use Cisco ACI in networks with the Redundancy Management Interface (RMI) to ensure resilience for mission-critical workloads.

This diagram shows separate components that the ACI controller connects in a spine-and-leaf topology and manages as a single infrastructure fabric.

Figure 30: Cisco ACI Network Deployment



Example: Avoid interleaving traffic in ACI deployments

In a high-availability data center, Cisco ACI controllers integrate multiple workloads over a unified spine-and-leaf fabric. To maximize service continuity and avoid traffic conflicts during failover, the organization accelerates the shutdown of wireless management interfaces when it detects failure.

It also disables fast switchover notifications to manage the timing of the switchover, tunes the keepalive timeout, and uses GARP burst mitigation to prevent ARP instability. These combined mechanisms ensure seamless and reliable control plane transitions.

Methods to prevent interleaving traffic during controller switchover

Provides methods to prevent interleaving traffic during controller switchover in ACI deployments, including best practices for managing wireless management interfaces, fast switchover notifications, and GARP mitigation.

To ensure a seamless controller switchover in ACI deployments and prevent disruptions for access points (APs) and clients, avoid interleaving traffic—that is, traffic originating from both the old and new active controllers at the same time. The following methods help eliminate management IP conflicts and ARP table instability during these events.

Bring down the wireless management interface faster

When a switchover occurs, APs and clients may disconnect if both controllers send traffic simultaneously. To prevent this, bring down the wireless management interface on the old active controller as soon as a failure is detected. This stops outgoing traffic from the previous controller and prevents management IP conflicts within the network. The standby controller then assumes the active role with a new IP-MAC binding.

The IP Data-Plane Learning feature in ACI tracks duplicate MAC addresses for the same IP address. It can trigger alarms and block affected IP addresses for a configurable period as a protective measure. When a failure is detected, the controller assigns the “non-participant” property to the affected chassis. The Data-Plane Learning feature monitors this status and quickly brings down the old management interface while blocking unwanted traffic, eliminating conflicts between old and new controllers.

Disable fast switchover notification

This mechanism provides more control during failover events. Normally, when a failure is detected, the active controller sends an explicit notification to the standby controller, enabling it to take over immediately. If you disable the fast switchover notification, the active controller no longer sends this alert. The standby controller instead detects the failure based solely on the keepalive timeout.

This configuration lets you control when traffic from the new active controller begins during a failure scenario. Adjusting the keepalive timeout allows you to match your operational needs. However, switchover will wait for the timeout to expire, which introduces a delay. This delay reduces the chance of overlapping traffic and minimizes the risk that both old and new controllers send management traffic at the same time.

GARP burst mitigation

During a controller switchover, a burst of Gratuitous ARP (GARP) traffic may occur. This burst can overwhelm ARP learning in ACI and destabilize ARP tables. To prevent this, the system retransmits GARP packets at a lower rate after the switchover. Lowering the GARP packet rate prevents excessive ARP table churn and supports smoother and more reliable controller transitions.

Best practices for deploying the ACI network in the controller

Use these guidelines when deploying the ACI network in the controller to ensure that the maximum supported clients and endpoint limits are not exceeded.

Check the maximum supported clients in High Availability to ensure that Cisco ACI does not exceed the configured IPv4 and IPv6 end points.

Disable the fast switchover notification mechanism (CLI)

Learn how to disable the fast switchover notification mechanism using CLI commands.

Use these instructions to disable the fast switchover notification mechanism on your network device using CLI. Disabling this feature stops your device from sending explicit fast switchover notifications, which may help in some operational scenarios.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Disable explicit fast switchover notification.

   Example

   ```
   Device(config)# no redun-management fast-switchover
   ```

   Note

   Configure the fast switchover notification mechanism only on the primary controller. You do not need to configure it on the secondary controller.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

After you complete this procedure, your device no longer sends explicit notifications for fast switchover events. Make configuration changes only on the primary controller. You do not need to make changes on the secondary controller.

Configure the gratuitous ARP (GARP) retransmit (CLI)

Learn how to configure Gratuitous ARP (GARP) retransmit using CLI commands to control the rate and interval of GARP resend operations.

Set the rate and interval for sending gratuitous ARP (GARP) packets to enhance network redundancy and update address tables.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Determine the rate at which the GARP resend performs.

Example

```
Device(config)# redun-management garp-retransmit burst packet-burst-size
interval time-interval
```

Note

- *packet-burst-size* : The valid range is from 0 to 1000. The value 0 refers to the disabled retransmit.
- *time-interval* : Refers to the time interval, in seconds. The valid range is from 0 to 5 seconds. The value 0 refers to the disabled retransmit.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

The GARP retransmit settings are updated for the device.

Disable the initial GARP (CLI)

Learn how to disable the initial GARP using CLI commands on your device.

Disable the initial GARP retransmit mechanism in redundancy management to optimize ARP traffic and prevent unnecessary broadcasts.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Disable the initial GARP.

Example

```
Device(config)# no redun-management garp-retransmit initial
```

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

The device will no longer send initial gratuitous ARP messages during redundancy failover, helping to minimize broadcast traffic in your network.

Configure a switchover

Configure a switchover using the redundancy force-switchover command to test high availability on Cisco Catalyst 9800 series wireless controllers.

Enable manual failover to the standby unit and verify that high availability and redundancy work as expected.

Procedure

Force a failover to the standby unit by entering this command:

Example

```
Device#redundancy force-switchover
```

After you enter this command, the standby controller takes the active role. The active controller reloads and becomes the standby controller. Use this command to test high availability cluster stability and confirm that switchovers work as expected.

Note

Use only the recommended command to test switchovers between Cisco Catalyst 9800 series wireless controllers. Using other commands, such as **reload slot X** (where X is the active controller), might cause unexpected behavior.

Note

In a scaled environment, avoid performing an immediate switchover after modifying WLAN or policy profile configurations. Doing so might cause unexpected behavior.

The system completes the switchover. The standby controller becomes active, and the previous active controller reloads to become standby. The switchover verifies high availability cluster stability.

Auto-upgrade feature

Explains the Auto-Upgrade feature that enables the standby controller to upgrade with the software image of the active controller so that both controllers form an HA pair.

The Auto-Upgrade feature is a wireless controller capability that

- enables the standby controller to upgrade with the software image of the active controller
- allows both controllers to form an HA pair, and
- supports the active controller in INSTALL mode.

Auto-upgrade feature requirements

The Auto-Upgrade feature has the following requirements:

- This feature supports the active controller in INSTALL mode.
- This feature supports Cisco Catalyst 9800 Series Wireless Controller software versions 17.5.1 and later.
- This feature is triggered in the standby controller only when the active image is in committed state.

Auto-upgrade use cases

Describes the use cases and functionalities supported by the Auto-Upgrade feature.

Auto-upgrade use cases are scenarios that

- handle software version mismatches during upgrades by copying packages between redundancy ports
- support HA pair deployments where controllers may have different versions, and
- manage SMU installations when standby controllers come online after being offline.

Auto-upgrade functionality details

The following are the use cases and functionalities supported by the Auto-Upgrade feature:

- Handling software version mismatch: During an upgrade, if one of the redundancy port is upgraded to a newer version, and the other one is not upgraded at the same time, the active port tries to copy its packages to the other port using the Auto-Upgrade feature. You can enable Auto-Upgrade in this situation using configuration or by manually running the **software auto-upgrade enable** privileged EXEC command.

  The auto-upgrade configuration is enabled by default.

  > **Note**
  >
  > Auto-upgrade upgrades the mismatched redundancy port only when both the active redundancy port and the mismatched redundancy port are in INSTALL mode.

- HA pair: If one of the controller is not upgraded successfully, use Auto-Upgrade to upgrade the controller on the newly deployed HA pair, which can each be a different version.
- SMUs (APSP, APDP, and so on): If the SMUs that are successfully installed on the active controller when the standby controller was offline. In this scenario, when the standby controller comes up online, the Auto-Upgrade copies this SMU to the standby controller and installs it.

Configuration workflow

Configure workflow.

#unique_2256

Configure auto-upgrade (CLI)

Configure auto-upgrade using commands.

Enable the Auto-Upgrade feature to automatically upgrade software when new versions become available.

The Auto-Upgrade feature is enabled by default but can be disabled. When disabled, manual upgrades must be performed using the install autoupgrade command.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable the Auto-Upgrade feature.

   Example

   ```
   Device(config)# software auto-upgrade enable
   ```

   This feature is enabled by default.

   If you disable this feature using the no form of this command, you need to manually auto upgrade using the install autoupgrade command in privileged EXEC mode.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

Auto-upgrade is now enabled and will automatically upgrade software when new versions are available.

Link layer discovery protocol use cases

Describes Link Layer Discovery Protocol (LLDP) use cases in high-availability setups.

A Link Layer Discovery Protocol (LLDP) use case is a network topology scenario that demonstrates how LLDP operates independently in high-availability wireless unit configurations where active and standby units maintain separate protocol instances.

High-availability LLDP operation

In a high-availability (HA) setup, when two wireless units act as active and standby, the LLDP still runs independently in both.

When you execute the LLDP neighbors command, the system name as the neighbor entry in the uplink switch is displayed as *hostname*-**stbdy**.

Enable LLDP (CLI)

Configure LLDP using commands.

Enable Link Layer Discovery Protocol (LLDP) to allow network devices to advertise information about themselves to other devices on the network.

LLDP is used for network topology discovery and network management. Enable LLDP when you need devices to share information such as system name, port description, and VLAN configuration with neighboring devices.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable Link Layer Discovery Protocol (LLDP).

   Example

   ```
   Device(config)# lldp run
   ```

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

LLDP is now enabled and the device will begin advertising and receiving LLDP information from neighboring devices.

Enable LLDP timers (CLI)

Configure LLDP timers using commands.

Configure LLDP timer settings to control packet transmission rates and retention periods.

LLDP timers control how frequently LLDP packets are sent, how long receivers keep packets, and initialization delays. These settings optimize LLDP performance for your network environment.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable LLDP holdtime timer.

Example

```
Device(config)# lldp holdtime time-in-seconds
```

Example

```
Device(config)# lldp holdtime 100
```

The timer decides how long the receiver must keep the packet. Valid range is from 0 to 65535 seconds.

3. Specify the delay for LLDP to initialize.

Example

```
Device(config)# lldp reinit delay-in-seconds
```

Example

```
Device(config)# lldp reinit 3
```

Valid range is from 2 to 5 seconds.

4. Specify the rate at which LLDP packets are sent.

Example

```
Device(config)# lldp timer time-in-seconds
```

Example

```
Device(config)# lldp timer 7
```

Valid range is from 5 to 65534 seconds.

5. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

LLDP timers are now configured with the specified holdtime, reinitialization delay, and transmission rate settings.

Enable LLDP TLV-Select (CLI)

Configure LLDP TLV-Select using commands.

Enable specific LLDP TLV types to control what information is advertised by the device.

LLDP TLV-Select allows you to control which Type-Length-Value elements are included in LLDP advertisements, providing granular control over the information shared with neighboring devices.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable type, length, and value (TLV) selection for LLDP.

   Example

   ```
   Device(config)# lldp tlv-select tlv-type
   ```

   Example

   ```
   Device(config)# lldp tlv-select port-vlan
   ```

   Available TLV types:

   - **mac-phy-cfg**: IEEE 802.3 MAC, physical configuration, or status TLV.
   - **management-address**: Management address TLV.
   - **port-description**: Port description TLV.
   - **port-vlan**: Port VLAN ID TLV.
   - **system-capabilities**: System capabilities TLV.
   - **system-description**: System description TLV.

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

The specified LLDP TLV type is now enabled and will be included in LLDP advertisements sent by the device.

LLDP verification

Lists the show commands to verify LLDP details in active and standby controllers.

Use the following **show** commands to view the LLDP details independently in the active and standby controller.

Timer and status verification

To verify the timer and status in the active and standby controller, use the following command:

```
Device# show lldp
Global LLDP Information:
    Status: ACTIVE
    LLDP advertisements are sent every 30 seconds
```

```
    LLDP hold time advertised is 120 seconds
    LLDP interface reinitialisation delay is 2 seconds
```

Neighbor details verification

To verify the neighbor details in the active controller, use the following command:

```
Device# show lldp neighbors
Capability codes:
    (R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
    (W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other
Device ID           Local Intf     Hold-time  Capability      Port ID
9500-SW             Tw0/0/0        120        B,R             Twe1/0/14
```

To verify the neighbor details in the standby controller, use the following command:

```
Device# show lldp neighbors
Capability codes:
(R) Router, (B) Bridge, (T) Telephone, (C) DOCSIS Cable Device
(W) WLAN Access Point, (P) Repeater, (S) Station, (O) Other
Device ID           Local Intf     Hold-time  Capability      Port ID
9500-SW             Tw0/0/0        120        B,R             Twe1/0/13
Total entries displayed: 1
```

LLDP neighbor TLV details verification

To verify the LLDP neighbor (TLV) detail, use the following command:

```
Device# show lldp neighbors detail
------------------------------------------------
Local Intf: Te0/0/0
Chassis id: 2cd0.2d62.be80
Port id: Te1/1
Port Description: TenGigabitEthernet1/1
System Name: HSRP-ROUTER-1-15.cisco.com

System Description:
Cisco IOS Software, IOS-XE Software, Catalyst 4500 L3 Switch  Software
(cat4500e-UNIVERSAL-M), Version 03.09.00.E RELEASE SOFTWARE (fc1)
Technical Support: http://www.cisco.com/techsupport
Copyright (c) 1986-2016 by Cisco Systems, Inc.
Compiled Tue 19-Jul

Time remaining: 99 seconds
System Capabilities: B,R
Enabled Capabilities: B,R
Management Addresses:
    IP: 8.109.0.1
    IPV6: 2001:12:1::2
Auto Negotiation - not supported
Physical media capabilities:
    Other/unknown
Media Attachment Unit type - not advertised
Vlan ID: 109
Peer Source MAC: 2cd0.2d62.be80
```

Uplink switch LLDP details verification

To verify the LLDP details in the uplink switch, use the following command:

```
Device# show lldp neighbors detail
------------------------------------------------
Local Intf: Te1/1
Chassis id: d4e8.80b3.0420
Port id: Te0/0/0
Port Description: TenGigabitEthernet0/0/0
System Name: WLC-BGL15.cisco.com

System Description:
Cisco IOS Software [Bangalore], C9800 Software (C9800_IOSXE-K9), Experimental
 Version 17.9.20220630:200739
Copyright (c) 1986-2022 by Cisco Systems, Inc.
Compiled Thu 30-Jun-22 13:19

Time remaining: 107 seconds
System Capabilities: B,R
Enabled Capabilities: R
Management Addresses:
    IP: 8.109.0.47
    IPV6: FD09:8:109::45
Auto Negotiation - not supported
Physical media capabilities - not advertised
Media Attachment Unit type - not advertised
Vlan ID: 109
```

LLDP packet errors verification

To verify LLDP packet errors, use the following command:

```
Device# show lldp errors
LLDP errors/overflows:
Total memory allocation failures: 0
Total encapsulation failures: 0
Total input queue overflows: 0
Total table overflows: 0
```

LLDP traffic statistics verification

To verify LLDP traffic statistics, use the following command:

```
Device# show lldp traffic
LLDP traffic statistics:
Total frames out: 18470
Total entries aged: 0
Total frames in: 6156
Total frames received in error: 0
Total frames discarded: 0
Total TLVs discarded: 0
Total TLVs unrecognized: 0
```

Reload reason history

Describes a diagnostic and serviceability feature that records controller reload events and maintains historical data for troubleshooting purposes.

Reload reason history is a diagnostic and serviceability feature that

- records the reasons for controller reloads
- maintains a log of the last ten reload events, and
- provides access to this data through CLI commands or programmable interfaces like NETCONF/YANG.

This feature is useful for troubleshooting abnormal reboots and understanding system behavior in production environments. Cisco introduced it in IOS XE Dublin 17.11.1, enhancing previous capabilities that only recorded the last reload reason.

View reload reason history

View the history using the **show version** command and the Network Configuration Protocol (NETCONF).

> **Note**
>
> When you reload the standby controller, the system report files are generated in the controller hard disk.

Feature history

Lists feature history for reload reason history.

This reference provides the feature history for reload reason history functionality.

Table 147: Feature history for reload reason history

Feature name	Release information	Feature description
Reload reason history	Cisco IOS XE 17.11.1	The reload reason history feature tracks the reasons for controller reload. This is done for the last 10 reloads. In Cisco IOS-XE 17.10.x and earlier releases, it was possible to track only the reason for the last reload.

Request reload reason history using YANG

Configure YANG with NETCONF and RESTCONF to request reload reason history for automated and programmable network operations.

Use YANG with NETCONF and RESTCONF to provide the desired solution for automated and programmable network operations.

Procedure

Use RPC to create a NETCONF GET request for reload history data.

Example

```
<nc:rpc xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0"
```

```
message-id="urn:uuid:da15955f-5bb7-437c-aeb5-0fc7901a1e9e">
  <nc:get>
    <nc:filter>
      <device-hardware-data
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-device-hardware-oper">
        <device-hardware>
          <device-system-data>
            <reload-history/>
          </device-system-data>
        </device-hardware>
      </device-hardware-data>
    </nc:filter>
  </nc:get>
</nc:rpc>

<rpc-reply message-id="urn:uuid:da15955f-5bb7-437c-aeb5-0fc7901a1e9e"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <device-hardware-data
xmlns="http://cisco.com/ns/yang/Cisco-IOS-XE-device-hardware-oper">
      <device-hardware>
        <device-system-data>
          <reload-history>
            <rl-history>
              <reload-category>rc-rld</reload-category>
              <reload-desc>Reload Command</reload-desc>
              <reload-time>2022-11-30T01:33:44+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-crit-proc-fault</reload-category>
               <reload-desc>Critical process stack_mgr fault on rp_0_0 (rc=137),
 system report at
bootflash:core/Yang_Test-system-report_20221130-012929-UTC.tar.gz</reload-desc>

              <reload-time>2022-11-30T01:31:11+00:00</reload-time>
              <reload-severity>abnormal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-img-install</reload-category>
              <reload-desc>Image Install </reload-desc>
              <reload-time>2022-11-30T01:25:03+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-crit-proc-fault</reload-category>
              <reload-desc>Critical process rif_mgr fault on rp_0_0 (rc=137),
 system report at
bootflash:core/Yang_Test-system-report_20221130-011127-UTC.tar.gz</reload-desc>

              <reload-time>2022-11-30T01:13:08+00:00</reload-time>
              <reload-severity>abnormal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-rld</reload-category>
              <reload-desc>Reload Command</reload-desc>
              <reload-time>2022-11-30T01:08:26+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-crit-proc-fault</reload-category>
```

```
              <reload-desc>Critical process wncmgrd fault on rp_0_0 (rc=137),
 system report at
bootflash:core/Yang_Test-system-report_20221130-010338-UTC.tar.gz</reload-desc>

              <reload-time>2022-11-30T01:05:23+00:00</reload-time>
              <reload-severity>abnormal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-rld</reload-category>
              <reload-desc>Reload Command</reload-desc>
              <reload-time>2022-11-30T01:01:09+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-rld</reload-category>
              <reload-desc>Reload Command</reload-desc>
              <reload-time>2022-11-30T00:57:27+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-rld</reload-category>
              <reload-desc>Reload Command</reload-desc>
              <reload-time>2022-11-30T00:22:34+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
            <rl-history>
              <reload-category>rc-force-switchover</reload-category>
              <reload-desc>redundancy force-switchover</reload-desc>
              <reload-time>2022-11-29T23:40:01+00:00</reload-time>
              <reload-severity>normal</reload-severity>
            </rl-history>
          </reload-history>
        </device-system-data>
      </device-hardware>
    </device-hardware-data>
  </data>
</rpc-reply>
```

A NETCONF GET request for reload history data is created.

What to do next

Formore information about the YANG models, see these documents:

- The Cisco IOS XE Programmability Configuration Guide at https://www.cisco.com/c/en/us/support/wireless/catalyst-9800-series-wireless-controllers/products-installation-and-configuration-guides-list.html
- The YANG Data Models on Github at https://github.com/YangModels/yang/tree/main/vendor/cisco/xe.
- Contact the Developer Support Community for NETCONF and YANG features at: https://developer.cisco.com/

Reload reason history verification

Lists commands to verify reload history and reason details in the controller console.

Verify reload history and reason details using specific commands to troubleshoot controller issues and understand system behavior.

View reload history details

To view the reload history details, use this command:

```
Device# show reload-history

Reload History:

Reload Index: 1
Reload Code: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 01:33:44 UTC Wed Nov 30 2022

Reload Index: 2
Reload Code: Critical Process Fault
Reload Description: Critical process stack_mgr fault on rp_0_0 (rc=137), system
 report at bootflash:core/Yang_Test-system-report_20221130-012929-UTC.tar.gz
Reload Severity: Abnormal Reboot
Reload Time: 01:31:11 UTC Wed Nov 30 2022

Reload Index: 3
Reload Code: Image Install
Reload Description: Image Install
Reload Severity: Normal Reboot
Reload Time: 01:25:03 UTC Wed Nov 30 2022

Reload Index: 4
Reload Code: Critical Process Fault
Reload Description: Critical process rif_mgr fault on rp_0_0 (rc=137), system
 report at bootflash:core/Yang_Test-system-report_20221130-011127-UTC.tar.gz
Reload Severity: Abnormal Reboot
Reload Time: 01:13:08 UTC Wed Nov 30 2022

Reload Index: 5
Reload Code: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 01:08:26 UTC Wed Nov 30 2022

Reload Index: 6
Reload Code: Critical Process Fault
Reload Description: Critical process wncmgrd fault on rp_0_0 (rc=137), system
 report at bootflash:core/Yang_Test-system-report_20221130-010338-UTC.tar.gz
Reload Severity: Abnormal Reboot
Reload Time: 01:05:23 UTC Wed Nov 30 2022

Reload Index: 7
Reload Code: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 01:01:09 UTC Wed Nov 30 2022

Reload Index: 8
Reload Code: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 00:57:27 UTC Wed Nov 30 2022

Reload Index: 9
Reload Code: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
```

```
Reload Time: 00:22:34 UTC Wed Nov 30 2022

Reload Index: 10
Reload Code: Fast Switchover
Reload Description: redundancy force-switchover
Reload Severity: Normal Reboot
Reload Time: 23:40:01 UTC Tue Nov 29 2022
```

View reason for last reload

To view reason for the last reload, use this command:

```
Device# show platform software tdl-database content ios device data
Device Current time: 04:06:04
Device boot time: 01:33:37
Software version: Cisco IOS Software [Dublin], C9800-CL Software
(C9800-CL-K9_IOSXE), Experimental Version 17.11.20221012:120806
[BLD_POLARIS_DEV_S2C_20221010_023625-1-g5ebdd5c35512:/nobackup/saikarth/polaris_relhis
 103]
Copyright (c) 1986-2022 by Cisco Systems, Inc.
Compiled Wed 12-Oct-22 05:08 by saikarth
Rommon version: IOS-XE ROMMON
Last Reboot reason: Reload Command
Reboot reason severity: Normal Reboot
Unsaved configuration:  * Unknown boolean *

Reload History:

Reload Category: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 11/30/2022 01:33:44 UTC

Reload Category: Critical Process Fault
Reload Description: Critical process stack_mgr fault on rp_0_0 (rc=137), system
 report at bootflash:core/Yang_Test-system-report_20221130-012929-UTC.tar.gz
Reload Severity: Abnormal Reboot
Reload Time: 11/30/2022 01:31:11 UTC

Reload Category: Image Install
Reload Description: Image Install
Reload Severity: Normal Reboot
Reload Time: 11/30/2022 01:25:03 UTC

Reload Category: Critical Process Fault
Reload Description: Critical process rif_mgr fault on rp_0_0 (rc=137), system
 report at bootflash:core/Yang_Test-system-report_20221130-011127-UTC.tar.gz
Reload Severity: Abnormal Reboot
Reload Time: 11/30/2022 01:13:08 UTC

Reload Category: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 11/30/2022 01:08:26 UTC

Reload Category: Critical Process Fault
Reload Description: Critical process wncmgrd fault on rp_0_0 (rc=137), system
 report at bootflash:core/Yang_Test-system-report_20221130-010338-UTC.tar.gz
Reload Severity: Abnormal Reboot
Reload Time: 11/30/2022 01:05:23 UTC
```

```
Reload Category: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 11/30/2022 01:01:09 UTC

Reload Category: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 11/30/2022 00:57:27 UTC

Reload Category: Reload
Reload Description: Reload Command
Reload Severity: Normal Reboot
Reload Time: 11/30/2022 00:22:34 UTC

Reload Category: Fast Switchover
Reload Description: redundancy force-switchover
Reload Severity: Normal Reboot
Reload Time: 11/29/2022 23:40:01 UTC
```

173 Redundancy Management Interface

Topics:

- Redundancy management interfaces

Details the Redundancy Management Interface for monitoring gateway statistics, standby controller health, and configuring redundancy parameters across network infrastructure.

Redundancy management interfaces

Explains the foundational concepts and principles of Redundancy Management Interfaces (RMI) for high availability in Cisco Catalyst 9800 Series Wireless Controllers, including pairing scenarios, gateway monitoring, and troubleshooting for IPv4 and IPv6 dual stack environments.

A redundancy management interface is a high availability feature that

- enables pairing scenarios and gateway monitoring for Cisco Catalyst 9800 Series Wireless Controllers
- supports both IPv4 and IPv6 dual stack environments, and
- facilitates dynamic pairing, upgrade/downgrade behaviors, ARP handling, and AAA integration.

Gateway monitoring

Describes a network management capability that automatically selects gateway IP addresses and provides enhanced visibility into gateway reachability and RMI state.

Gateway monitoring is a network management capability that

- selects the gateway IP based on the static routes that match the RMI subnet using the broadest mask and least gateway IP
- provides enhanced visibility and statistics related to gateway reachability and RMI state, and
- simplifies troubleshooting and control over high availability (HA) and RMI functionality by offering detailed diagnostics.

Key aspects

Gateway IP selection:

- From Cisco IOS XE 17.2.1 onwards, the **ip default-gateway gateway-ip** command is deprecated for RMI gateway configuration.
- The system automatically chooses a gateway IP by evaluating static routes and selecting the gateway that:
  - matches the RMI subnet,
  - uses the broadest subnet mask,
  - and has the lowest IP address.
- If no matching static route exists, gateway failover does not operate—even if management gateway-failover is enabled.

Enhanced statistics and transparency:

- From Cisco IOS XE 17.18.1 onwards, gateway monitoring provides improved visibility into gateway reachability, RMI state, and detailed diagnostics.
- **Benefits include:**
  - Enhanced visibility into gateway health and RMI state.
  - Simplified troubleshooting through detailed statistics and state information.
  - Greater transparency and control over HA and RMI, improving reliability and diagnostics.

Refer to the Gateway statistics monitoring on page 2132 for gateway monitoring commands.

Airport's air traffic control system

Gateway monitoring works like an airport's air traffic control system. Instead of a pilot manually selecting a runway (like using the old **ip default-gateway** command), a central system automatically identifies the best runway based on current conditions (route matching, subnet mask, and lowest IP address). It then tracks the success or failure of each takeoff and landing (probe statistics and diagnostics), making sure the whole airport runs smoothly and efficiently while providing real-time updates and troubleshooting support when issues arise.

Configure gateway monitoring interval (CLI)

Configure gateway monitoring interval using commands.

Set the interval at which the management gateway failover feature monitors gateway availability on a switch.

Adjusting the gateway monitoring interval determines how frequently the device checks gateway status to trigger failover if necessary. This procedure is performed using command-line interface (CLI) commands.

Before you begin

- Ensure you have access to the device CLI in privileged EXEC mode.
- Confirm that you have the required permissions to enter configuration mode and modify gateway monitoring settings.

Follow these steps to configure the gateway monitoring interval using the CLI:

Procedure

1. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. **management gateway-failover interval** *interval-value*

   Example

   ```
   Device(config)# management gateway-failover interval 6
   ```

   Configures the gateway monitoring interval.

   interval-value - Refers to the gateway monitoring interval. The valid range is from 6 to 12. Default value is 8.

3. **end**

   Example

   ```
   Device(config)# end
   ```

   Saves the configuration and exits configuration mode and returns to privileged EXEC mode.

The system monitors the gateway at the specified interval. Your configuration is saved, and the device returns to privileged EXEC mode.

Configure gateway monitoring (CLI)

Configure gateway monitoring on a device to ensure effective monitoring and management of gateway connections for smooth network operations.

Enable and configure gateway monitoring on a device.

This task is typically performed to ensure that network devices can effectively monitor and manage gateway connections, facilitating smooth network operations and management. Gateway monitoring involves setting up a default gateway and enabling monitoring features.

Follow these steps to configure gateway monitoring (CLI):

Procedure

1. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. **[no] management gateway-failover enable**

   Example

   ```
   Device(config)# management gateway-failover enable
   ```

   Enables gateway monitoring. (Use the **no** form of this command to disable gateway monitoring.)

3. **end**

   Example

   ```
   Device(config)# end
   ```

   Returns to privileged EXEC mode.

The gateway monitoring is enabled and configured, and the configuration changes are active.

What to do next

To save the configuration, use the **write memory** command.

Gateway statistics monitoring

Provides commands and output for displaying gateway statistics and status information.

To display detailed gateway statistics, use the **show platform software rif-mgr chassis {active | standby } r0 gateway-statistics** command and to display gateway status information, use the **show platform software rif-mgr chassis { active | standby } r0 resource-status** command.

```
Device# show platform software rif-mgr chassis active r0 gateway-statistics
```

```
Number of L2 (ARP/ND) probes sent        : 164890
Number of ICMP probes sent               : 20
Number of L2 (ARP/ND) responses received : 164878
Number of ICMP responses received        : 0
Number of L2 (ARP/ND) responses failed   : 12
Number of ICMP responses failed          : 20
Last fail count                          : 32  at (05/26/25 18:21:33)
Longest fail count                       : 32  at (05/26/25 18:21:33)
```

```
Device# show platform software rif-mgr chassis active r0 resource-status

RIF Resource Status

RP Status  : Up
RMI Status : Up
Current Chassis State : Active
Peer Chassis State    : Standby
Gateway Monitoring Configured  : True
Gateway Monitoring Interval    : 10
Gateway Status      : Up
Peer Gateway Status : Up
```

Gateway reachability detection

Describes a feature that minimizes the downtime on APs and clients when the gateway reachability is lost on the active controller.

Gateway reachability detection is a feature that

- minimizes the downtime on APs and clients when the gateway reachability is lost on the active controller
- enables both active and standby controllers to keep track of gateway reachability by sending Internet Control Message Protocol (ICMP) and ARP requests periodically to the gateway, and
- detects gateway failures within approximately 8 seconds using consecutive failure monitoring.

Gateway reachability detection behavior

Both active and standby controllers use the RMI IP as the source IP. The messages are sent at 1 second interval. If it takes 8 (or configured value) consecutive failures in reaching the gateway, the controller declares the gateway as non-reachable. It takes approximately 8 seconds to detect if a controller has lost gateway reachability.

Gateway monitoring with native IPv6 uses ICMP Neighbor Discovery protocols and ICMPv6 ECHO to check gateway reachability.

Therefore, you can monitor only the IPv6 gateway when RMI IPv6 is configured.

This means that only one IPv4 or IPv6 gateways can be monitored.

> **Note**
>
> If the standby controller loses gateway, the standby moves to the standby recovery mode.
>
> If the active controller loses gateway, the active reloads and standby becomes active.

Requirement: proper configuration of default gateway

Important note regarding default gateway configuration for gateway reachability check

To ensure the gateway reachability check functions correctly, configure the default gateway next hop in the same subnet as the WMI, which also serves the RMI.

If you configure the default gateway with a next hop IP address outside the WMI subnet, the controller does not send packets for the gateway reachability check. This configuration can cause failover or redundancy issues.

Configure system redundancy workflow

Describes the workflow for configuring system redundancy management interfaces and monitoring.

Summary

The key components involved in the configuration workflow are:

- Redundancy Management Interface (RMI): Provides redundancy capabilities through GUI or CLI configuration
- Controllers: Require reloading for RMI configuration to take effect
- IPv6 Static Route: Enables network routing configuration
- Gateway Monitoring Interval: Monitors gateway connectivity at specified intervals

Workflow

The configuration workflow involves these stages:

1. Configure Redundancy Management Interface using either GUI or CLI method

   For more information, see Configure redundancy management interface (GUI) on page 2156.

   > **Note**
   >
   > For RMI configuration to take effect, ensure that you reload your controllers.

2. Configure IPv6 Static Route to establish network routing

   For information, see Gateway monitoring on page 2130.

3. Configure Gateway Monitoring Interval using CLI to set monitoring parameters

   For more information, see Configure gateway monitoring interval (CLI) on page 2131.

Migrate to RMI IPv6

Configure RMI IPv6 migration from either RMI IPv4 or HA pairing without RMI.

You can migrate to RMI IPv6 from two different configurations: from RMI IPv4 or from HA pairing without RMI. The migration process varies depending on your current configuration.

Before you begin

Follow these steps to migrate to RMI IPv6:

Procedure

Determine your current configuration and follow the appropriate migration path.

- If migrating from RMI IPv4, continue with the following substeps:

  a. Unconfigure the RMI IPv4 using the following CLIs:

  ```
  Device# conf t
  Device(config)# no redun-management interface <vlan_name> chassis 1 address
   <ip_address1> chassis 2 address <ip_address2>
  ```

  > **Note**
  >
  > This CLI unconfigures RMI on both the controllers.

  b.

  > **Note**
  >
  > Take a backup of the running config on active before you reload the controller.

  Reload the controller.

  c. Copy the backed up config to the running config on the box which would have lost all the config.

  d. Configure the RMI IPv6 on both the controllers.

  e. Reload the controller.

- If migrating from HA Pairing (without RMI), for information on HA pairing, see Configuring Redundancy Management Interface (GUI).

Standby monitoring

Describes the standby monitoring feature that allows monitoring of a system on a standby controller using programmatic interfaces and commands.

Standby monitoring is a system health feature that

- monitors the health of a system on a standby controller using programmatic interfaces and commands
- allows monitoring of parameters such as CPU, memory, interface status, power supply, fan failure, and the system temperature

- is enabled when Redundancy Management Interface (RMI) is configured, and
- cannot be dynamically enabled or disabled.

Requirements and configuration

Standby Monitoring is enabled when Redundancy Management Interface (RMI) is configured, no other configuration is required. The RMI itself is used to connect to the standby and perform standby monitoring.

> **Note**
>
> The active controller uses the management or RMI IP to initiate AAA requests. Whereas, the standby controller uses the RMI IP to initiate AAA requests. Thus, the RMI IPs must be added in AAA servers for a seamless client authentication and standby monitoring.

To enable standby console, ensure that the following configuration is in place:

```
redundancy
main-cpu
secondary console enable
```

> **Note**
>
> The Standby Monitoring feature is not supported on a controller in the active-recovery and the standby-recovery modes.

The Standby Monitoring feature supports only the following traffic on the RMI interface of the standby controller:

- Address Resolution Protocol (ARP)
- Internet Control Message Protocol (ICMP)
- TCP Traffic (to or from) ports: 22, 443, 830, and 3200
- UDP RADIUS ports: 1645 and 1646
- UDP Extended RADIUS ports: 21645 to 21844

Feature scenarios include:

- Monitoring the health of the standby directly from the standby controller using Standby RMI IP
- Getting syslogs from the standby controller using the Standby RMI IP

Use cases include:

- Enabling SNMP agent and programmatic interfaces on the standby controller: You can directly perform an SNMP query or programmatic interface query to the standby's RMI IP and active controller
- Enabling syslogs on the standby controller: You can directly get the standby syslogs from the standby controller

RADIUS accounting support:

Whenever you log in to a standby device, the RADIUS start record must be sent to the external RADIUS server. Similarly, when you log out of a device, the RADIUS stop record must be sent to the external RADIUS server.

TACACS+ authentication support:

Users are authenticated through the RMI using the external TACACS+ server. The username and password are evaluated in the TACACS+ server. Depending on the response received from the server, a user will be able to log in to the standby device.

TACACS+ accounting support:

Whenever you log in to the standby device, the TACACS+ accounting start record must be sent to the external TACACS+ server. Similarly, when you log out of a device, the TACACS+ accounting stop record must be sent to the external TACACS+ server.

Note

This configuration must be in place to configure AAA to send the accounting packets:

```
aaa accounting exec {default | named-list} start-stop group {RAD |
tac-group-name}
```

Note

The TACACS+ login to the standby device is not supported when TACACS+ server is configured with hostname.

Monitor the health of standby parameters using SNMP

-

Standby monitoring using standby RMI IP

Describes standby monitoring using standby RMI IP.

Standby monitoring using standby RMI IP is a network management capability that

- enables direct SNMP queries to the standby controller's RMI IP address
- supports IF-MIB monitoring for interface statistics on standby controllers from Release 17.5 onwards, and
- automatically enables SNMP on the standby controller when SNMP agent is enabled on the active controller.

Supported mibs and monitoring capabilities

When an SNMP agent is enabled on the standby controller, you can directly perform an SNMP query to the standby's RMI IP. From Release 17.5 onwards, you can query the following MIB on the standby controller:

Table 148: MIB name and notes

MIB Name	Notes
IF-MIB	This MIB is used to monitor the interface statistics of the standby controller using the standby RMI IP address.

Note

If an SNMP agent is enabled on the active controller, by default, the SNMP is enabled on the standby controller.

Standby monitoring using the active controller

Lists MIB objects for monitoring standby controller health parameters through the active controller.

Monitor the health parameters of the standby controller, including memory, CPU, port status, power statistics, and peer gateway latencies, using various MIB objects through the active controller.

CISCO-LWAPP-HA-MIB

The CISCO-LWAPP-HA-MIB monitors the health parameters of the standby controller, that is, memory, CPU, port status, power statistics, peer gateway latencies, and so on.

You can query these MIB objects of CISCO-LWAPP-HA-MIB.

Table 149: MIB objects and notes

MIB Objects	Notes
cLHaPeerHotStandbyEvent	This object can be used to check if the standby controller has turned hot-standby or not.
cLHaBulkSyncCompleteEvent	This object represents the time at which the bulksync is completed.

CISCO-PROCESS-MIB

The CISCO-PROCESS-MIB monitors CPU and process statistics. Use it to monitor CPU-related or memory-related BINOS processes. The standby CISCO-PROCESS-MIB can be monitored using the active controller.

ENTITY-MIB

The ENTITY-MIB is used to monitor hardware details of the active and standby controllers using the active controller.

Note

The standby Route Processor (RP) sensors are appended in the active RP sensors.

Standby IOS linux syslogs

Explains how standby IOS Linux syslogs are relayed and forwarded to external servers in wireless controller HA configurations.

Standby IOS Linux syslogs are a logging mechanism that

- relay logs using the same method as on the active Cisco IOS for wireless controllers
- enable external logging from the standby IOS starting from Release 17.5 onwards, and

- forward all syslogs generated on the standby controller to the configured external server through BINOS processes.

Standby syslog behavior

From Release 17.5 onwards, external logging of syslogs from the standby IOS is enabled. As BINOS processes on standby also forwards the syslogs to Cisco IOS, all the syslogs generated on the standby controller is forwarded to the configured external server.

> **Note**
>
> RMI IP address is used for logging purpose.

The standby controller tries to send syslogs to the IPv4 server because logging is only configured on IPv4 even though IPv4 is not supported by standby. This is the expected behavior when an HA pair is configured with the RMI IPv6 address, the active controller has dual stack, and logging is configured on the IPv4 address

Standby interface status using active SNMP

Lists the MIB objects of CISCO-LWAPP-HA-MIB that can be queried for standby interface information.

The standby interface information is sent to the active controller using IPC in these scenarios:

- When there is a change in the interface status.
- When a new interface is added or deleted on the standby controller.

When the active controller receives the interface information from the standby controller, the active controller's database is populated with the standby interface information.

When an SNMP query is received for the standby interface information, the SNMP handlers corresponding to the CISCO-LWAPP-HA-MIB reads them from the standby interface database on the active and populates the MIB objects in CISCO-LWAPP-HA-MIB.

You can query the following MIB objects of CISCO-LWAPP-HA-MIB.

Table 150: MIB objects of CISCO-LWAPP-HA-MIB

MIB Object	Notes
stbyIfIndex	This is a unique value (greater than zero) for each interface of the standby controller.
stbyIfName	This is the name of the standby interface.
stbyIfPhysAddress	This is the interface address of the standby controller in the protocol sublayer.
stbyifOperStatus	This is the current operational state of the interface in the standby controller.
stbyifAdminStatus	This is the desired state of the interface of the standby controller.

To verify the logging on the active when the standby fails to send interface statistics, use this command:

```
Device# debug snmp ha-chkpt
Device# debug snmp ha-intf_db
```

Standby controller health monitoring using programmatic interfaces

Lists the programmatic interfaces that can be used to monitor the health of standby controller.

You can monitor parameters such as CPU, memory, sensors, and interface status on a standby controller using programmatic interfaces such as NETCONF and RESTCONF. The RMI IP of the standby controller can be used for access to the following operational models:

The models can be accessed through:

- Cisco-IOS-XE-device-hardware-oper.yang
- Cisco-IOS-XE-process-cpu-oper.yang
- Cisco-IOS-XE-platform-software-oper.yang
- Cisco-IOS-XE-process-memory-oper.yang
- Cisco-IOS-XE-interfaces-oper.yang

For more information on the YANG models, see the *Programmability Configuration Guide, Cisco IOS XE 17.3.x*.

Standby controller health monitoring (CLI)

Lists commands to monitor the health of standby controller.

This section describes the different commands that can be used to monitor the standby device.

You can connect to the standby controller through SSH using the RMI IP of the standby controller. The user credentials must have been configured already. Both local authentication and RADIUS authentication are supported.

> **Note**
>
> The **redun-management** command needs to be configured on both the controllers, primary and standby, prior to high availability (HA) pairing.

Port state monitoring

This is a sample output of the **show interfaces** *interface-name* command:

```
Device-standby# show interfaces GigabitEthernet1

GigabitEthernet1 is down, line protocol is down

Shadow state is up, true line protocol is up
  Hardware is CSR vNIC, address is 000c.2909.33c2 (bia 000c.2909.33c2)
  MTU 1500 bytes, BW 1000000 Kbit/sec, DLY 10 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation ARPA, loopback not set
  Keepalive set (10 sec)
  Full Duplex, 1000Mbps, link type is force-up, media type is Virtual
```

```
  output flow-control is unsupported, input flow-control is unsupported
  ARP type: ARPA, ARP Timeout 04:00:00
  Last input 00:00:06, output 00:00:24, output hang never
  Last clearing of "show interface" counters never
  Input queue: 30/375/0/0 (size/max/drops/flushes); Total output drops: 0
  Queueing strategy: fifo
  Output queue: 0/40 (size/max)
  5 minute input rate 389000 bits/sec, 410 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     3696382 packets input, 392617128 bytes, 0 no buffer
     Received 0 broadcasts (0 multicasts)
     0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored
     0 watchdog, 0 multicast, 0 pause input
     18832 packets output, 1218862 bytes, 0 underruns
     Output 0 broadcasts (0 multicasts)
     0 output errors, 0 collisions, 2 interface resets
     3 unknown protocol drops
     0 babbles, 0 late collision, 0 deferred
     0 lost carrier, 0 no carrier, 0 pause output
     0 output buffer failures, 0 output buffers swapped out
```

This is a sample output of the **show ip interface brief** command:

```
Device# show ip interface brief

Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet1       unassigned      YES unset  down                  down

GigabitEthernet0       unassigned      YES NVRAM  administratively down down

Capwap1                unassigned      YES unset  up                    up

Capwap2                unassigned      YES unset  up                    up

Capwap3                unassigned      YES unset  up                    up

Capwap10               unassigned      YES unset  up                    up

Vlan1                  unassigned      YES NVRAM  down                  down

Vlan56                 unassigned      YES unset  down                  down

Vlan111                111.1.1.85      YES NVRAM  up                    up
```

CPU or memory monitoring

This is a sample output of the **show process cpu sorted 5sec** command:

```
Device-standby# show process cpu sorted 5sec

CPU utilization for five seconds: 0%/0%; one minute: 0%; five minutes: 0%
 PID Runtime(ms)     Invoked      uSecs   5Sec   1Min   5Min TTY Process
  10     1576556      281188       5606  0.15%  0.05%  0.05%   0 Check heaps

 232      845057    54261160         15  0.07%  0.05%  0.06%   0 IPAM Manager

 595         177         300        590  0.07%  0.02%  0.01%   2 Virtual Exec
```

```
 138     1685973   108085955         15  0.07%  0.08%  0.08%   0 L2 LISP Punt
 Pro
 193       19644      348767         56  0.07%  0.00%  0.00%   0 DTP Protocol

   5           0           1          0  0.00%  0.00%  0.00%   0 CTS SGACL db
 cor
   4          24          15       1600  0.00%  0.00%  0.00%   0 RF Slave Main
 Th
   6           0           1          0  0.00%  0.00%  0.00%   0 Retransmission
 o
   7           0           1          0  0.00%  0.00%  0.00%   0 IPC ISSU
Dispatc
   2      117631      348801        337  0.00%  0.00%  0.00%   0 Load Meter

   8           0           1          0  0.00%  0.00%  0.00%   0 EDDRI_MAIN
```

To check CPU and memory utilization of binOS processes, run this command:

```
Device-standby# show platform software process slot chassis standby R0 monitor


top - 23:24:14 up 8 days, 3:38, 0 users, load average: 0.69, 0.79, 0.81
Tasks: 433 total, 1 running, 431 sleeping, 1 stopped, 0 zombie
%Cpu(s): 1.7 us, 2.8 sy, 0.0 ni, 95.6 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 32059.2 total, 21953.7 free, 4896.8 used, 5208.6 buff/cache
MiB Swap: 0.0 total, 0.0 free, 0.0 used. 26304.6 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
23565 root 20 0 2347004829924 100840 S 6.7 2.5 550:29 btrace_rotator
23595 root 20 0 1820072 334636 100840 S 3.3 1.0 179:05 wcm
23597 root 20 0 1820072 334636 100840 S 3.3 1.0 179:03 wcm
23599 root 20 0 1820072 334636 100840 S 3.3 1.0 179:04 wcm
23601 root 20 0 1820072 334636 100840 S 3.3 1.0 179:05 wcm
23603 root 20 0 1820072 334636 100840 S 3.3 1.0 179:04 wcm
```

To check the environment temperature and fan status, run this command:

```
Device-standby# show environment summary

 Switch   FAN                FAN                POWER SUPPLY       POWER SUPPLY
          (Intake)           (Exhaust)          1                  2
--------- ------------------ ------------------ ------------------
------------------
 R0       Normal             Normal             Normal             Normal


 Switch   Inlet Temp (C)     Outlet Temp (C)
--------- ------------------ ------------------
 R0       22                 28

 R0          Stby Temp: DDC INormal          22   Celsius (55 ,65 ,75 ,80
)(Celsius)
 R0          Stby Temp: DDC ONormal          28   Celsius (75 ,85 ,95
,100)(Celsius)
```

> **Note**
>
> The command displays both active and standby hardware details.

> **Note**
>
> The **show environment summary** command displays data only for physical appliances such as Cisco Catalyst 9800-80 Wireless Controller, Cisco Catalyst 9800-40 Wireless Controller, Cisco Catalyst 9800-L Wireless Controller, and Cisco Catalyst 9800 Embedded Wireless Controller for Switch. The command does not display data for Cisco Catalyst 9800 Wireless Controller for Cloud.

Gateway-monitoring configuration verification

Provides commands to verify the gateway-monitoring configuration status.

Verify the status of the gateway-monitoring configuration on active and standby controllers using specific show commands.

To verify the status of the gateway-monitoring configuration on an active controller, run this command:

```
Device# show redundancy states

my state = 13 -ACTIVE
peer state = 8 -STANDBY HOT
Mode = Duplex
Unit = Primary
Unit ID = 1

Redundancy Mode (Operational) = sso
Redundancy Mode (Configured) = sso
Redundancy State = sso
Maintenance Mode = Disabled
Manual Swact = enabled
Communications = Up

client count = 129
client_notification_TMR = 30000 milliseconds
RF debug mask = 0x0
Gateway Monitoring = Disabled
Gateway monitoring interval = 8 secs
```

To verify the status of the gateway-monitoring configuration on a standby controller, run this command:

```
Device-stby# show redundancy states

my state = 8 -STANDBY HOT
peer state = 13 -ACTIVE
Mode = Duplex
Unit = Primary
Unit ID = 2

Redundancy Mode (Operational) = sso
Redundancy Mode (Configured) = sso
```

```
Redundancy State = sso
Maintenance Mode = Disabled
Manual Swact = cannot be initiated from this the standby unit
Communications = Up

client count = 129
client_notification_TMR = 30000 milliseconds
RF debug mask = 0x0
Gateway Monitoring = Disabled
Gateway monitoring interval = 8 secs
```

RMI IPv4 configuration verification

Lists commands to verify the Redundancy Management Interface (RMI) IPv4 configuration.

Verify the RMI IPv4 configuration.

```
Device# show running-config interface vlan management-vlan

Building configuration...

Current configuration : 109 bytes
!
interface Vlan90
ip address 9.10.90.147 255.255.255.0 secondary
ip address 9.10.90.41 255.255.255.0
end
```

To verify the interface configuration for a standby controller, use this command:

```
Device-stby# show running-config interface vlan 90

Building configuration...

Current configuration : 62 bytes
!
interface Vlan90
ip address 9.10.90.149 255.255.255.0
end
```

To verify the chassis redundancy management interface configuration for an active controller, use this command:

```
Device# show chassis rmi

Chassis/Stack Mac Address : 000c.2964.1eb6 - Local Mac Address
Mac persistency wait time: Indefinite
   H/W Current
Chassis# Role      Mac Address     Priority  Version  State  IP
RMI-IP
---------------------------------------------------------------------------------------
*1       Active    000c.2964.1eb6  1         V02      Ready  169.254.90.147
9.10.90.147
2        Standby   000c.2975.3aa6  1         V02      Ready  169.254.90.149
9.10.90.149
```

To verify the chassis redundancy management interface configuration for a standby controller, use this command:

```
Device-stby# show chassis rmi

Chassis/Stack Mac Address : 000c.2964.1eb6 - Local Mac Address
Mac persistency wait time: Indefinite
                                             H/W   Current
Chassis#   Role    Mac Address     Priority Version  State  IP
RMI-IP
---------------------------------------------------------------------------------------
1         Active   000c.2964.1eb6     1      V02     Ready  169.254.90.147
9.10.90.147
*2        Standby  000c.2975.3aa6     1      V02     Ready  169.254.90.149
9.10.90.149
```

To verify the ROMMON variables on an active controller, use this command:

```
Device# show romvar | include RMI

RMI_INTERFACE_NAME = Vlan90
RMI_CHASSIS_LOCAL_IP = 9.10.90.147
RMI_CHASSIS_REMOTE_IP = 9.10.90.149
```

To verify the ROMMON variables on a standby controller, use this command:

```
Device-stby# show romvar | include RMI

RMI_INTERFACE_NAME = Vlan90
RMI_CHASSIS_LOCAL_IP = 9.10.90.149
RMI_CHASSIS_REMOTE_IP = 9.10.90.147
```

To verify the switchover reason, use this command:

```
Device# show redundancy switchover history

Index  Previous  Current  Switchover             Switchover
       active    active   reason                 time
-----  --------  -------  ----------             ----------
   1       2        1     Active lost GW         17:02:29 UTC Mon Feb 3 2020
```

RMI IPv6 configuration verification

Provides commands to verify the RMI IPv6 configuration.

Verify the RMI IPv6 configuration.

To verify the chassis redundancy management interface configuration for both active and standby controllers, run this command:

```
Device# show chassis rmi

Chassis/Stack Mac Address : 00a3.8e23.a540 - Local Mac Address
Mac persistency wait time: Indefinite
Local Redundancy Port Type: Twisted Pair
                                             H/W   Current
Chassis#   Role     Mac Address    Priority Version State     IP
```

```
RMI-IP
------------------------------------------------------------------------------------------

  1        Standby  706d.1536.23c0    1      V02     Ready  169.254.254.17
2020:0:0:1::211
 *2        Active   00a3.8e23.a540    1      V02     Ready  169.254.254.18
2020:0:0:1::212
```

To verify the RMI related ROMMON variables for both active and standby controllers, run this command

```
Device# show romvar | i RMI

RMI_INTERFACE_NAME = Vlan52
RMI_CHASSIS_LOCAL_IPV6 = 2020:0:0:1::212
RMI_CHASSIS_REMOTE_IPV6 = 2020:0:0:1::211
```

Redundancy port interface configuration verification

Provides commands to verify redundancy port interface configuration.

Use these commands to verify redundancy port interface configuration for RMI and RP connections in both active and standby instances.

RMI link verification commands

To verify the RMI link re-establishment count and the time since the RMI link is Up in the active instance, run these command:

```
Device# show platform software rif-mgr chassis active R0 rmi-connection-details
RMI Connection Details
    RMI Link re-establish count : 2
    RMI Link Uptime             : 21 hours 8 minutes 43 seconds
    RMI Link Upsince            : 08/05/2021 13:46:01
```

To verify the RMI link re-establishment count and the time since the RMI link is Down in the active instance, run these command:

```
Device# show platform software rif-mgr chassis active R0 rmi-connection-details
RMI Connection Details
    RMI Link re-establish count : 1
    RMI Link Downtime           : 28 seconds
    RMI Link Downsince          : 07/16/2021 03:19:11
```

To verify the RMI link re-establishment count and the time since the RMI link is Up in the standby instance, run these command:

```
Device# show platform software rif-mgr chassis standby R0 rmi-connection-details
RMI Connection Details
    RMI Link re-establish count : 1
    RMI Link Uptime             : 1 hour 39 minute 9 seconds
    RMI Link Upsince            : 07/16/2021 01:31:41
```

To verify the RMI link re-establishment count and the time since the RMI link is Down in the standby instance, run these command:

```
Device# show platform software rif-mgr chassis standby R0 rmi-connection-details
RMI Connection Details
    RMI Link re-establish count : 1
```

```
    RMI Link Downtime           : 22 seconds
    RMI Link Downsince          : 07/16/2021 03:19:17
```

RP link verification commands

To verify the RP link re-establishment count and the time since the RP link is UP for days in the active instance, run the following command:

```
Device# show platform software rif-mgr chassis active R0 rp-connection-details
RP Connection Details
    RP Connection Uptime  : 12 days 17 hours 1 minute 39 seconds
    RP Connection Upsince : 07/03/2021 07:06:20
```

To verify the RP link re-establishment count and the time since the RP link is Down in the active instance, run the following command:

```
Device# show platform software rif-mgr chassis active R0 rp-connection-details
RP Connection Details
    RP Connection Downtime     : 4 seconds
    RP Connection Downsince    : 07/16/2021 03:33:04
```

To verify the RP link re-establishment count and the time since the RP link is UP in the standby instance, run the following command:

```
Device# show platform software rif-mgr chassis standby R0 rp-connection-details
RP Connection Details
    RP Connection Uptime  : 12 days 17 hours 2 minutes 1 second
    RP Connection Upsince : 07/03/2021 07:05:58
```

To verify the RP link re-establishment count and the time since the RP link is Down in the standby instance, run the following command:

```
Device# show platform software rif-mgr chassis standby R0 rp-connection-details
RP Connection Details
    RP Connection Downtime    : 22 seconds
    RP Connection Downsince   : 07/16/2021 03:19:17
```

RIF and stack manager internal statistics verification

To verify the RIF and stack manager internal statistics in the active instance, run the following command:

```
Device# show platform software rif-mgr chassis active R0 rif-stk-internal-stats
RIF Stack Manager internal stats

    Stack-mgr reported RP down            : False
    DAD link status reported to Stack-Mgr : True
```

To verify the RIF and stack manager internal statistics in the standby instance, run the following command:

```
Device# show platform software rif-mgr chassis standby R0 rif-stk-internal-stats
RIF Stack Manager internal stats

    Stack-mgr reported RP down            : False
    DAD link status reported to Stack-Mgr : True
```

LMP statistics verification

To verify the number of packets sent or received for each type in the active instance, run the following command:

```
Device# show platform software rif-mgr chassis active R0 lmp-statistics
LMP Statistics

Info Type Sent                    : 6
Solicit Info Type Sent            : 0
Unsolicit Info Type Sent          : 6
Reload Type Sent                  : 0
Recovery Type Sent                : 1
Gateway Info Type Sent            : 0
Enquiry Type Sent                 : 0
Solicit Enquiry Type Sent         : 0
Unsolicit Enquiry Type Sent       : 0

Info Type Received                : 5
Solicit Info Type Received        : 2
Unsolicit Info Type Received      : 3
Reload Type Received              : 0
Recovery Type Received            : 0
Gateway Info Type Received        : 4
Enquiry Type Received             : 0
Solicit Enquiry Type Received     : 0
Unsolicit Enquiry Type Received   : 0
```

To verify the number of packets sent or received for each type in the standby instance, run the following command:

```
Device# show platform software rif-mgr chassis standby R0 lmp-statistics
LMP Statistics

Info Type Sent                   : 6
Solicit Info Type Sent           : 0
Unsolicit Info Type Sent         : 6
Reload Type Sent                 : 0
Recovery Type Sent               : 0
Gateway Info Type Sent           : 4
Enquiry Type Sent                : 0
Solicit Enquiry Type Sent        : 0
Unsolicit Enquiry Type Sent      : 0

Info Type Received               : 5
Solicit Info Type Received       : 3
Unsolicit Info Type Received     : 2
Reload Type Received             : 0
Recovery Type Received           : 1
Gateway Info Type Received       : 0
Enquiry Type Received            : 0
Solicit Enquiry Type Received    : 0
Unsolicit Enquiry Type Received  : 0
```

Redundancy management interfaces

A redundancy management interface is a network interface that

- acts as a secondary link between active and standby wireless controllers,
- enables resource health information exchange (such as gateway reachability), and
- assists in the detection of dual-active controller conditions to maintain high availability.

The RMI might trigger a switchover based on the gateway status of the active controller.

Subdefinitions:

- **Active controller**: Uses the management IP as the primary address and RMI as the secondary IPv4 address on the management VLAN. RMI configuration is automatic.

  Analogy: Like a city's current mayor who's in charge, but always ready to hand over leadership if needed.

- **Standby controller**: Has the RMI IP as the primary address; upon switchover, roles and addresses are swapped.

  Analogy: Like a vice-mayor who takes the mayor's seat (with all responsibilities and keys) when the mayor is away.

- **RP (Redundancy Port)**: The main dedicated physical link for state and configuration synchronization between active and standby controllers; loss of both RP and RMI links results in high availability (HA) failures.

  (Analogy: Like the main road connecting two city offices, critical for daily business and coordination).

- **WMI (Wireless Management Interface)**: The main management interface for controller operations and communications; shares its subnet with the RMI and may serve as a source address for certain types of traffic such as AAA packets.

  Analogy: Like a city's public headquarters, used for both regular operations and official correspondence.

- RMI **(Redundancy Management Interface)**: A dedicated network interface that serves as a secondary communication path between controllers for exchanging resource health information, detecting dual-active conditions, and monitoring gateway reachability; shares the same subnet as the Wireless Management Interface (WMI).

  Analogy: Like an emergency side road connecting two city halls, used for urgent official communication.

- **HA (High Availability)**: A deployment setup where controllers operate as an active-standby pair to ensure uninterrupted wireless services; relies on RP and RMI links for failover and role management.

- **ARP table**: A database on a network device such as a switch that maintains mappings between IP addresses and MAC addresses; determines how to forward traffic within the local network.

  (Analogy: Like a city map book held by delivery drivers in neighboring towns, showing them which mayor is at which city hall, so the right deliveries go to the right address.)

- **GARP (Gratuitous ARP):** A type of ARP (Address Resolution Protocol) packet broadcast by a controller, typically after a switchover, to update the ARP tables in connected network switches with the correct IP-to-MAC address mappings.

  (Analogy: Like sending out an urgent memo to all nearby towns, telling them who the new mayor is so they update their city maps immediately after a leadership change.)

- **ARP cache timeout**: The duration for which an entry in the ARP table is considered valid before it must be refreshed; reducing this value helps the network recover quickly after role or address changes in the controllers.

  (Analogy: Like setting a regular schedule for when all delivery drivers check for updates to the city map books, ensuring they quickly recognize changes in city leadership or addresses.)

- **SGACL (Security Group Access Control List)**: A policy configuration that determines which types of network traffic (e.g., ICMP, ARP) are permitted between specific interfaces or devices, such as the RMI addresses of controllers.

  (Analogy: Like special city rules that decide which types of emergency vehicles are allowed to travel the emergency road between city halls.)

- **ICMP (Internet Control Message Protocol)**: A network protocol used for sending error messages and operational information between network devices; essential for controllers to monitor each other's status over the RMI.

  (Analogy: Like sending quick bike couriers to report on the status of the roads or to alert if there's trouble between cities.)

- **AAA (Authentication, Authorization, and Accounting**): A framework for controlling user access to network resources and tracking user activity; controllers may send AAA-related packets from either the WMI IP or the RMI IP, so the AAA server must recognize both as valid.

(Analogy: Like having a security checkpoint that logs who enters or exits city buildings, whether they come from Main Street (WMI) or the emergency road (RMI).)

- **SGT (Security Group Tag)**: An identification label assigned to network devices for policy enforcement with Cisco TrustSec; mapping is applied for both RMI and WMI addresses when device SGTs are used.

  Analogy: Like giving every city vehicle a badge, so officials can enforce special rules for each group whether they use Main Street or the emergency road.

- **Cisco TrustSec**: A security architecture that uses SGTs to enforce network segmentation and access control policies; not supported on the RMI interface.

  Analogy: Like a city's security zone system—effective on main city roads, but not supported on the emergency side road.

Limitations for RMI

- Cisco TrustSec is not supported on the RMI.
- From the Cisco IOS XE 17.14.1 release onwards, RP-only SSO is not supported for CW9800H1, CW9800H2, or CW9800M Wireless Controllers. These controllers support RP+RMI SSO deployment only. In contrast, Cisco Catalyst 9800 Wireless Controllers support both RP-only SSO and RP+RMI SSO.

Best practices for RMI

- Ensure that the SGACL is defined appropriately to allow ICMP and ARP traffic between the active and standby RMI addresses when device SGT is used, since the IP-SGT mapping applies to both the RMI and WMI addresses.
- Configure the AAA server to recognize both the WMI IP and the RMI IP as valid source addresses, because AAA packets from the controller may originate from either.
- RMI-based "RP" High Availability is mandatory in the Cisco Catalyst CW9800H1 Wireless Controller, Cisco Catalyst CW9800H2 Wireless Controller and Cisco Catalyst CW9800M Wireless Controller.

Important: gateway monitoring interval and detection time

- When gateway reachability is enabled, both the active and standby controllers check gateway status through the RMI interface.
- It takes approximately the configured gateway monitoring interval to detect when a controller has lost gateway reachability.
- The default gateway monitoring interval is eight seconds, so the minimum detection time is about eight seconds unless you configure a different value.

Recommendation: set ARP cache timeout to ensure fast HA recovery:

When both the RP and RMI links are down, the HA setup breaks and both controllers become active, causing an IP conflict in the network. The HA setup is restored once the RP link is up. Depending on the external switch, its ARP table may correctly update to the new active controller or remain stale if the switch ignores GARP packets, potentially prolonging the conflict.

- We recommend setting the ARP cache timeout to a low value. This practice enables faster recovery from multiple fault scenarios.

  You should choose a timeout value that does not negatively affect network traffic. For example, 30 minutes is a suitable interval.

Analogy

Think of redundancy management interfaces (RMI) as an emergency backup road that connects two cities (the active and standby controllers). The main highway (the redundancy port, RP) handles most of the day-to-day traffic and coordination. If the main highway is blocked or damaged, the emergency road (RMI) ensures that vital information—such as each city's health, status, and road conditions—can still be exchanged quickly.

Just as emergency vehicles and communication must be allowed to travel the backup road (analogous to allowing ICMP and ARP traffic via SGACL), the RMI helps both cities detect if both try to take charge at the same time (dual-active condition) and exchange important information about the area's main gateway (gateway status). If both the highway and the emergency road are inaccessible, each city might assume it's in charge, resulting in confusion (an IP conflict). The ARP table is like city maps in other towns (switches) that may not immediately recognize which city is currently in charge after both reconnect—setting these maps to update frequently (low ARP cache timeout) allows for faster recovery from major outages.

Controller operations with Redundant Management Interface

Explains how active and standby controllers assign and manage IP addresses using the redundant management interface (RMI) in high-availability setups.

A controller operation with redundant management interface is a high-availability configuration that

- assigns the management IP address as the primary address on the active controller
- uses the secondary IPv4 address on the management VLAN as the RMI IP address for the active controller, and
- configures the RMI IP address as the primary IP address on the standby controller.

IP address assignment in controller operations

The active controller assigns IP addresses as follows:

- The primary address is the management IP address.
- The secondary IPv4 address on the management VLAN is the RMI (Redundant Management Interface) IP address for the active controller.

The standby controller manages IP addresses in a high-availability setup as follows:

- It does not have the wireless management IP configured.
- The RMI IP address is configured as the primary IP address on the standby controller.
- When the standby controller becomes active, the management IP address becomes the primary IP address, and the RMI IP address becomes the secondary IP address.
- If the interface on the active controller is administratively down, the same state is reflected on the standby controller.

> **Note**
>
> Do not configure the secondary IPv4 address explicitly. RMI automatically configures a single secondary IPv4 address under the RMI.

Dual stack configurations on management vlans with RMI

Explains the capability of the management VLAN to be configured with both IPv4 and IPv6 addresses for RMI and WMI.

A dual stack configuration is a network interface setup that

- allows both IPv4 and IPv6 addresses to be configured on the wireless management interface,
- permits monitoring only of the gateway that matches the configured RMI address family (IPv4 or IPv6), and
- restricts the visibility of the alternate family's management address on the standby controller.

Expanded explanation

- Dual stack refers to the fact that the wireless management interface can be configured with IPv4 and IPv6 addresses. If an RMI IPv4 address is configured along with an IPv4 management IP address, you can additionally configure an IPv6 management address on the wireless management interface. This IPv6 management IP address will not be visible on the standby controller.
- If an RMI IPv6 address is configured along with an IPv6 management IP address, you can additionally configure an IPv4 management address on the wireless management interface. This IPv4 management IP address will not be visible on the standby controller.
- Therefore, you can monitor only the IPv6 gateway when the RMI IPv6 address is configured, or only the IPv4 gateway when the RMI IPv4 address is configured.

> **Note**
>
> The RMI feature supports the RMI IPv4 or IPv6 addresses.

RMI-based high-availability pairings

Describes a controller deployment configuration that uses Remote Machine Interface to synchronize two controllers for redundancy and failover.

A RMI-based high-availability pair is a controller deployment configuration that

- uses Remote Machine Interface (RMI) to synchronize two controllers,
- provides redundancy by designating active and standby roles, and
- ensures failover and persistent state during controller reloads or outages.

RMI-based high-availability pairing scenarios and device support

You should consider RMI-based high-availability pairs in the following scenarios:

- Fresh installation: Configure high availability during the initial setup of controllers.
- Already paired controllers: Adjust or reconfigure pairing for controllers that are already part of a high-availability pair.
- Upgrade scenario: Maintain or update the pair relationship during software or hardware upgrades.
- Downgrade scenario: Ensure pairing remains stable and functional during downgrades.

Dynamic high-availability (HA) pairing requires both the active and standby controllers to reload. In practice, on the Cisco Catalyst 9800-L, 9800-40, and 9800-80 Wireless Controllers, dynamic pairing occurs when one controller reloads and becomes the standby member of the pair.

> **Note**
>
> Unique chassis numbers must be configured for each controller before forming an HA pair, as these numbers identify the controllers within the pair.

HA pairing without previous configuration

Describes a deployment scenario for wireless controllers that enables initial HA setup on devices without existing ROMMON variables.

A high-availability (HA) pairing without previous configuration is a deployment scenario for wireless controllers that

- initiates the HA setup on devices without existing ROMMON variables for RP (Route Processor) IP addresses
- allows selection between the soon-to-be-deprecated privileged EXEC mode RP-based commands and the newer RMI IP-based mechanisms, and
- derives RP IPs from RMI IPs after forming the HA pair, with restrictions on method transitions.

Command usage and method selection

When HA pairing is performed for the first time (without previous setup), devices do not have ROMMON variables for RP IP addresses.

After RMI-based HA pairing on a brand-new system:

- RP IPs are derived from RMI IPs and used in HA pairing.
- Privileged EXEC mode RP-based CLIs method of clearing and forming an HA pair is not allowed.
- To view the ROMMON variables, use the **show romvars** command.

> **Caution**
>
> Privileged EXEC mode RP-based commands are deprecated and will be blocked after choosing the RMI-based HA pairing.

Method selection considerations:

- You can still choose from the existing privileged EXEC mode RP-based commands or the RMI IP-based mechanisms. However, the privileged EXEC mode RP-based commands are deprecated.
- If you use Cisco Catalyst Center, you can choose the privileged EXEC mode RP-based CLI mechanism till the Cisco Catalyst Center migrates to support the RMI.
- If you choose privileged EXEC RP-based CLI mechanism, the RP IPs are configured the same way as in the 16.12 release.

Use the RMI IP-based mechanism for fresh installations, even though both RP-based and RMI methods may initially be available.

Software version requirements:

- The RMI migration is supported from Cisco Catalyst Center, 2.3.3.x release version.

- RMI-based High Availability requires Cisco IOS XE release version 17.3 or above.

Cisco Catalyst Center interoperability:

- If you use Cisco Catalyst Center, you can choose the privileged EXEC mode RP-based CLI mechanism till the Cisco Catalyst Center migrates to support the RMI.
- The RMI migration is supported from Cisco Catalyst Center, 2.3.3.x release version.

Negative cases where RMI migration fails include:

- Devices are not reachable.
- Non-Cisco Catalyst 9800 Series Wireless Controllers are in use.
- Controller is running Cisco IOS XE 17.3 or below
- High Availability is not configured.
- High Availability RMI is already configured.
- Attempting upgrade to an already failed High Availability paired controller.

> ⚠ **Caution**
>
> The controller GUI prohibits applying RMI migration configuration to High Availability failed devices.

Security system installation analogy

Establishing HA pairing without previous configuration is like setting up a new security system in a building that has never had one before.

At first, you have the choice between using an old key-based system (privileged EXEC mode RP-based commands), which will soon be phased out, or installing a new state-of-the-art digital access system (RMI IP-based mechanism).

Once you install the modern digital system and program it to generate security codes based on the latest technology (RP IPs derived from RMI IPs), the use of physical keys becomes unavailable.

If you later decide to upgrade, you can only add new features to the digital system; you cannot go back to using the old keys because the doors no longer have compatible locks.

This analogy illustrates how choosing the RMI-based approach in HA pairing establishes a new baseline that does not allow reverting to the older method.

Paired controllers

Describes a high availability infrastructure configuration that links two controllers for redundancy and failover operations.

A paired controller is a high availability (HA) infrastructure configuration that

- links two controllers to operate jointly for redundancy and failover,
- allows seamless migration from traditional EXEC mode RP-based commands to RMI-based HA pairing, and
- ensures controller identity and connectivity are maintained even when core pairing mechanisms are updated or reloaded.

Expanded explanation

When controllers are already in an HA pair, they continue to use existing EXEC mode RP-based commands unless Remote Management Interface (RMI) is enabled. Enabling RMI migrates the system to use RMI-derived HA pairing, overwriting any existing RP IPs with those derived from the RMI configuration. The HA pair remains stable immediately after this change, but the controllers only adopt the new IPs following their next reload.

RMI requires controllers to be reloaded for the changes to take effect. Once both controllers restart, they reestablish the HA pair using the new RMI-derived RP IPs. After pairing through RMI, EXEC mode RP-based commands are blocked, preventing configuration conflicts.

Examples

- Two controllers configured as a high availability pair, where enabling RMI changes the way their active-standby relationship is managed and what IPs are used for internal communication.
- An active and standby controller pair that continues functioning during migration from legacy RP-IP pairing to RMI, without disruption until reload.

Counter-examples

- Two standalone controllers operating independently without HA pairing cannot be considered paired controllers.
- A controller pair where RMI is never configured and all management remains through EXEC mode RP-based commands does not benefit from RMI-derived features.

Analogy

Imagine a paired controller setup as two co-pilots flying an airplane together (the airplane represents your network environment). Traditionally, they use walkie-talkies (EXEC mode RP-based commands) to coordinate their flying activities. If you upgrade their communication system to headsets (RMI-based pairing), the co-pilots continue flying the plane using walkie-talkies until they both put on the new headsets (after a "reload" or restart).

From that point onward, all their coordination happens via the more reliable and advanced headsets. The co-pilots' ability to work together—their partnership—remains unbroken throughout; it is only how they communicate and identify each other's messages that changes, and only becomes effective after both are using the new headsets.

Upgrade from Cisco IOS XE 16.1.x to a later release

Provides upgrade and downgrade options when migrating a system from Cisco IOS XE 16.1.x to a later release.

When upgrading a system, you have these options:

- Migrate while retaining the existing RP IP configuration: In this scenario, the current RP IP configuration remains unchanged, and future modifications will utilize EXEC mode RP-based commands.
- Migrate after clearing the HA configuration: Here, you have the choice to use either the traditional EXEC mode RP-based commands or adopt the new RMI-based RP configuration. If the previous configuration is preserved, RMI will update the RP IPs with those derived from the RMI IPs.

Downgrade scenario

Important

The downgrade scenario given below is not applicable for Cisco IOS XE 17.1.x.

In a downgrade scenario, only EXEC mode RP-based commands are available. The downgrade process may follow one of these paths:

- If the upgraded system used the RMI-based RP configuration.
- If the upgraded system continued to use the EXEC mode RP-based commands.

In the above cases, the downgraded system uses the EXEC mode RP-based commands to modify the configuration. However, the downgraded system will continue to use the new derived RP IPs.

In both of these cases, the system will revert to EXEC mode RP-based commands for configuration alterations, yet will still utilize the newly derived RP IPs.

> **Note**
>
> When you downgrade the Cisco Catalyst 9800 Series Wireless Controller to any version below Cisco IOS XE 17.1 and if the mDNS gateway is enabled on the WLAN/RLAN/GLAN interfaces, the mdns-sd-interface gateway goes down after the downgrade.
>
> To enable the mDNS gateway on the WLAN/RLAN/GLAN interfaces in Cisco IOS XE 16.12 and earlier versions, use these commands:
>
> **wlan test 1 test**
>
> **mdns-sd gateway**
>
> To enable the mDNS gateway on the WLAN/RLAN/GLAN interfaces from version Cisco IOS XE 17.1 onwards, use these command:
>
> **mdns-sd-interface gateway**

Configure redundancy management interface (GUI)

Configure redundancy management interface using GUI to enable failover capabilities on Cisco Catalyst 9800 Series Wireless Controllers.

Enable redundancy management for Cisco Catalyst 9800 Series Wireless Controllers using the graphical user interface (GUI).

Use this task to configure the redundancy management interface (RMI) and set up either RMI+RP or RP redundancy pairing on Cisco Catalyst 9800 Series Wireless Controllers. Configuring redundancy improves system availability and failover capabilities.

Before you begin

Ensure that Wireless Management Interface (WMI) is available before configuring RMI + RP using the GUI.

Follow these steps to configure redundancy management interface using GUI:

Procedure

1. In the **Administration > Device > Redundancy** window, perform the following:
   a. Set the **Redundancy Configuration** toggle button to **Enabled** to activate redundancy configuration.
   b. In the **Redundancy Pairing Type** field, select **RMI+RP** to perform RMI+RP redundancy pairing as follows:
      - In the **RMI IP for Chassis 1** field, enter the RMI IP address for chassis 1.
      - In the **RMI IP for Chassis 2** field, enter the RMI IP address for chassis 2.
      - From the **HA Interface** drop-down list, choose one of the HA interfaces.

You can select the HA interface only for Cisco Catalyst 9800 Series Wireless Controllers.

- Set the **Management Gateway Failover** toggle button to **Enabled** to activate management gateway failover.
- In the **Gateway Failure Interval** field, enter an appropriate value. The valid range is 6–12 seconds (default 8 seconds).

c. In the **Redundancy Pairing Type** field, select **RP** to perform RP redundancy pairing as follows:

- In the **Local IP** field, enter an IP address for the local chassis.
- In the **Netmask** field, enter the subnet mask assigned to all wireless clients.
- From the **HA Interface** drop-down list, choose one of the HA interfaces.

  You can select the HA interface only for Cisco Catalyst 9800 Series Wireless Controllers.
- In the **Remote IP** field, enter an IP address for the remote chassis.

d. In the **Keep Alive Timer** field, enter an appropriate timer value (1–10 × 100 milliseconds).

e. In the **Keep Alive Retries** field, enter an appropriate retry value (3–10 seconds).

f. In the **Active Chassis Priority** field, enter a value.

2. Click **Apply** and reload controllers.

The redundancy management interface is configured, and redundancy pairing is established based on your chosen method. The controller is now set up for improved high availability and failover.

Configure redundancy management interface (CLI)

Configure redundancy management interface using commands.

Configure a Redundancy Management Interface (RMI) on Cisco Catalyst 9800 controllers using CLI commands to support high availability (HA) between two devices.

Use this task when you want to set up high availability and redundancy between two Catalyst 9800 series controllers. The RMI coordinates HA communication and failover, ensuring service continuity in case of device failure.

Before you begin

- Ensure both controllers are cabled and powered on.
- Verify you have administrator access to both devices via CLI.
- Gather the following information:
  - Chassis number (1 or 2 for each controller)
  - Desired chassis priority for HA (if overriding default)
  - A dedicated GigabitEthernet interface for HA communication (required for 9800-CL controllers)
  - Management VLAN and corresponding IP addresses for each chassis

Procedure

---

1. Optional: Configure the priority of the specified device.

Example

```
Device# chassis chassis-num priority chassis-priority
```

Example

```
Device# chassis 1 priority 1
```

From Cisco IOS XE Gibraltar 16.12.x onwards, device reload is not required for the chassis priority to become effective.

- *chassis-num*–Enter the chassis number. The range is from 1 to 2.
- *chassis-priority*–Enter the chassis priority. The range is from 1 to 2. The default value is 1.

When both the devices boot up at the same time, the device with higher priority becomes active, and the other one becomes standby. If both the devices are configured with the same priority value, the one with the smaller MAC address acts as active and its peer acts as standby.

2. Create an HA interface for your controller.

Example

```
Device# chassis redundancy ha-interface GigabitEthernet interface-number
```

Example

```
Device# chassis redundancy ha-interface
GigabitEthernet 3
```

- *interface-number*: GigabitEthernet interface number. The range is from 1 to 32.

This step is applicable only for Cisco Catalyst 9800-CL Series Wireless Controllers. The chosen interface is used as the dedicated interface for HA communication between the 2 controllers.

3. Enter global configuration mode.

Example

```
Device# configure terminal
```

4. Configure Redundancy Management Interface.

Example

```
Device(config)# redun-management interface vlan vlan-interface-number chassis
 chassis-number address ip-address chassis chassis-number address ip-address
```

Example

```
Device(config)# redun-management interface
Vlan 200 chassis 1 address 9.10.90.147
chassis 2 address 9.10.90.149
```

- *vlan-interface-number*: VLAN interface number. The valid range is from 1 to 4094.

Here, the *vlan-interface-number* is the same VLAN as the Management VLAN. That is, both must be on the same subnet.

- *chassis-number*: Chassis number. The valid range is from 1 to 2.
- *ip-address*: Redundancy Management Interface IP address.

Each controller must have a unique chassis number for RMI to form the HA pair. The chassis number can be observed as SWITCH_NUMBER in the output of **show romvar** command. Modification of SWITCH_NUMBER is currently not available through the web UI.

To disable the HA pair, use the **no redun-management interface vlan chassis** command.

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

6. Save the configuration.

   Example

   ```
   Device# write memory
   ```

7. Reload the controllers.

   Example

   ```
   Device# reload
   ```

   When the RMI configuration is done, you must reload the controllers for the configuration to take effect.

   For Cisco Catalyst 9800-CL Wireless Controller VM, both the active and standby controllers reload automatically. In the case of hardware platforms, you should reload the active controller manually, as only standby the controller reloads automatically.

The redundancy management interface is configured. After reload, an HA pair is established between the two controllers, enabling redundancy and failover support.

Part XIII

Quality of Service

Topics:

- Quality of Service
- Wireless Auto-QoS
- Native Profiling
- Air Time Fairness
- IPv6 Non-AVC QoS Support
- QoS Basic Service Set Load

Configuring QoS

174 Quality of Service

Topics:

- Wireless Quality of Service
- Bidirectional rate limiting
- Per-client bidirectional rate limiting
- How to configure wireless QoS

This chapter explains how to configure and apply wireless QoS on Cisco Catalyst 9800 controllers, including metal policies, DSCP/User Priority mappings, bidirectional and per-client rate limiting, GUI/CLI procedures, and related prerequisites and restrictions.

Wireless Quality of Service

Introduces the fundamentals of wireless QoS, including target selection for SSID and clients, supported QoS features, mobility considerations, metal policy options, prerequisites, and operational restrictions, with details on client connectivity and TCP packet handling on CWA SSID.

A wireless Quality of Service policy is a network management policy that

- prioritizes specific types of wireless traffic by giving preferential treatment
- applies different rules to SSID and client targets in both upstream and downstream directions, and
- supports traffic marking, rate limiting (policing), mobility features, and compatibility with advanced controller functions.

- Upstream traffic: The flow of data from a wireless source to a wired target.
- Downstream traffic: The flow of data from a wired source to a wireless target.
- Target: The entity (SSID or client) where the QoS policy is enforced.

Additional reference information

- Without QoS, network devices transmit packets with best-effort service, offering no guarantees for reliability, delay bounds, or throughput. Wireless QoS policies enhance the network by ensuring that traffic with higher priority receives preferential treatment. This policy improves overall performance for critical applications.
- Applying a wireless QoS policy to prioritize voice traffic on an SSID ensures that calls suffer less latency or jitter compared to general web browsing traffic.
- Rate limiting is used to prevent a single client from consuming excessive bandwidth and to maintain fair usage for all wireless clients.
- A network with no configured QoS policies treats all wireless traffic equally, which can lead to poor performance for delay-sensitive applications.
- QoS policies designed for wired networks do not automatically apply to wireless environments, as wireless traffic has unique constraints.

Wireless QoS targets

This section describes the various wireless QoS targets available on a device.

Service set identifiers policies

SSID policies.

An Service Set Identifiers (SSID) policy is a wireless network configuration policy that

- controls the application of QoS settings to a wireless SSID in both ingress and egress directions
- applies per AP and per SSID, and
- allows configuration of policing and marking actions on SSID traffic.

If an SSID policy is not configured, no QoS policy is applied to the SSID.

Client policies

Wireless QoS mobility enables configuration of QoS policies so that the network provides the same service anywhere in the network.

Client policies are applicable in the ingress and egress direction. You can configure policing and marking policies on clients. AAA override is also supported.

Supported QoS features on wireless targets

The precious metal policies are system-defined policies that are available on the controller.

Wireless controllers support various QoS features to manage traffic and ensure optimal performance for both SSIDs and client devices. These tables show the supported features, applicable directions, and configuration modes for wireless targets.

Table 151: QoS features available on wireless targets

This table describes the various features available on wireless targets.

Target	Features	Direction where policies Are applicable
SSID	• Set • Police • Drop	Upstream and downstream
Client	• Set • Police • Drop	Upstream and downstream

Table 152: QoS policy actions

This table describes the various features available on wireless targets.

Policy action types	Wireless target support	
	Local mode	FlexConnect Mode
Police	Supported	Supported with restrictions **Note** Per-SSID policing is not supported for FlexConnect locally switched traffic.
Set	Supported	Supported

Table 153: QoS policy set actions

This table describes the various features available on wireless targets.

Set action types	Supported	
	Local mode	FlexConnect mode
set dscp	Supported	Supported

Set action types	Supported	
	Local mode	FlexConnect mode
set qos-group	Supported	Not Supported
set wlan user-priority (downstream only)	Supported (BSSID only)	Supported (BSSID only)

Wireless QoS mobility

Wireless Quality of Service (QoS) mobility enables you to configure QoS policies so that the network provides the same service anywhere in the network.

A wireless QoS mobility feature is a network mobility mechanism that

- enables configuration of QoS policies to provide consistent service for wireless clients
- supports seamless roaming between different access points and network devices and
- maintains the same service levels no matter where the client connects in the network.

Wireless client roaming can occur in two forms:

- Intra-device roaming: Roaming across access points managed by the same device.
- Inter-device roaming: Roaming across access points managed by different devices.

Additional reference information

Note

- In a foreign wireless LAN controller (WLC), client statistics are not displayed.
- Make sure that all client policies are available on every device in the mobility group.
- Apply the same Service Set Identifier (SSID) policy to every device in the mobility group to ensure clients receive consistent treatment.

Precious metal policies for wireless QoS

Precious metal policies for wireless QoS.

Precious metal policies for wireless QoS are system-defined QoS policies

- assign different service levels to wireless network traffic based on pre-set categories,
- remain unmodifiable, cannot be removed by administrators, and
- affect packet attributes, such as 802.11e (WMM) and DSCP fields, when policies are applied.

- Platinum: Used for VoIP clients, assigns the highest priority.
- Gold: Used for video clients, assigns high priority, lower than platinum.
- Silver: Used for best-effort traffic, assigns standard priority.

- Bronze: Used for Non-Real-Time (NRT) traffic, assigns the lowest priority.

Additional reference information

Preconfigured precious metal policies are available on wireless controllers. Administrators cannot modify or delete these policies. AAA mechanisms may push client metal policies. These policies determine packet scheduling and marking on the network.

In FlexConnect local switching mode, APs do not enforce QoS metal policy ceiling limits for upstream traffic. Limit enforcement occurs at the controller exit point. The AP does not change DSCP values.

See the Metal policy format section for more information about metal policy formats.

See the Metal policy map section for more information about metal policies.

> **Note**
>
> In FlexConnect local switching mode, default precious metal QoS policies are not pushed to access points. To apply QoS in FlexConnect local switching deployments, configure and apply custom QoS profiles.

Precious metal policies for wireless QoS

- Assigning the platinum policy to VoIP devices ensures minimal latency for voice communication.
- Assigning the gold policy to video streaming clients optimizes traffic for media applications.
- The silver policy applied to laptops browsing the internet handles general best-effort data.
- The bronze policy used for devices performing background updates minimizes their network priority.
- Custom user-defined QoS policies that administrators can modify or remove are not considered precious metal policies.
- Policies that do not map traffic types to platinum, gold, silver, or bronze levels are not included.

Prerequisites for wireless QoS

Prerequisites for wireless QoS.

Before you configure wireless QoS, make sure you understand key concepts and network factors to ensure effective deployment.

The required prerequisites include:

- Understand wireless concepts and network topologies.
- Understand QoS implementation.
- Modular QoS CLI (MQC). To learn more about Modular QoS CLI, see the MQC guide.
- Understand the types of applications used and the traffic patterns on your network.
- Understand your network's bandwidth requirements and speed.

Restrictions for QoS on wireless targets

Restrictions for QoS on wireless targets.

QoS policy application on wireless targets (such as SSIDs, BSSIDs, and wireless clients) has these key restrictions and considerations.

General Restrictions

- QoS policy application on wireless targets (such as Service Set Identifier (SSID), BSSID, and wireless clients) includes these key restrictions and considerations:

Hierarchical (parent policy and child policy) QoS is not supported.

- QoS policy maps and class maps that are created or customized through the CLI may include options or policy variations that are not supported for modification through the GUI. To modify QoS policy maps and class maps that were created or customized through the CLI, use the CLI.
- Configure SSID and client targets only with marking and policing policies.
- You can assign only one policy per target for each direction.
- Although class maps in a policy map can have different types of filters, only one marking action (set dscp) is supported.
- Only one set action per class is supported.
- You cannot use access group matching.
- Access points in flex mode do not support Access Control List (ACL) matching for local switching traffic.
- SIP Call Admission Control (CAC) is not supported on central switching mode.
- From Cisco IOS XE Amsterdam 17.3.1 and later, SIP Call Admission Control (CAC) is not supported.
- Do not apply QoS on the WMI interface because it may reboot the controller.
- AP QoS statistics for each radio stop updating after 32,768 minutes (546 hours). At that point, the offered rate shows zero and the minute counter stops increasing.

  The system calculates rates within a 32,768-minute window (546 hours). After 32,768 minutes, the data rate calculation is zero.

  To reset the statistics, run this command to clear the QoS statistics for the target policy map and SSID:

```
show policy-map interface wireless ssid profile-name <wlan_profile_name>
radio type <radio_type> ap name <AP_name> input clear
```

AP Side Restrictions

- In Cisco Embedded Wireless Controller, FlexConnect local switching, and Software-Defined Access (SDA) deployments, the AP enforces QoS policies and rate-limiting actions at the per-flow (5-tuple) level, not per client.
- For FlexConnect local switching with local authentication and AAA override enabled (using an external AAA server), you can use only air space VLAN and ACL as AAA overrides. QoS and other overrides are not available.
- Rate limiting per SSID is not supported in FlexConnect local switching mode. For rate limiting to work as expected, FlexConnect central switching mode should be used.
- In FlexConnect local switching mode, per-SSID QoS policies that use a policing action (**police cir**) are not supported. Use FlexConnect central switching when per-SSID traffic policing is required.

Control Plane Rate Limiting and Policing

You do not need to configure control plane rate limiting or policing. Built-in mechanisms, such as policers, protect the CPU from control plane traffic. Migrations from AireOS to IOS-XE handle this automatically.

Restriction: Client connectivity and TCP packet drops on SSIDs with AAA rate limiting

Outlines restrictions for troubleshooting client connectivity and TCP packet drops on local-mode APs with CWA SSIDs, especially when rate limits and AAA overrides are configured.

Do not configure rate limits or AAA overrides on Cisco ISE for any SSID where AAA rate limiting is applied. These settings can cause the APs in local mode to drop TCP packets. In such a scenario, use Modular QoS CLI (MCQ)-based method only for configuring Quality of Service (QoS) and rate limiting.

Behavior explanation

When a client connects to an SSID with AAA rate limiting configured on APs in local mode, the AP forwards ICMP, DHCP, and DNS packets to the controller but drops TCP packets. This issue is caused when rate limits are configured on Cisco ISE and AAA overrides are in place. You can verify that TCP packets are dropped using the **show datapath drop trace** comand.

Workaround

Review the configuration of rate limits and consider removing them or adjusting the settings to avoid dropping TCP packets. Use Modular QoS CLI (MCQ)-based method only for configuring Quality of Service (QoS) and rate limiting. For more information about the MCQ-based method, see Configure QoS Rate Limit on Catalyst 9800 Wireless Controllers.

Verify client connectivity

- To observe the AP configuration, use this command.

```
show ap dot11 5ghz network
```

- To verify the AP configuration, use this command.

```
show ap dot11 5ghz network
```

- To check the client summary, use this command.

```
show wireless client summary
```

- To verify client details and troubleshoot connectivity issues, use this command:

```
show wireless client mac-address MAC-address detail
```

Bidirectional rate limiting

Explains bi-directional rate limiting concepts and outlines prerequisites, application of metal policies, configuration steps for SSID and client, and traffic-classification-based rate limiting, including procedures for applying policy maps to profiles.

A bidirectional rate limit is a wireless network traffic management feature that

- establishes configurable rate limits for both upstream and downstream traffic directions
- enables administrators to set individual limits per direction directly on the WLAN, overriding QoS profile values, and global controller configurations, and
- supports prioritization of client groups by assigning them to specific QoS profiles.

QoS profiles

There are four distinct QoS profiles to configure rate limits:

- Gold

- Platinum
- Silver
- Bronze

Additional reference information

- Apply rate limiting directly to a WLAN. This action overrides both global QoS settings and QoS profiles for the controller and clients.
- Bidirectional rate limits apply to all clients associated with a given SSID. Every client connected to the same SSID has identical rate restrictions.
- Set throughput limits to control wireless client performance for both traffic directions. Assign service priority to specific client sets for prioritization.

Configuration guidance

- Select a QoS profile and configure the rate limiting parameters for upstream and downstream directions. Setting a parameter to `0` disables rate limiting for that direction.
- Assign a QoS profile to each WLAN to determine the rate limits for all connected clients.

Scenario considerations

- Configure bidirectional rate limits on both the Anchor and Foreign controllers in mobility Anchor–Foreign controller setups. Use identical configuration across both controllers to prevent feature failure.
- The feature is supported in guest anchor scenarios, including IRCM guest deployments where AireOS devices function as guest anchor or guest foreign.
- Cisco Catalyst 9800 Series Wireless Controller uses a policing option to enforce bidirectional rate limits.

Bidirectional rate limiting

If a guest SSID uses the Bronze QoS profile and sets both upstream and downstream limits to specific values, all guests on that SSID have consistent rate restrictions, regardless of any global QoS or profile settings.

Apply metal policy with bidirectional rate limiting

Summary

Applying metal policies with bidirectional rate limiting involves configuring network policies to manage bandwidth usage and enforce traffic control on both upload and download directions.

Workflow

To apply metal policy with bidirectional rate limiting, perform these tasks:

1. Configure Metal Policy on SSID.
2. Configure Metal Policy on Client.
3. Configure bidirectional rate limiting for all traffic.
4. Configure bidirectional rate limiting Based on traffic classification.
5. Apply bidirectional rate limiting policy map to policy profile.
6. Apply Metal Policy with bidirectional rate limiting.

Result

The network enforces metal policies with bidirectional rate limiting, providing effective traffic management and resource fairness across SSIDs and client devices.

Requirements for bidirectional rate limiting configuration

Prerequisites for bidirectional rate limiting.

Ensure you meet the essential requirements before configuring bidirectional rate limiting.

To configure bidirectional rate limiting, ensure these prerequisites are met:

- Apply the client metal policy through AAA override.
- Specify the metal policy on the Identity Services Engine (ISE) server.
- Enable AAA override on the policy profile.

Configure the metal policy on the SSID

Configure the metal policy on the SSID.

Apply a metal service policy to a wireless SSID using a WLAN policy profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a WLAN policy profile and enter wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy policy-profile-name
   ```

3. Add a user-defined description to the new wireless policy.

   Example

   ```
   Device(config-wireless-policy)# description description
   ```

4. Set the platinum policy for input.

   Example

   ```
   Device(config-wireless-policy)# service-policy input input-policy
   ```

5. Set the platinum policy for output.

Example

```
Device(config-wireless-policy)# service-policy output output-policy
```

The system applies the configured policy profile to the SSID. It enforces platinum-level input and output service policies.

Configure a metal policy on your client device

Configure a metal policy on your client device.

Set up a wireless metal policy profile that uses AAA override on your client device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile and enter wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy policy-profile-name
   ```

3. Add a user-defined description to the new wireless policy.

   Example

   ```
   Device(config-wireless-policy)# description description
   ```

4. Enable AAA override on your WLAN.

   Example

   ```
   Device(config-wireless-policy)# aaa-override
   ```

   Note

   After you enable AAA override and the ISE server starts sending policy, the client policy defined in the service-policy client does not take effect.

With AAA override enabled, your wireless policy profile allows external RADIUS authentication and policy control.

Configure bidirectional rate limiting for all traffic

Configure bidirectional rate limiting for all traffic.

Enforce bidirectional rate limiting on all traffic by using policy maps.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a named object to apply policies to traffic classes.

   Example

   ```
   Device(config)# policy-map policy-map
   ```

   Policy-map names can contain alphabetic, hyphen, or underscore characters. The names are case sensitive. Names can be up to 40 characters long.

3. Associate a class map with the policy map and enter policy-map class configuration mode.

   Example

   ```
   Device(config-pmap)# class class-map-name
   ```

4. Configure traffic policing for your traffic class.

   Example

   ```
   Device(config-pmap-c)# police rate
   ```

   Valid values are 8,000 to 200,000,000 bps.

The bidirectional rate limiting policy limits all traffic that matches the class map.

Configure Bi-Directional Rate Limiting Based on Traffic Classification

Configure bi-directional rate limiting based on traffic classification.

Procedure

1. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. **policy-map** *policy-map*

   Example

   ```
   Device(config)# policy-map policy-sample2
   ```

   Creates a named object representing a set of policies that are to be applied to a set of traffic classes. Policy-map names can contain alphabetic, hyphen, or underscore characters, are case sensitive, and can be up to 40 characters.

3. **class** *class-map-name*

   Example

   ```
   Device(config-pmap)# class class-sample-youtube
   ```

   Associates a class map with the policy map, and enters policy-map class configuration mode.

4. **police** *rate*

   Example

   ```
   Device(config-pmap-c)# police 1000000
   ```

   Configures traffic policing (average rate, in bits per second). Valid values are 8000 to 200000000.

5. **conform-action drop**

   Example

   ```
   Device(config-pmap-c-police)# conform-action drop
   ```

   Specifies the drop action to take on packets that conform to the rate limit.

6. **exceed-action drop**

   Example

   ```
   Device(config-pmap-c-police)# exceed-action drop
   ```

   Specifies the drop action to take on packets that exceeds the rate limit.

7. **exit**

   Example

   ```
   Device(config-pmap-c-police)# exit
   ```

   Exits the policy-map class configuration mode.

8. **set dscp default**

   Example

   ```
   Device(config-pmap-c)# set dscp default
   ```

   Sets the DSCP value to default.

9. **police** *rate*

   Example

   ```
   Device(config-pmap-c)# police 500000
   ```

   Configures traffic policing (average rate, in bits per second). Valid values are 8000 to 200000000.

10. **exit**

    Example

    ```
    Device(config-pmap-c)# exit
    ```

    Exits the policy-map class configuration mode.

11. **exit**

    Example

    ```
    Device(config-pmap)# exit
    ```

    Exits the policy-map configuration mode.

12. **class-map match-any** *class-map-name*

    Example

    ```
    Device(config)# class-map match-any class-sample-youtube
    ```

    Selects a class map.

13. **match protocol** *protocol*

    Example

    ```
    Device(config-cmap)# match protocol youtube
    ```

    Configures the match criteria for a class map on the basis of the specified protocol.

Apply bidirectional rate limiting policy map to policy profile

Apply bidirectional rate limiting policy map to policy profile.

Apply bidirectional rate limiting to your wireless network by attaching a policy map to a policy profile.

Procedure

1. Enter global configuration mode to begin setup.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile to enter wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy policy-profile-name
   ```

3. Add a user-defined description to your new wireless policy.

   Example

   ```
   Device(config-wireless-policy)# description description
   ```

4. Set the input client service policy to platinum.

   Example

   ```
   Device(config-wireless-policy)# service-policy client input platinum-up
   ```

5. Set the output client service policy to platinum.

   Example

   ```
   Device(config-wireless-policy)# service-policy client output output-policy
   ```

6. Set the input service policy to platinum.

   Example

   ```
   Device(config-wireless-policy)# service-policy input input-policy
   ```

7. Set the output service policy to platinum.

   Example

   ```
   Device(config-wireless-policy)# service-policy output platinum
   ```

Your wireless policy profile applies bidirectional rate limiting based on the service policies you set.

Apply metal policy with bidirectional rate limiting

Apply metal policy with bidirectional rate limiting.

Use a metal policy to enforce bidirectional bandwidth limits on wireless traffic.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy policy-profile-name
   ```

3. Add a description for the new wireless policy.

   Example

   ```
   Device(config-wireless-policy)# description description
   ```

4. Assign 'platinum' as the input client service policy.

   Example

   ```
   Device(config-wireless-policy)# service-policy client input input-policy
   ```

5. Assign 'platinum' as the output client service policy.

   Example

   ```
   Device(config-wireless-policy)# service-policy client output output-policy
   ```

6. Assign 'platinum' as the input service policy.

   Example

   ```
   Device(config-wireless-policy)# service-policy input input-policy
   ```

7. Assign 'platinum' as the output service policy.

   Example

   ```
   Device(config-wireless-policy)# service-policy output platinum
   ```

8. Exit the policy configuration mode.

   Example

   ```
   Device(config-wireless-policy)# exit
   ```

9. Create a named object to apply policies to traffic classes.

   Example

   ```
   Device(config)# policy-map policy-sample 1
   ```

   Policy map names can contain alphabetic characters, hyphens, or underscores. They are case sensitive and can be up to 40 characters long.

10. Associate a class map with the policy map, and enter configuration mode for the specified system class.

Example

```
Device(config-pmap)# class class-map-name
```

11. Configure traffic policing.

Example

```
Device(config-pmap-c)# police 500,000
```

Valid values range from 8,000 to 200,000,000 bytes.

The metal policy enforces bidirectional rate limiting so wireless traffic meets the input and output bandwidth limits you specified.

Per-client bidirectional rate limiting

Describes per-client bi-directional rate limiting, presenting use cases, process workflows, prerequisites, restrictions, GUI-based configuration, AAA override integration, and verification procedures for policy enforcement.

A per-client bidirectional rate limit is a wireless traffic control feature that

- enforces, for each wireless client, bandwidth caps separately on both upstream and downstream traffic,
- ensures that each client receives a single aggregate limit, no matter how many concurrent flows or streams are active,
- addresses limitations of legacy per-flow rate limiting on 802.11ac Wave 2 APs in Flex local switching mode.

Additional reference information

Previously, per-flow rate limiting capped each client stream, such as a YouTube stream or an FTP transfer, independently. As a result, clients could exceed the intended per-client bandwidth limit.

With per-client bidirectional rate limits, the total bandwidth used by all of a client's streams cannot exceed the configured limit. This restriction applies no matter how many streams are active.

Per-client bidirectional rate limiting

If a controller limits each client to 1000 Kbps (1 megabit per second), and a client initiates both a YouTube and an FTP stream, both streams together will share the 1000 Kbps (1 megabit per second) limit, ensuring the client cannot exceed the cap.

Use case for per-client bidirectional rate limiting

Use case 1: configuring only default class map

If you configure the policy map with only the default class map and map it to the QoS client policy, the AP applies a per-client rate limit to each connected client.

Use case 2: changing from per client rate limit to per flow rate limit

If you configure the policy map with an additional class map along with the default class map and map it to the QoS client policy, the AP applies a per-flow rate limit to each client. Because the policy map includes multiple class maps, the AP clears any previously configured per-client rate limit values.

When the policy map includes more than one class map, the rate limit changes from per-client to per-flow, and the AP deletes the per-client rate limit values from the rate information token bucket.

Use case 3: changing from per flow rate limit to per client limit

If you remove the additional class map and the policy map contains only the default class map, the AP applies a per-client rate limit to each client.

Process for per-client bidirectional rate limiting

Process for per-client bidirectional rate limiting.

Summary

This section covers the high-level steps for per client bidirectional rate limiting.

Workflow

1. Configure a policy map for the WLAN using the policy profile.
2. Map the QoS-related policy map to the WLAN.
3. Configure policy map with the default class map.
4. Configure a different police rate value for the class default map.

> **Note**
>
> If the policy map has a class default with a valid police rate value, the AP applies that rate limit to the overall client data traffic flow.

5. Apply the policy map with class Default to the QoS client policy in the WLAN policy profile.

Prerequisites for per-client bidirectional rate limiting

Prerequisites for per-client bidirectional rate limiting.

- You can use this feature only with a QoS client policy. Ensure the policy profile contains only a QoS Policy or a policy target as client.
- If the policy map includes class default with a valid police rate value, the access point applies this rate limit to the the data traffic flow for the client.

Restrictions on per-client bidirectional rate limiting

Restrictions on per-client bidirectional rate limiting.

- If the policy map has a class map other than the Default class map, the per-client rate limit does not work on the AP.
- From Cisco IOS XE 17.5.x onwards, use AAA override to push attributes and achieve a per-client rate limit.

Configure per-client bidirectional rate limiting (GUI)

Configure per-client bidirectional rate limiting for each client by using the GUI.

Set limits on upload and download data rates for each client. This ensures fair bandwidth distribution.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click the Policy Profile Name.

   The **Edit Policy Profile** window is displayed.

   > **Note**
   >
   > The **Edit Policy Profile** window is displayed and configured in default class map only.

3. Choose the **QOS And AVC** tab.
4. In the **QoS Client Policy** settings, choose the policies from the **Egress** and **Ingress** drop-down lists.

   > **Note**
   >
   > You need to apply the default policy map to the QoS Client Policy.

5. Click **Update & Apply to Device**.

The selected policies in the policy profile enforce bidirectional rate limiting for each client.

Verify per client bi-directional rate limiting

Verify per client bi-directional rate limiting.

To verify whether per client is applied in AP, use this command:

```
Device# show rate-limit client
Config:
              mac vap rt_rate_out rt_rate_in rt_burst_out rt_burst_in
nrt_rate_out nrt_rate_in nrt_burst_out nrt_burst_in
A0:D3:7A:12:6C:5E   0           0           0            0           0
    0           0             0            0
Statistics:
            name     up down
        Unshaped      0    0
  Client RT pass 697610 8200
 Client NRT pass      0    0
 Client RT drops      0    0
Client NRT drops      0   16
               9    180    0
Per client rate limit:
              mac vap rate_out rate_in            policy
A0:D3:7A:12:6C:5E   0       88      23 per_client_rate_2
```

Configure bi-directional rate limiting using AAA override (CLI)

Configure BDRL using AAA override.

Enable WLAN policy profiles to enforce upstream and downstream bandwidth limits based on RADIUS server (e.g., Cisco ISE) attributes through AAA override.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile and enter wireless policy configuration mode.

   Example

   ```
   Device (config)# wireless profile policy profile-name
   ```

3. Configure AAA override to apply policies coming from the AAA server or Cisco Identity Services Engine (ISE) server.

   Example

   ```
   Device(config-wireless-policy)# aaa-override
   ```

   These attributes are available in the RADIUS server:

   - Airespace-Data-Bandwidth-Average-Contract: 8001
   - Airespace-Real-Time-Bandwidth-Average-Contract: 8002
   - Airespace-Data-Bandwidth-Burst-Contract: 8003
   - Airespace-Real-Time-Bandwidth-Burst-Contract: 8004
   - Airespace-Data-Bandwidth-Average-Contract-Upstream: 8005
   - Airespace-Real-Time-Bandwidth-Average-Contract-Upstream: 8006
   - Airespace-Data-Bandwidth-Burst-Contract-Upstream: 8007
   - Airespace-Real-Time-Bandwidth-Burst-Contract-Upstream: 8008

   Note

   8001, 8002, 8003, 8004, 8005, 8006, 8007, and 8008 are example rate-limit values.

The WLAN policy profile is configured to honor bi-directional rate limiting instructions received via AAA override from the RADIUS server.

Verify bi-directional rate-limit

Verify bi-directional rate-limit.

To verify the bi-directional rate limit, use this command:

```
Device# show wireless client mac-address E8-8E-00-00-00-71 detailClient MAC
Address : e88e.0000.0071
Client MAC Type     : Universally Administered Address
Client IPv4 Address : 192.0.2.1
Client Username     : e88e00000071
AP MAC Address      : 0a0b.0c00.0200
AP Name             : AP6B8B4567-0002
AP slot             : 0
Client State        : Associated
Policy Profile      : dnas_qos_profile_policy
Flex Profile        : N/A
Wireless LAN Id     : 10
WLAN Profile Name   : QoS_wlan
Wireless LAN Network Name (SSID): QoS_wlan
BSSID : 0a0b.0c00.0200
Connected For       : 28 seconds
Protocol            : 802.11n - 2.4 GHz
Channel             : 1
Client IIF-ID       : 0xa0000034
Association Id      : 10
Authentication Algorithm : Open System
Idle state timeout  : N/A
Session Timeout     : 1800 sec (Remaining time: 1777 sec)
Session Warning Time : Timer not running
Input Policy Name    : None
Input Policy State   : None
Input Policy Source  : None
Output Policy Name   : None
Output Policy State  : None
Output Policy Source : None
WMM Support          : Enabled
U-APSD Support       : Disabled
Fastlane Support     : Disabled
Client Active State  : In-Active
Power Save           : OFF
Supported Rates : 1.0,2.0,5.5,6.0,9.0,11.0,12.0,18.0,24.0,36.0,48.0,54.0
AAA QoS Rate Limit Parameters:
  QoS Average Data Rate Upstream             : 8005 (kbps)
  QoS Realtime Average Data Rate Upstream    : 8006 (kbps)
  QoS Burst Data Rate Upstream               : 8007 (kbps)
  QoS Realtime Burst Data Rate Upstream      : 8008 (kbps)
  QoS Average Data Rate Downstream           : 8001 (kbps)
  QoS Realtime Average Data Rate Downstream  : 8002 (kbps)
  QoS Burst Data Rate Downstream             : 80300 (kbps)
  QoS Realtime Burst Data Rate Downstream    : 8004 (kbps)
```

To verify the rate-limit details from the AP terminal, use this command

```
Device# show rate-limit client
Config:
mac vap rt_rate_out rt_rate_in rt_burst_out rt_burst_in nrt_rate_out nrt_rate_in
 nrt_burst_out nrt_burst_in
00:1C:F1:09:85:E7 0 8001 8002 8003 8004 8005 8006 8007 8008
Statistics:
name up down
Unshaped 0 0
Client RT pass 0 0
Client NRT pass 0 0
Client RT drops 0 0
Client NRT drops 0 0
```

```
Per client rate limit:
mac vap rate_out rate_in policy
```

How to configure wireless QoS

Provides instructions for configuring wireless QoS, including policy maps with class maps for voice and video, GUI and CLI steps for applying QoS policies to profiles and tags, AP policy tag attachment, and advanced mapping options such as DSCP to UP exception and trust upstream DSCP values.

Configure a policy map with class map (GUI)

Configure a policy map with class map using GUI.

Define and apply a QoS policy map with associated class maps to control network traffic behavior.

Use this task when you need to create or update a QoS policy map containing class maps, specifying how different types of network traffic are marked, policed, or dropped.

> **Note**
>
> Do not configure the Police (kbps) action for an SSID QoS policy used with FlexConnect local switching. Use FlexConnect central switching when per-SSID policing is required.

Before you begin

Use these steps to configure a policy map with class map using the GUI.

Procedure

1. Choose **Configuration** > **Services** > **QoS**.
2. Click **Add** to view the **Add QoS** window.
3. In the text box next to **Policy Name**, enter the name of the new policy map.
4. Click **Add Class-Maps**.
5. Configure **AVC** based policies or **User Defined** policies. To enable **AVC** based policies, and configure the following:
   a) Choose either **Match Any** or **Match All**.
   b) Choose the required **Mark Type**. If you choose **DSCP** or **User Priority**, you must specify the appropriate **Mark Value**.
   c) Check the **Drop** check box to drop traffic from specific sources.

      > **Note**
      >
      > When **Drop** is enabled, the **Mark Type** and **Police(kbps)** options are disabled.

   d) Select the required protocols from the **Available Protocol(s)** list based on the chosen **Match Type**. Move them to the **Selected Protocol(s)** list. The system drops traffic from these selected protocols.
   e) Click **Save**.

> **Note**
>
> To add more Class Maps, repeat steps 4 and 5.

> **Note**
>
> To add more Class Maps, repeat steps 4 and 5.

6. To enable the **User-Defined** QoS policy, configure these options:
   a) Choose either **Match Any** or **Match All**.
   b) From the drop-down list, choose either **ACL** or **DSCP** as the **Match Type**. Specify the appropriate **Match Value**.
   c) Choose the required **Mark Type** to associate with the mark label. If you choose *DSCP*, you must specify an appropriate **Mark Value**.
   d) Check the **Drop** check box to drop traffic from specific sources.

   > **Note**
   >
   > When **Drop** is enabled, the **Mark Type** and **Police(kbps)** options are disabled.

   e) Click **Save**.

   > **Note**
   >
   > To define actions for all the remaining traffic, in the Class Default, choose either **Mark**, **Police(kbps)**, or both as appropriate.

7. Click **Save & Apply to Device**.

The system deploys the defined policy map with its class maps and associated actions. QoS settings are enforced on network traffic as specified.

Configure a class map (CLI)

Configure a class map using commands.

Define and customize a class map to identify and match specific network traffic, such as voice and video, using CLI commands.

Use this procedure to configure class maps for voice and video traffic:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a class map.

   Example

   ```
   Device(config)# class-map class-map-name
   ```

3. Match the DSCP value in IPv4 and IPv6 packets.

   Example

   ```
   Device(config-cmap)# match dscp dscp-value
   ```

   > **Note**
   >
   > By default, the class map uses match-all.

4. Exit class map configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-cmap)# end
   ```

5. Verify class map details.

   Example

   ```
   Device# show class-map class_map_name
   ```

You have configured the class map with the specified matching criteria. It is ready for use in traffic policies.

Configuring Policy Profile to Apply QoS Policy (GUI)

Configure policy profile to apply QoS policy using GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. On the **Policy Profile** page, click the name of the policy profile.
3. In the **Edit Policy Profile** window, click the **QoS and AVC** tab.

4. Under **QoS SSID Policy**, choose the appropriate **Ingress** and **Egress** policies for WLANs.

> **Note**
>
> The ingress policies can be differentiated from the egress policies by the suffix *-up*. For example, the Platinum ingress policy is named *platinum-up*.

5. Under **QoS Client Policy**, choose the appropriate **Ingress** and **Egress** policies for clients.
6. Click **Update & Apply to Device**.

> **Note**
>
> Only custom policies are displayed under **QoS Client Policy**. AutoQoS policies are auto generated and not displayed for user selection.

Configure policy profile to apply QoS policy (CLI)

Configure policy profile to apply QoS policy using commands.

Apply a QoS policy to a WLAN policy profile using CLI.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the WLAN policy profile. Enter the wireless policy configuration mode.

Example

```
Device(config)# wireless profile policy profile-policy
```

3. Apply the policy.

Example

```
Device(config-wireless-policy)# service-policy client input policy-map-client
```

These options are available:

- **input**–Assigns the client policy for ingress direction on the policy profile.
- **output**–Assigns the client policy for egress direction on the policy profile.

4. Apply the policy to the Basic Service Set Identifier (BSSID).

   Example

   ```
   Device(config-wireless-policy)# service-policy input output policy-name
   ```

   These options are available:

   - **input**–Assigns the policy-map to all clients in WLAN.
   - **output**–Assigns the policy-map to all clients in WLAN.

5. Enable the wireless policy profile.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

The specified QoS policy is applied to the WLAN policy profile, and all clients associated with the WLAN receive the defined QoS treatment.

Apply policy profile to policy tag (GUI)

Apply a policy profile to a policy tag by using the GUI.

Associate a policy profile with a policy tag to manage network behavior for specific WLANs.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. On the **Manage Tags** page in the **Policy** tab, click **Add**.
3. In the **Add Policy Tag** window, enter a name and description for the policy tag.
4. Map the required WLAN IDs and WLAN profiles with appropriate policy profiles.
5. Click **Update & Apply to Device**.

The device applies the specified policy profile to the policy tag and controls WLAN behavior based on the configuration.

Apply policy profile to policy tag (CLI)

Apply policy profile to policy tag using commands.

Associate a policy profile with a policy tag through command-line configuration to enforce desired wireless policy settings.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure policy tag and enter the policy tag configuration mode.

   Example

   ```
   Device(config-policy-tag)# wireless tag policy policy-tag-name
   ```

3. Map a policy profile to a WLAN profile.

   Example

   ```
   Device(config-policy-tag)# wlan test policy profile-policy-name
   ```

4. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

   Example

   ```
   Device(config-policy-tag)# end
   ```

5. Display the configured policy tags.

   Example

   ```
   Device# show wireless tag policy summary
   ```

   Note

   To view the detailed information of a policy tag, use the **show wireless tag policy detailed** *policy-tag-name* command.

The policy profile is applied to the policy tag. You can verify the configuration through summary and detailed display commands.

Attach policy tag to an AP

Attach a policy tag to an AP.

Assign a policy tag to a wireless AP to determine its network and functional behavior.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the AP, then enter the AP profile configuration mode.

Example

```
Device(config)# ap F866.F267.7DFB
```

3. Map a policy tag to the AP.

Example

```
Device(config-ap-tag)# policy-tag policy-tag-name
```

4. Save the configuration. Exit configuration mode. Return to privileged EXEC mode.

Example

```
Device(config-ap-tag)# end
```

5. Display the AP details and the tags associated with it.

Example

```
Device# show ap tag summary
```

The specified AP is associated with the chosen policy tag. You can verify the mapping by displaying the AP details.

Configure custom QoS mapping (CLI)

Configure custom QoS mapping.

Create a custom mapping between IP DSCP values and 802.11e user priorities to support Hotspot 2.0 interworking on the WLAN.

The system creates a map between the 802.11e user priorities and the IP differentiated services code point (DSCP) for interworking with IP networks. Enable Hotspot 2.0 on the WLAN to support mapping exception.

> **Note**
>
> Custom QoS mapping applies only to Hotspot 2.0.

Specify the mapping by assigning DSCP ranges to individual user priority values, and set exceptions by mapping DSCP values to UP values one-to-one. If you enable a QoS map and do not add custom mappings, the system uses default values.

> **Note**
>
> Egress = Downstream = Output; and Ingress = Upstream = Input

The table shows a QoS map, where an AP provides a wireless client with the required mapping from IP DSCP to 802.11e user priority.

Table 154: Default DSCP-Range-to-User Priority Mapping

IP DSCP Range	802.11e User Priority
0-7	0
8-15	1
16-23	2
24-31	3
32-39	4
40-47	5
48-55	6
56-63	7

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

3. Configure DSCP-to-user priority mapping.

   Example

   ```
   Device(config-ap-profile)# qos-map dscp-to-up-range user-priority up-to-dscp
    dscp-start dscp-end
   ```

   You can configure up to eight entries–one for each *user-priority* value. If you do not configure a custom value, the system sends a non-configured value (0xFF) to the AP.

Use the **no** form of this command to disable the configuration. To delete all the custom mappings, use the **no dscp-to-up-range** command.

When you apply custom DSCP to user priority mappings for an AP profile, the system translates the QoS policy correctly for Hotspot 2.0 clients.

Configure a DSCP-to-user priority mapping exception

Configure DSCP-to-user priority mapping exception.

Define custom exceptions so you can map IP DSCP values to 802.11e user priorities in an AP profile. This provides control over QoS behavior.

When you configure a QoS mapping or exception, the system creates a custom QoS map. The map is sent to the corresponding AP.

If you do not configure DSCP-to-user priority mapping or exception entries, the system uses an empty QoS map.

The table displays exceptions with a one-to-one mapping between DSCP values and user priority values.

Table 155: Default DSCP-Range-to-User Priority Mapping Exceptions

IP DSCP	802.11e User Priority
0	0
2	1
4	1
6	1
10	2
12	2
14	2
18	3
20	3
22	3
26	4
34	5
46	6
48	7
56	7

> **Note**
>
> Disable voice admission control for user priorities 6 and 7 using the controller GUI. To disable **Admission Control (ACM)**, choose **Configuration** > **Radio Configurations** > **Media Parameters**.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap profile name
   ```

3. Configure a DSCP-to-user priority exception.

   Example

   ```
   Device(config-ap-profile)# qos-map dscp-to-up-exception dscp-num user-priority
   ```

With these settings, your AP profile has customized DSCP-to-user priority mapping exceptions. This configuration provides tailored QoS mapping for wireless clients.

Configure trust upstream DSCP value

Configure trust upstream DSCP value.

Configure the wireless controller to trust the upstream DSCP value, not the user priority. This optimizes end to end QoS marking.

The controller marks the 802.11 user priority value in Traffic Identifier (TID) field based on the DSCP value in IP header.

> **Note**
>
> The AP forwards the DSCP value to Air when the 802.11 user priority value is set.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure an AP profile and enter AP profile configuration mode.

Example

```
Device(config)# ap profile ap profile-name
```

3. Configure the AP to trust upstream DSCP instead of user priority.

Example

```
Device(config-ap-profile)# qos-map trust-dscp-upstream
```

Use the **no** form of the command to disable the configuration.

> **Note**
>
> Starting with Cisco IOS XE 17.4.1 release, **qos-map trust-dscp-upstream** is the default setting. As a result, client DSCP is maintained all the way through by default.

> **Note**
>
> When you enable the **trust-dscp-upstream** command, the DSCP value is 18. If you do not configure any value, Silver is the default.

The AP profile is set to trust upstream DSCP values, ensuring that client QoS markings are preserved from end to end.

175 Wireless Auto-QoS

Topics:

- Auto QoS information

This chapter explains how to enable, disable, clear, view, and rollback AutoQoS on wireless policy profiles using CLI and GUI, automating QoS deployment and allowing manual override or troubleshooting.

Auto QoS information

Lists wireless Auto QoS features that automate deployment of wireless QoS policies and traffic prioritization.

Wireless Auto QoS automates deployment of wireless QoS features. It has a set of predefined profiles which can be further modified by the customer to prioritize different traffic flows. Auto-QoS matches traffic and assigns each matched packet to qos-groups. This allows the output policy map to put specific qos-groups into specific queues, including into the priority queue.

Autoqos policy configuration
Table 156: Autoqos policy configuration

Mode	Client Ingress	Client Egress	BSSID Ingress	BSSID Egress	Port Ingress	Port Egress	Radio
Voice	N/A	N/A	P3	P4	N/A	P7	ACM on
Guest	N/A	N/A	P5	P6	N/A	P7	
Fastlane	N/A	N/A	N/A	N/A	N/A	P7	edca-parameters fastlane
Enterprise-avc	N/A	N/A	P1	P2	N/A	P7	

P1	AutoQos-4.0-wlan-ET-SSID-Input-AVC-Policy
P2	AutoQos-4.0-wlan-ET-SSID-Output-Policy
P3	platinum-up
P4	platinum
P5	AutoQos-4.0-wlan-Dot1x-Output-Policy
P6	AutoQos-4.0-wlan-GT-SSID-Output-Policy
P7	AutoQos-4.0-wlan-Port-Output-Policy

Configure wireless auto-QoS on profile policy (CLI)

Configure wireless Auto QoS to automate deployment of wireless QoS features.

You can enable AutoQoS on a profile policy to automate deployment of wireless QoS features.

You can enable AutoQoS on a profile policy.

Procedure

1. Enable privileged EXEC mode.

Example

```
Device# enable
```

Enter your password if prompted.

2. Configure AutoQoS wireless policy.

Example

```
Device# wireless autoqos policy-profile test-profile mode voice
```

- **enterprise-avc**–Enables AutoQoS Wireless Enterprise AVC Policy.
- **fastlane**–Enable AutoQoS Wireless Fastlane Policy.
- **guest**–Enable AutoQoS Wireless Guest Policy.
- **voice**–Enable AutoQoS Wireless Voice Policy.

> **Note**
>
> AutoQoS MIB attribute does not support full functionality with service policy. Service policy must be configured manually. Currently, there is only support for AutoQoS mode.

What to do next

> **Note**
>
> After enabling AutoQoS, we recommend that you wait for a few seconds for the policy to install and then try and modify the AutoQoS policy maps if required; or retry if the modification is rejected.

Disable wireless autoqos

Disable wireless AutoQoS globally on the device.

Disable wireless AutoQoS functionality to manually control QoS parameters or troubleshoot QoS-related issues.

Wireless AutoQoS automatically configures QoS settings for optimal wireless performance. You may need to disable it when implementing custom QoS policies or when troubleshooting network issues.

Procedure

1. Enable privileged EXEC mode.

Example

```
Device> enable
```

Enter your password if prompted.

2. Shut down the policy profile.

Example

```
Device# shutdown
```

Shuts down the policy profile.

3. Globally disable wireless AutoQoS.

Example

```
Device# wireless autoqos disable
```

Globally disables wireless AutoQoS.

4. Enable the wireless policy profile.

Example

```
Device# no shutdown
```

Enables the wireless policy profile.

> **Note**
>
> Disabling Auto QoS does not reset global radio configurations like CAC and EDCA parameters.

Wireless AutoQoS is now disabled globally, allowing for manual QoS configuration or troubleshooting without automatic QoS interference.

Rollback auto-QoS configuration (GUI)

Configure the rollback of AutoQoS settings using the GUI interface.

Before you begin

Follow these steps to rollback AutoQoS configuration using the GUI:

Procedure

---

1. Choose **Configuration** > **Services** > **QoS**.
2. Click **Disable AutoQoS**.
3. Click **Yes** to confirm.

Rollback AutoQoS configuration (CLI)

Configure the reset of AutoQoS configuration templates to restore default settings.

Before you begin

> **Note**
>
> AutoQoS MIB attribute does not support the full functionality with service policy. Currently, there is only support for AutoQoS mode. Service policy must be configured manually.

Follow these steps to rollback AutoQoS configuration:

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Reset AutoQoS configuration.

   Example

   ```
   Device# clear platform software autoqos config template guest
   ```

   - enterprise-avc–Resets AutoQoS Enterprise AVC Policy Template.
   - guest–Resets AutoQoS Guest Policy Template.

Clear wireless autoqos policy profile (GUI)

Configure wireless AutoQoS policy profile to remove automatic Quality of Service settings using the GUI.

Before you begin

Follow these steps to clear wireless AutoQoS policy profile using the GUI

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click on the **Policy Profile Name**.
3. Go to **QOS and AVC** tab.
4. From the **Auto Qos** drop-down list, choose **None**.
5. Click **Update & Apply to Device**.

Clear wireless autoqos policy profile (CLI)

Clear wireless AutoQoS policy profile using commands.

Clear the AutoQoS configuration from a wireless policy profile to reset or remove QoS settings.

Use this procedure when you need to remove AutoQoS settings from a wireless policy profile, typically during reconfiguration or troubleshooting scenarios.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Shut down the policy profile.

   Example

   ```
   Device# shutdown
   ```

3. Clear the configured AutoQoS wireless policy.

   Example

   ```
   Device# wireless autoqos policy-profile policy-name mode clear
   ```

   Example

   ```
   Device# wireless autoqos policy-profile test-profile mode clear
   ```

4. Enable the wireless policy profile.

   Example

   ```
   Device# no shutdown
   ```

The AutoQoS configuration is cleared from the specified wireless policy profile, and the profile is re-enabled.

View Auto-QoS on policy profile (CLI)

Configure AutoQoS viewing on policy profile to display detailed parameters and configurations.

Before you begin

AutoQoS is supported on the local mode and flex mode. AutoQoS configures a set of policies and radio configurations depending on the template. It is possible to override the service-policy that is configured by AutoQoS. The latest configuration takes effect, with AAA override policy being of highest priority.

Follow these steps to view AutoQoS on policy profile:

Procedure

1. **enable**

   Example

   ```
   Device#enable
   ```

   Enables privileged EXEC mode. Enter your password if prompted.

2. **show wireless profile policy detailed** *policy-profile-name*

   Example

   ```
   Device# show wireless profile policy detailed testqos
   ```

   Shows policy-profile detailed parameters.

   Note

   When the AutoQoS mode is configured to voice within the policy profile, any modifications to the global ACM settings require a shutdown and subsequent no shutdown operation on the policy profile to activate the changes. Without performing this shut and no shut cycle, the updated ACM configurations are not applied to the beacon frames. This causes the beacon to continue broadcasting the old ACM values.

   This process ensures that the revised ACM settings are correctly propagated and reflected throughout the wireless environment, thereby maintaining accurate call admission control enforcement.

176 Native Profiling

Topics:

- Native profiling

This chapter guides administrators through configuring and verifying device-based profiling using HTTP/DHCP: creating class maps, service templates, parameter maps and policy maps, enabling profiling in local mode, and applying service templates to enforce per-device policies.

Native profiling

Describes the native profiling feature that allows device profiling based on HTTP and DHCP to identify end devices and enforce device-based policies.

Native profiling is a network security feature that

- profiles devices based on HTTP and DHCP to identify end devices on the network
- configures device-based policies and enforces these policies per user or per device policy on the network, and
- allows profiling of mobile devices and basic onboarding of the profiled devices to a specific VLAN.

Native profiling characteristics

Policies assign ACL and QoS or configure session timeouts.

The policies are defined based on the following attributes:

- User group or user role
- Device type such as Windows clients, smartphones, tablets, and so on
- Service Set Identifier (SSID)
- Location, based on the access point group that the end point is connected to
- Time of the day
- Extensible Authentication Protocol (EAP) type, to check what EAP method that the client is getting connected to

When a wireless client joins an access point, certain QoS policies get enforced on the access point. One such feature is the native profiling for both upstream and downstream traffic at AP. The native profiling feature when clubbed with AAA override supports specific set of policies based on the time of day and day of week. The AAA override then applies these policies coming from a RADIUS server to the access point.

Let's consider a use case of time of the day in conjunction with user role. Usually, the user role is used as an extra matching criteria along with the time of day. You can club the time of day usage with any matching criteria to get the desired result. The matching will be performed when the client joins the controller .

You can configure policies as two separate components:

- Defining policy attributes as service templates that are specific to clients joining the network and applying policy match criteria
- Applying match criteria to the policy.

> **Note**
>
> Before proceeding with the native profile configuration, ensure that HTTP Profiling and DHCP Profiling are enabled.

> **Note**
>
> Native profiling is not supported with FlexConnect Local Authentication and Local Switching. Hence, do not configure **no central switching**, **no central authentication**, and **subscriber-policy-name** *name* commands together. ISSU will fail for this type of configuration. Ensure that you remove the configuration before attempting ISSU.

To configure Native Profiling, use one of the following procedures:

- Create a service template
- Create a class map

  > **Note**
  >
  > You can apply a service template using either a class map or parameter map.

- Create a parameter-map and associate the service template to parameter-map
  - Create a policy map
    1. If class-map has to be used: Associate the class-map to the policy-map and associate the service-template to the class-map.
    2. If parameter-map has to be used: Associate the parameter-map to the policy-map
  - Associate the policy-map to the policy profile.

Create a class map (GUI)

Configure a class map using GUI.

Procedure

1. Click **Configuration** > **Services** > **QoS**.
2. In the **QoS – Policy** area, click **Add** to create a new QoS Policy or click the one you want to edit.
3. Add **Add Class Map** and enter the details.
4. Click **Save**.
5. Click **Update and Apply to Device**.

Create a class map (CLI)

Configure a class map using commands for more granular control than GUI options.

> **Note**
>
> Configuration of class maps via CLI offer more options and can be more granular than GUI.

Before you begin

Follow these steps to create a class map using CLI commands:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the class map type and name.

   Example

   ```
   Device(config)# class-map type control subscriber match-any cls_user
   ```

3. Specify the class map attribute filter criteria.

   Example

   ```
   Device(config-filter-control-classmap)# match username ciscoise
   ```

4. Specify the class map type and name.

   Example

   ```
   Device(config)# class-map type control subscriber match-any cls_userrole
   ```

5. Specify the class map attribute filter criteria.

   Example

   ```
   Device(config-filter-control-classmap)# match user-role engineer
   ```

6. Specify the class map type and name.

   Example

   ```
   Device(config)# class-map type control subscriber match-any cls_oui
   ```

7. Specify the class map attribute filter criteria.

   Example

   ```
   Device(config-filter-control-classmap)# match oui 48.f8.b3
   ```

8. Specify the class map type and name.

   Example

   ```
   Device(config)# class-map type control subscriber match-any cls_mac
   ```

9. Specify the class map attribute filter criteria.

   Example

   ```
   Device(config-filter-control-classmap)# match mac-address 0040.96b9.4a0d
   ```

10. Specify the class map type and name.

    Example

    ```
    Device(config)# class-map type control subscriber match-any cls_devtype
    ```

11. Specify the class map attribute filter criteria.

    Example

    ```
    Device(config-filter-control-classmap)# match device-type windows
    ```

12. Specify the class map type and name.

    Example

    ```
    Device(config)# class-map type control subscriber match-all match_tod
    ```

13. Specify a match to the time of day.

    Example

    ```
    Device(config-filter-control-classmap)# match join-time-of-day 10:30 12:30
    ```

    Here, join time is considered for matching. For example, if the match filter is set from 11:00 am to 2:00 pm, a device joining at 10:59 am is not considered, even if it acquires credentials after 11:00 am.

    Here,

    start-time and *end-time* specifies the 24-hour format.

    Use the **show class-map type control subscriber name** *name* command to verify the configuration.

> **Note**
>
> You should also disable AAA override for this command to work.

14. Match name using the device type.

Example

```
Device(config-filter-control-classmap)# match device-type android
```

Type a question mark (?) after the device type and select the device from the list.

> **Note**
>
> You should enable the device classifier for the device list to be populated.

Create a service template (GUI)

Create a service template using the GUI interface to define network security policies.

Procedure

1. Choose **Configuration** > **Security** > **Local Policy**.
2. On the **Local Policy** page, **Service Template** tab, click **ADD**.
3. In the **Create Service Template** window, enter the following parameters:
   - **Service Template Name:** Enter a name for the template.
   - **VLAN ID:** Enter the VLAN ID for the template. Valid range is between 1 and 4094.
   - **Session Timeout (secs):** Sets the timeout duration for the template. Valid range is between 1 and 65535.
   - **Access Control List:** Choose the Access Control List from the drop-down list.
   - **Ingress QOS:** Choose the input QoS policy for the client from the drop-down list
   - **Egress QOS:** Choose the output QoS policy for the client from the drop-down list.
4. Click **Save & Apply to Device**.

Create a service template (CLI)

Configure a service template using CLI commands to define network parameters and policies.

Before you begin

Follow these steps to create a service template using CLI commands:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter service template configuration mode.

   Example

   ```
   Device(config)# service-template svc1
   ```

3. Specify the VXLAN network identifier (VNID).

   Example

   ```
   Device(config-service-template)# vnid test
   ```

   Use the **show service-template** *service-template-name* command to verify the configuration.

4. Specify the access list to be applied.

   Example

   ```
   Device(config-service-template)# access-group acl-auto
   ```

5. Specify VLAN ID.

   Example

   ```
   Device(config-service-template)# vlan 10
   ```

   Valid range is from 1-4094.

6. Specify session timeout value for a service template.

   Example

   ```
   Device(config-service-template)# absolute-timer 1000
   ```

   Valid range is from 1-65535.

7. Configure an input QoS policy for the client.

   Example

   ```
   Device(config-service-template)# service-policy qos input in_qos
   ```

8. Configure an output QoS policy for the client.

Example

```
Device(config-service-template)# service-policy qos output out_qos
```

Create a parameter map (CLI)

Configure a parameter map to specify attribute filter criteria and associated service templates.

Before you begin

Follow these steps to create a parameter map:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the parameter map type and name.

   Example

   ```
   Device(config)# parameter-map type subscriber attribute-to-service param
   ```

3. Specify the parameter map attribute filter criteria.

   Example

   ```
   Device(config-parameter-map-filter)# 1 map device-type eq "windows"
   mac-address eq 3c77.e602.2f91 username eq "cisco"
   ```

   Multiple filters are used in the example provided here.

4. Specify the service template and its precedence.

   Example

   ```
   Device(config-parameter-map-filter-submode)# 1 service-template svc1 precedence
    150
   ```

Create a policy map (GUI)

Configure a policy map using the GUI to define service templates and match criteria for devices.

Before you begin

Follow these steps to create a policy map using the GUI:

Procedure

1. Choose **Configuration** > **Security** > **Local Policy** > **Policy Map** tab.
2. Enter a name for the Policy Map in the **Policy Map Name** text field.
3. Click **Add**
4. Choose the service template from the **Service Template** drop-down list.
5. For the following parameters select the type of filter from the drop-down list and enter the required match criteria
   - Device Type
   - User Role
   - User Name
   - OUI
   - MAC Address
6. Click **Add Criteria**
7. Click **Update & Apply to Device**.

Create a policy map (CLI)

Configure a policy map using commands.

Create and configure a policy map to control subscriber behavior and apply service templates in wireless environments.

Policy maps define how the system responds to specific events and conditions. This procedure creates a subscriber control policy map that can be applied to wireless profiles for client management and service provisioning.

Before you begin

Before removing a policy map or parameter map, you should remove it from the target or shut down the WLAN profile or delete the session.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the policy map type.

   Example

   ```
   Device(config)# policy-map type control subscriber policy-map-name
   ```

   Example

   ```
   Device(config)# policy-map type control subscriber polmap5
   ```

3. Specify the match criteria to the policy map.

Example

```
Device(config-event-control-policymap)# event identity-update match-all
```

4. You can apply a service template using either a class map or a parameter map, as shown here.
   - Configure the local profiling policy class map number and specify how to perform the action.
   - Activate the service template.
   - Map an identity-update attribute to an auto-configured template.

   Example

   The following example shows how a class-map with a service-template has to be applied:

   ```
   Device(config-class-control-policymap)# class-num class class-map-name
   do-until-failure
   Device(config-action-control-policymap)# action-index activate service-template
    service-template-name
   ```

   Example

   The following example shows how a parameter map has to be applied (service template is already associated with the parameter map 'param' while creating it):

   ```
   Device(config-action-control-policymap)# action-index map attribute-to-service
    table parameter-map-name
   ```

5. Exit configuration mode.

   Example

   ```
   Device(config-action-control-policymap)# end
   ```

6. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

7. Configure a wireless policy profile.

   Example

   ```
   Device(config)# wireless profile policy wlan-policy-profile-name
   ```

   > ⚠ **Caution**
   >
   > Do not configure aaa-override for native profiling under a named wireless profile policy. Native profiling is applied at a lower priority than AAA policy. If aaa-override is enabled, the AAA policies will override native profile policy.

8. Add a description for the policy profile.

Example

```
Device(config-wireless-policy)# description profile-policy-description
```

Example

```
Device(config-wireless-policy)# description "default policy profile"
```

9. Configure DHCP TLV caching on a WLAN.

Example

```
Device(config-wireless-policy)# dhcp-tlv-caching
```

10. Configure client HTTP TLV caching on a WLAN.

Example

```
Device(config-wireless-policy)# http-tlv-caching
```

11. Configure the subscriber policy name.

Example

```
Device(config-wireless-policy)# subscriber-policy-name policy-name
```

Example

```
Device(config-wireless-policy)# subscriber-policy-name polmap5
```

12. Configure a VLAN name or VLAN ID.

Example

```
Device(config-wireless-policy)# vlan vlan-id
```

Example

```
Device(config-wireless-policy)# vlan 1
```

13. Save the configuration.

Example

```
Device(config-wireless-policy)# no shutdown
```

The policy map is created and configured with the specified parameters, and the wireless policy profile is activated with the associated subscriber policy.

Configure native profiling in local mode (CLI)

Learn how to configure native profiling in local mode by enabling central switching in the policy profile.

To enable native profiling for wireless users in local mode by updating the policy profile configuration.

To configure native profiling in the local mode, you must follow the steps described in Create a policy map (CLI) on page 2211 . In the policy profile, you must enable central switching as described in the step given below in order to configure native profiling.

Procedure

Enable central switching.

Example

```
Device(config-wireless-policy)# central switching
```

Native profiling is now enabled in local mode. The device uses central switching, allowing profiling information to be collected and used for network management.

Verify native profile configuration

Verify native profile configuration.

Use these **show** commands to verify the native profile configuartion:

```
Device# show wireless client device summary

Active classified device summary
MAC Address       Device-type                       User-role
            Protocol-map
-------------------------------------------------------------------------------------
1491.82b8.f94b    Microsoft-Workstation             sales
                        9
1491.82bc.2fd5    Windows7-Workstation              sales
                       41
```

```
Device# show wireless client device cache

Cached classified device info

MAC Address       Device-type                       User-role
            Protocol-map
-------------------------------------------------------------------------------------
2477.031b.aa18    Microsoft-Workstation
                        9
30a8.db3b.a753    Un-Classified Device
                        9
4400.1011.e8b5    Un-Classified Device
                        9
980c.a569.7dd0    Un-Classified Device
```

```
Device# show wireless client mac-address 4c34.8845.e32c detail | s
Session Manager:
  Interface :
  IIF ID            : 0x90000002
  Device Type       : Microsoft-Workstation
  Protocol Map      : 0x000009
  Authorized        : TRUE
  Session timeout   : 1800
  Common Session ID: 78380209000000174BF2B5B9
  Acct Session ID   : 0
  Auth Method Status List
   Method : MAB
    SM State         : TERMINATE
    Authen Status    : Success
  Local Polices:
   Service Template : wlan_svc_C414.3CCA.0A51 (priority 254)
    Absolute-Timer   : 1800
  Server Polices:
  Resultant Policies:
  Filter-ID         : acl-auto
  Input QOS         : in_qos
  Output QOS        : out_qos
  Idle timeout      : 60 sec
  VLAN               : 10
  Absolute-Timer    : 1000
```

Use this **show** command to verify the class map details for a class map name:

```
Device# show class-map type control subscriber name test
Class-map              Action                        Exec  Hit  Miss  Comp
---------               ------                       ----  ---  ----   ---
match-any test     match day Monday                    0     0    0     0
match-any test     match join-time-of-day 8:00 18:00   0     0    0     0
Key:
  "Exec" - The number of times this line was executed
  "Hit"  - The number of times this line evaluated to TRUE
  "Miss" - The number of times this line evaluated to FALSE
  "Comp" - The number of times this line completed the execution of its
       condition without a need to continue on to the end
```

177 Air Time Fairness

Topics:

- Air time fairness

Air time fairness

Describes how Cisco ATF allows administrators to prioritize device categories, ensuring selected groups receive WLAN traffic more frequently.

Cisco Air Time Fairness (ATF) is a WLAN management feature that

- allocates Wi-Fi air time for user groups or device categories
- allows network administrators to group devices of a defined category and enables some groups to receive traffic from the WLAN more frequently than other groups
- provides air time fairness defined by the network administrator and not by the network, and
- dynamically adapts to changing conditions in a WLAN.

Air time fairness capabilities

Cisco ATF provides these additional capabilities:

- Provides a simplified mechanism for allocating air time.
- Enables a more efficient fulfillment of service-level agreements.
- Augments standards-based Wi-Fi QoS mechanisms.

By enabling network administrators to define what fairness means in their environments with regards to the amount of air time per client group, the amount of traffic is also controlled.

To control air time on a percentage basis, the air time including both uplink and downlink transmissions of a client or SSID is continuously measured.

Only air time in the downlink direction, that is AP to client, can be controlled accurately by the AP. Although air time in the uplink direction, that is client to AP can be measured, it cannot be controlled. Although the AP can constrain air time for packets that it sends to clients, the AP can only measure air time for packets that it hears from clients because it cannot strictly limit their air time.

Cisco ATF establishes air time limits (defined as a percentage of total air time) and applies those limits on a per SSID basis, where the SSID is used as a parameter to define a client group. Other parameters can be used as well to define groups of clients. Furthermore, a single air time limit can be applied to individual clients.

If the air time limit for an SSID (or client) is exceeded, the packets in the downlink direction are dropped. Dropping downlink packets (AP to client) frees up air time whereas dropping uplink packets (client to AP) does not do anything to free up air time because the packet has already been transmitted over the air by the client.

Client fair sharing ensures that all clients in an SSID or WLAN are treated equally based on their utilization of the radio bandwidth:

- The percentage of air time to be given to each client is recomputed every time a client connects or disconnects.
- Client fair sharing is applicable only to downstream traffic.
- Clients can be categorized into usage groups at the policy level.
- Client-based ATF metrics accumulation is performed in the transmit complete routine. This allows the air time that is unused by clients in low-usage or medium-usage groups to be accumulated to a common share pool bucket where the high-usage clients can be replenished.

Cisco ATF is supported on the following APs:

- Cisco Aironet 2700 Series Access Points
- Cisco Aironet 3700 Series Access Points

- Cisco Aironet 2800 Series Access Points
- Cisco Aironet 3800 Series Access Points
- Cisco Aironet 4800 Series Access Points
- Cisco Aironet 1540 Series Access Points
- Cisco Aironet 1560 Series Access Points

Note

Cisco ATF is supported on MESH, if APs support ATF. ATF is supported on FlexConnect mode and the Local mode.

Note

Cisco Catalyst APs offer capabilities that are equivalent to ATF by leveraging the enhancements in the Wi-Fi 6 and 6E protocols. 802.11ax features such as OFDMA, bidirectional MU-MIMO, and BSS coloring, combined with the advanced QoS features in the Cisco Catalyst 9800 Series Wireless Controllers, help resolve scheduling and congestion problems, accommodate multiple users at the same time, and allocate bandwidth more efficiently.

Cisco ATF operates in the following modes:

- Monitor mode: View the air time, report air time usage for all AP transmissions, view reports per SSID or WLAN, view reports per site group/tag, report air time usage at periodic intervals. No enforcement as part of Monitor mode.
- Enforce Policy mode: Enforce air time based on configured policy, enforce air time on a WLAN, all APs connected in a Cisco Catalyst 9800 Series Wireless Controller network, or per site group/tag.

Restrictions on Cisco air time fairness

Lists the restrictions on Cisco Air Time Fairness implementation and configuration.

This reference provides the specific restrictions that apply when implementing and configuring Cisco Air Time Fairness (ATF) functionality.

- Cisco ATF can be implemented only on data frames in the downstream direction.
- When ATF is configured in per-SSID mode, all the WLANs are disabled before you enter any ATF configuration commands. The WLANs are enabled after you enter all the ATF commands.

Cisco air time fairness use cases

Describes common deployment scenarios for Cisco Air Time Fairness (ATF) in various network environments.

A Cisco Air Time Fairness use case is a deployment scenario that

- demonstrates how air time allocation can be partitioned between different user groups or service providers in shared WLAN environments
- addresses specific network sharing requirements in venues such as public hotspots, educational institutions, enterprises, and managed hotspot services, and

- enables fair distribution of wireless bandwidth resources based on business or operational requirements.

Common ATF deployment scenarios

The following scenarios illustrate typical ATF implementations across different environments:

- **Public hotspots (stadium/airport/convention center/other)**: A public network is sharing a WLAN between two (or more) service providers and the venue. Subscribers to each service provider can be grouped and allocated a certain percentage of air time.
- **Education**: A university is sharing a WLAN between students, faculty, and guests. The guest network can be further partitioned by the service provider, for distribution of bandwidth privileges to the guests. Each group can be assigned a certain percentage of air time.
- **Enterprise/hospitality/retail**: The venue is sharing a WLAN between employees and guests. The guest network can be further partitioned by service provider. The guests could be sub-grouped by tier of service type with each subgroup being assigned a certain percentage of air time, for example a paid group is entitled for more air time than the free group.
- **Time shared managed hotspot**: The business entity managing the hotspot, such as a service provider or an enterprise, can allocate and subsequently lease air time to other business entities.

Configure Cisco air time fairness

Describes the high-level steps to configure Cisco Air Time Fairness (ATF) including policy creation, WLAN configuration, and monitoring.

Summary

The key components involved in configuring Cisco Air Time Fairness are:

- Monitor mode: Optional network usage determination tool
- Cisco ATF policies: Configuration rules that define air time allocation
- WLAN ATF policy assignment: Network or site group/tag level policy application
- Optimization settings: Performance enhancement configurations
- ATF statistics monitoring: Ongoing performance tracking and validation

Workflow

The configuring Cisco Air Time Fairness process involves these stages:

1. Enable Monitor mode to determine network usage.
2. Create Cisco ATF policies.
3. Add WLAN ATF policies per network or per site group/tag.
4. Determine, if optimization must be enabled.
5. Periodically check the Cisco ATF statistics.

Create a Cisco ATF profile (GUI)

Configure a Cisco ATF profile using the GUI.

Before you begin

Follow these steps to create a Cisco ATF profile using the GUI:

Procedure

1. Choose **Configuration** > **Wireless** > **Air Time Fairness**.
2. Click the **Profiles** tab and click the **Add** button, to create a new ATF policy.

   The **Add ATF Policy** window is displayed.
3. Specify a name, ID, and weight to the ATF policy.

   Weighted ratio is used instead of percentages so that the total can exceed 100. The minimum weight that you can set is 5. For example, if you configure the weight as 50, this means that the air time for this ATF profile is 50% when applied to an policy profile.
4. Use the slider to enable or disable the **Client Sharing** feature.

   When you enable this option in the Web UI, the default ATF configuration is set to **Enforce** and not **Monitor**.
5. Click **Apply to Device**.

Create Cisco ATF profile (CLI)

Create a Cisco ATF profile using commands.

Create and configure a Cisco ATF (Airtime Fairness) profile to manage airtime allocation and ensure fair distribution of wireless resources among clients.

Cisco ATF profiles are used to control how airtime is allocated among wireless clients on an access point. This helps ensure that slower clients do not monopolize airtime at the expense of faster clients, improving overall network performance.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Create a new Cisco ATF policy.

   Example

   ```
   Device(config)# wireless profile airtime-fairness atf-policy-name
   atf-profile-id
   ```

   Example

   ```
   Device(config)# wireless profile airtime-fairness atf-policy-name 1
   ```

   - *atf-policy-name*–Enters the ATF profile name.
   - *atf-profile-id*–Enters the ATF profile ID. Range is from 0 to 511.
3. Add a weight to the Cisco ATF policy.

Example

```
Device(config-config-atf)# weight policy-weight
```

Example

```
Device(config-config-atf)# weight 5
```

- *policy-weight*–Enters the policy weight. Range is from 5 to 100.

4. Enable client sharing for Cisco ATF policy.

Example

```
Device(config-config-atf)# client-sharing
```

5. Return to privileged EXEC mode.

Example

```
Device(config-config-atf)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The Cisco ATF profile is created and configured with the specified weight and client sharing settings, ready to be applied to wireless policies for airtime management.

Attach Cisco ATF profile to a policy profile (GUI)

Configure Cisco ATF profile attachment to a policy profile using GUI.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click **Add**.

   The **Add Policy Profile** window is displayed.
3. Click the **Advanced** tab.
4. Under the **Air Time Fairness Policies** section, select the required policy for 2.4 GHz and 5 GHz policies.
5. Click **Apply to Device**.

Attach Cisco ATF profile to a policy profile (CLI)

Configure attachment of a Cisco ATF profile to a policy profile using commands.

Attach a Cisco ATF profile to a policy profile to enable airtime fairness management for wireless clients.

This procedure is used when you need to associate an existing ATF policy with a policy profile to control airtime allocation across wireless clients on specific radio bands.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create policy profile for the WLAN.

   Example

   ```
   Device(config)# wireless profile policy profile-name
   ```

   The *profile-name* is the profile name of the policy profile.

3. Configure air time fairness policy for 2.4- or 5-GHz radio.

   Example

   ```
   Device(config-wireless-policy)# dot11 band airtime-fairness atf-policy-name
   ```

   Example

   ```
   Device(config-wireless-policy)# dot11 24ghz airtime-fairness atf-policy-name
   ```

   The *band* can be either 24ghz or 5ghz. The *atf-policy-name* is the name of the air time fairness policy.

   > **Note**
   >
   > You can assign the same ATF policy to both 2.4-GHz and 5-GHz radios (or) have two different ATF policies as well.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The Cisco ATF profile is now attached to the policy profile, enabling airtime fairness management for the specified radio band.

Enable ATF in the RF profile (GUI)

Configure ATF in the RF profile using GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF**.
2. Click **Add**.
   The **Add RF Profile** window is displayed.
3. Click the **Advanced** tab.
4. Under the **ATF Configuration** section, complete the following :
   a) Use the slider to enable or disable the **Status**. The **Mode** field is displayed.
   b) Click the **Monitor** mode or **Enforced** mode radio option. If you enable the **Enforced** mode, use the slider to enable or disable **Optimization**.
   c) Use the slider to enable to disable **Bridge Client Access**. This is applicable for mesh mode APs. Bridge Client Access determines the percentage of the ATF policy weight that is allocated to clients connected to the mesh APs.
5. Specify the **Airtime Allocation** value between 5 and 90.
6. Click **Apply to Device**.

Enable ATF in the RF profile (CLI)

Configure ATF in the RF profile using commands.

Enable Cisco ATF on 2.4 GHz or 5 GHz radios to manage air time allocation and improve network performance.

Cisco ATF must be enabled on 2.4 GHz or 5 GHz radios separately. Use this procedure when you need to configure air time fairness policies to ensure optimal wireless network performance and client experience.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an RF profile for 2.4- or 5-GHz radio.

   Example

   ```
   Device(config)# ap dot11 frequency-band rf-profile rf-profile-name
   ```

   Example

   ```
   Device(config)# ap dot11 24ghz rf-profile rfprof24_1
   ```

   Where *frequency-band* is either 24ghz or 5ghz.

3. Configure air time fairness mode.

   Example

   ```
   Device(config-rf-profile)# airtime-fairness mode mode-type
   ```

Example

```
Device(config-rf-profile)# airtime-fairness mode enforce-policy
```

Configures air time fairness in either of the modes:

- Enforce-policy–This mode signifies that the ATF is operational.
- Monitor–This mode gathers information about air time and reports air time usage.

4. Enable the air time fairness optimization.

Example

```
Device(config-rf-profile)# airtime-fairness optimization
```

Optimization is effective when the current WLAN reaches the air time limit and the other available WLANs does not use air time to its full extent.

5. Return to privileged EXEC mode.

Example

```
Device(config-rf-profile)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

ATF is now enabled in the RF profile and will manage air time allocation according to the configured policy mode.

Cisco ATF configuration verification

Lists commands to verify Cisco ATF configurations.

You can verify Cisco ATF configurations using these commands:

Table 157: Commands for verifying Cisco ATF configurations

Commands	Description
show wireless profile airtime-fairness summary	Displays the summary of air time fairness profiles.
show wireless profile airtime-fairness mapping	Displays the ATF policy mapping with the wireless profiles.
show ap airtime-fairness summary	Displays the ATF configuration summary of all radios.
show ap dot11 24ghz airtime-fairness	Displays the ATF configuration for 2.4-GHz radio.
show ap dot11 5ghz airtime-fairness	Displays the ATF configuration for 5-GHz radio.
show ap name *ap-name* **airtime-fairness**	Displays the ATF configuration or statistics for an AP.
show ap name *ap-name* **dot11 {24ghz | 5ghz} airtime-fairness statistics summary**	Displays the ATF statistics of 2.4- or 5GHz radio.

Cisco ATF statistics verification

Lists the commands available to verify Cisco ATF statistics.

Use these commands to verify Cisco ATF statistics for WLANs, ATF policies, and individual clients.

Table 158: ATF statistics per WLAN

Commands	Description
show ap name *ap-name* **dot11** {**24ghz** | **5ghz**} **airtime-fairness wlan** *wlan_name* **statistics**	Displays the ATF statistics related to a WLAN.

Table 159: ATF statistics per ATF policy

Commands	Description
show ap name *ap-name* **dot11** {**24ghz** | **5ghz**} **airtime-fairness policy** *policy-name* **statistics**	Displays the ATF statistics related to an ATF policy.

Table 160: ATF statistics per client

Commands	Description
show ap airtime-fairness statistics client *mac_address*	Displays the ATF statistics related to a client.

178 IPv6 Non-AVC QoS Support

Topics:

- IPv6 non-AVC QoS support

This chapter explains configuring IPv6 QoS on access points, including DSCP marking, packet dropping, and policing for Fabric and FlexConnect local switching, with CLI procedures and verification commands.

IPv6 non-AVC QoS support

IPv6 non-AVC QoS support enables Quality of Service (QoS) for IPv6 traffic on Fabric and FlexConnect local switching and offers the same functionality as IPv4 QoS starting from Cisco IOS XE 17.2.1.

IPv6 non-AVC QoS support is a wireless network feature that

- enables QoS for IPv6 traffic on Fabric and FlexConnect local switching,
- performs QoS processing at the access point (AP), and
- offers the same functionality as IPv4 QoS starting from Cisco IOS XE 17.2.1.

Supported actions for IPv6 non-AVC QoS

You can use IPv6 non-AVC QoS to:

- mark the Differentiated Services Code Point (DSCP) value for IPv6 packets,
- drop IPv6 packets based on the DSCP value,
- police IPv6 traffic.

Platform limitations

IPv6 non-AVC QoS is not supported on Cisco Aironet 1700, 2700, or 3700 Series Access Points.

Mark DSCP values in an IPv6 packet (CLI)

Learn how to mark DSCP values for an IPv6 packet using the CLI by configuring a policy map and setting the DSCP value in the packet.

Assign differentiated services code point (DSCP) values to IPv6 packets for traffic classification and Quality of Service (QoS).

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a policy map.

   Example

   ```
   Device(config)# policy-map policy-map-name
   ```

3. Specify the class map criteria for the policy.

   Example

   ```
   Device(config-pmap)#class class-map-name
   ```

4. Set the DSCP value in an IPv6 packet.

   Example

   ```
   Device(config-pmap-c)#set dscp <0-63>
   ```

   DSCP values must be in the range of 0 to 63.

The selected DSCP value is assigned to matching IPv6 packets, enabling traffic classification and QoS handling according to your policy.

Drop an IPv6 packet with DSCP values (CLI)

Learn how to drop an IPv6 packet with DSCP values using CLI commands.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a policy map.

   Example

   ```
   Device(config)# policy-map policy-map-name
   ```

3. Create a policy criteria.

   Example

   ```
   Device(config-pmap)#class class-map-name
   ```

4. Police the committed information rate.

   Example

   ```
   Device(config-pmap-c)#police cir <8000 - 10000000000>
   ```

   the committed information rate is between 8000 and 10000000000. Target bit rate (Bits per second).

5. Configure the **conform-action drop** command.

   Example

   ```
   Device(config-pmap-c-police)#conform action drop
   ```

   This command is the action when the rate is less than the conform burst.

6. Configure the **exceed-action drop** command.

Example

```
Device(config-pmap-c-police)#exceed-action drop
```

This command is the action when the rate is within the conform and conform plus exceed burst.

Configure policing for IPv6 traffic (CLI)

Learn how to police IPv6 traffic by configuring a policy map, class map, and policing actions on your device.

Police IPv6 traffic by enforcing bandwidth limits and actions for conforming and exceeding packets.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a policy map.

   Example

   ```
   Device(config)# policy-map policy-map-name
   ```

3. Create a policy criteria.

   Example

   ```
   Device(config-pmap)#class class-map-name
   ```

4. Specify the committed information rate.

   Example

   ```
   Device(config-pmap-c)#police cir <8000 - 10000000000>
   ```

   The CIR (committed information rate) sets the allowed bits per second; choose a value between 8000 and 10,000,000,000.

5. Configure the **conform-action transmit** command, for transmitting packets.

   Example

   ```
   Device(config-pmap-c-police)#conform-action transmit
   ```

6. Configure the **exceed-action drop** command.

   Example

   ```
   Device(config-pmap-c-police)#exceed-action drop
   ```

This command specifies the action taken when the packet rate exceeds the CIR but is within conform plus exceed burst.

IPv6 traffic is policed according to your defined bandwidth and actions. The device transmits packets within the CIR and drops packets that exceed the specified rate.

IPv6 non-AVC QoS verification

Verify IPv6 non-AVC QoS.

- To verify the DSCP values for IPv6 packets, IPv6 packets that are dropped, and the policing of IPv6 traffic, use the **show policy-map** command:

  This is a sample output of the **show** command that verifies the DSCP value for an IPv6 packet:

```
Device# show policy-map
1 policymaps
Policy Map Set-dscp type:qos client:default
 Class Set-dscp1_ADV_UI_CLASS
 set dscp af41 (34)
 Class class-default
 no actions
```

- This is a sample output of the **show** command that verifies the IPv6 packets that are dropped:

```
Device# show policy-map
1 policymaps
Policy Map Drop-dscp type:qos client:default
        Class Drop-dscp1_ADV_UI_CLASS
        drop

        Class class-default
        no actions
```

- This is a sample output of the **show** command that verifies the policing of IPv6 traffic:

```
Device# show policy-map
1 policymaps
Policy Map Drop-traffic type:qos client:default
        Class Drop-traffic1_ADV_UI_CLASS
        police rate 2000000 bps (250000Bytes/s)
        conform-action
        exceed-action

        Class class-default
        no actions
```

179 QoS Basic Service Set Load

Topics:

- QoS basic set service load

This chapter explains QBSS IE behavior across Cisco IOS XE releases and provides CLI steps to enable or disable QBSS load and WMM on WLANs, configure policy profiles and VLANs, and verify QBSS load status.

QoS basic set service load

Explains the per-WLAN configuration knob that controls QBSS IE advertisement in beacon frames and probe responses.

A QoS Basic Set Service (QBSS) information element (IE) knob is a per-WLAN configuration feature that

- controls the inclusion or exclusion of the QBSS IE in beacon frames and probe responses
- advertises channel load information of an AP through the QBSS IE, and
- is enabled by default to provide QoS load information to wireless clients.

QBSS load behavior changes

The relationship between WMM and QBSS load advertisement has evolved across Cisco IOS XE releases.

Until Cisco IOS XE 17.1.1s, the enablement of Wi-Fi Multimedia (WMM) automatically enabled the QBSS load advertisement in the probes and beacons and there was no separate knob to turn on QBSS load IE. However, from Cisco IOS XE 17.2.1, this behavior has changed with the introduction of a separate configuration knob.

The behavior until Cisco IOS XE 17.1.1s:

- When WMM was enabled on WLAN, QBSS load was advertised in the beacon and probe frames.
- When WMM was disabled on WLAN, QBSS IE was not advertised in the beacon and probe frames.

The behavior from Cisco IOS XE 17.2.1:

- When you enable WMM and QBSS ID on WLAN, QBSS IE is advertised in the beacon and probe frames.
- When you enable WMM on WLAN and disable QBSS load IE on WLAN, QBSS load is not advertised in the beacon and probe frames.
- When you disable WMM on WLAN and enable QBSS load IE on WLAN, QBSS IE is advertised in the beacon and probe frames.

> **Note**
>
> By default, QBSS load IE is enabled. The behavior can be configured on policy profile.

Configure Wi-Fi multimedia (CLI)

Configure Wi-Fi multimedia on a WLAN to enable quality of service features for multimedia traffic.

Enable Wi-Fi Multimedia (WMM) on a WLAN to provide quality of service features for multimedia applications.

Create a WLAN and then enable WMM to prioritize multimedia traffic and improve network performance for applications that require specific bandwidth or latency requirements.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Specify the WLAN name and ID.

Example

```
Device(config)# wlan profile-name wlan-id ssid
```

Example

```
Device(config)# wlan mywlan 34 mywlan-ssid
```

Where:

- *profile-name*: Profile name of the WLAN. You can use between 1 to 32 alphanumeric characters.
- *wlan-id*: WLAN ID. You can use between 1 to 512 alphanumeric characters.
- *ssid*: Service Set Identifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID.

> **Note**
>
> By default, the WLAN is disabled.

3. Disable WPA2 ciphers for Advanced Encryption Standard (AES).

Example

```
Device(config-wlan)# no security wpa wpa2 ciphers aes
```

4. Disable security AKM for dot1x.

Example

```
Device(config-wlan)# no security wpa akm dot1x
```

5. Configure WMM and allow WMM on the WLAN.

Example

```
Device(config-wlan)# wmm allowed
```

6. Enable WLAN.

Example

```
Device(config-wlan)# no shutdown
```

WMM is now enabled on the WLAN, providing quality of service features for multimedia traffic prioritization.

Enable QoS basic set service load (CLI)

Configure QoS basic set service load to provide enhanced QoS information element functionality on the WLAN.

Enable QBSS load to provide QoS enhanced basic service set information element functionality on the WLAN.

QBSS load enables the transmission of enhanced QoS information elements that help optimize wireless network performance. Perform this procedure to enable QBSS load on a wireless policy profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy profile-policy
   ```

   Example

   ```
   Device(config)# wireless profile policy rr-xyz-policy-1
   ```

3. Configure VLAN name or VLAN ID.

   Example

   ```
   Device(config-wireless-policy)# vlan vlan-id
   ```

   Example

   ```
   Device(config-wireless-policy)# vlan 24
   ```

4. Enable QoS enhanced basic service set information element.

   Example

   ```
   Device(config-wireless-policy)# qbss-load
   ```

   Use the **no** form of this command to disable the feature.

5. Save the configuration and exit configuration mode.

Example

```
Device(config-wireless-policy)# no shutdown
```

QoS basic set service load is now enabled on the wireless policy profile. The configuration is saved and you have returned to privileged EXEC mode.

What to do next

1. Create a policy tag. For more information about creating policy tags, refer to *Configuring a Policy Tag (CLI)*.
2. Map the policy tag to the AP. For more information about mapping a policy tag to the AP, refer to *Attaching a Policy Tag and Site Tag to an AP (CLI)*.

QoS basic set service load verification

Verify QoS basic set service load.

To verify if QBSS load is enabled, use the **show wireless profile policy detailed** *named-policy-profile* command:

```
Device# show wireless profile policy detailed named-policy-profileshow wireless
profile policy detailed named-policy-profile
Policy Profile Name                 : named-policy-profile
Description                         :
Status                              : ENABLED
VLAN                                : 91
Multicast VLAN                      : 0
OSEN client VLAN                    :
Multicast Filter                    : DISABLED
QBSS Load                           : ENABLED
Passive Client                      : DISABLED
ET-Analytics                        : DISABLED
StaticIP Mobility                   : DISABLED
WLAN Switching Policy
  Flex Central Switching            : ENABLED
  Flex Central Authentication       : ENABLED
  Flex Central DHCP                 : ENABLED
  Flex NAT PAT                      : DISABLED
  Flex Central Assoc                : ENABLED
```

Part XIV

IPv6

Topics:

- IPv6 Client IP Address Learning
- IPv6 ACL
- Configuring IPv6 Mobility
- IPv6 Support on Flex and Mesh
- IPv6 CAPWAP UDP Lite Support
- Neighbor Discovery Proxy
- Address Resolution Protocol Proxy
- IPv6 Ready Certification

180 IPv6 Client IP Address Learning

Topics:

- IPv6 client address learning
- IPv6 address tracking for wireless clients
- Native IPv6

This chapter explains how to discover, track, and manage IPv6 addresses for wireless clients, and provides configuration steps (CLI and GUI) for SLAAC, DHCPv6, RA guard/throttle, ND suppression, and related IPv6 interface and DHCP settings.

IPv6 client address learning

Explains how wireless controllers learn IPv6 client addresses and the mechanisms clients use to acquire them.

A client address learning mechanism is a wireless controller feature that

- identifies and stores IPv4 and IPv6 addresses for wireless clients
- maintains the client's transition state during association and timeout, and
- operates through monitoring relevant network protocols.

There are three ways for an IPv6 client to acquire IPv6 addresses:

- Stateless Address Auto-Configuration (SLAAC)
- Stateful DHCPv6
- Static Configuration

In all of these methods, the IPv6 client always sends a Neighbor Solicitation Duplicate Address Detection (DAD) request to ensure that there is no duplicate IP address on the network. The device monitors Neighbor Discovery Protocol (NDP) and DHCPv6 packets from the client to learn its IP addresses.

Prerequisites for IPv6 client address learning

This topic outlines the requirements you must meet before configuring IPv6 client address learning

- Before configuring IPv6 client address learning, configure the clients to support IPv6.
- To enable wireless IPv6 client connectivity, the underlying wired network must support IPv6 routing and an address assignment mechanism, such as SLAAC or DHCPv6. The wireless LAN controller must have L2 adjacency to the IPv6 router.

> **Note**
>
> The AP learns the IPv6 client address based on the source IP address, even though neighbor advertisements can hold the remaining IPv6 addresses. The AP does not examine neighbor advertisements to learn the IPv6 address acquired by the client. This behavior occurs only with Apple clients and not with Microsoft Windows clients.

Configure IPv6 interface on a switch (GUI)

Configure an IPv6 interface on a switch by specifying VLAN, interface details, and IPv6 settings using the GUI.

Enable IPv6 connectivity on a switch interface by assigning IPv6 addresses and configure related settings via the GUI.

Procedure

1. Choose **Configuration** > **Layer2** > **VLAN** > **SVI**.
2. Click **Add**.
3. Enter **VLAN Number**, **Description** and **MTU (Bytes)**.
4. Enable or disable the **Admin Status** toggle button.
5. In **IP Options**, check the **IPv6** check box.

6. Select the type of **Static** address from the drop-down list and enter the static address.
7. Check or uncheck the **DHCP**, **Autoconfig** and **Act as an IPv6 DHCP client** check boxes.

   If you check the **DHCP** check box, the **Rapid Commit** check box is displayed. Check or uncheck the **Rapid Commit** check box.
8. Click **Apply to Device**.

Configure IPv6 on Interface (CLI)

Use this procedure to configure IPv6 addresses on a interface using the commands.

Use this procedure to assign IPv6 addresses to a specific controller interface using commands.

Before you begin

Enable IPv6 on the client and IPv6 support on the wired infrastructure.

Procedure

1. Enable the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password, if prompted.
2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
3. Create an interface and enter the interface configuration mode.

   Example

   ```
   Device(config)# interface vlan vlan-id
   ```
4. Configure IPv6 address on the GigabitEthernet interface using the link-local option.

   Example

   ```
   Device(config-if)# ip address 198.51.100.1 255.255.255.0
   Device(config-if)# ipv6 address fe80::1 link-local
   Device(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64
   Device(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64
   ```
5. Optional: Enable IPv6 on the GigabitEthernet interface.

   Example

   ```
   Device(config)# ipv6 enable
   ```

6. Exit interface mode.

Example

```
Device(config)# end
```

Address assignment using SLAAC

This topic explains how IPv6 address assignment works using SLAAC, including algorithm options and router configuration..

A SLAAC address assignment is an IPv6 address configuration mechanism that

- enables clients to automatically generate their own addresses based on a network prefix
- provides plug-and-play connectivity without requiring manual configuration, and
- utilizes router advertisements to communicate the IPv6 prefix and default gateway to clients.

SLAAC is configured as follows:

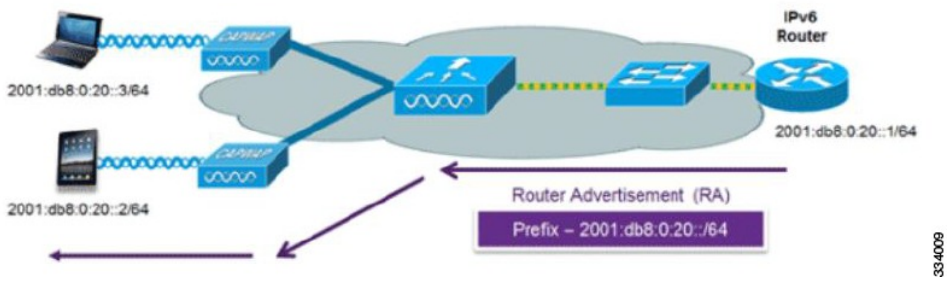
- A host sends a Router Solicitation message.
- The host waits for a Router Advertisement message.
- The host takes the first 64 bits of the IPv6 prefix from the Router Advertisement message and combines it with the 64-bit EUI-64 address (in the case of Ethernet, this is created from the MAC address) to create a global unicast address. The host also uses the source IP address in the IP header of the Router Advertisement message as its default gateway.
- IPv6 clients perform Duplicate Address Detection to ensure that randomly chosen addresses do not conflict with those of other clients.

Note

The last 64 bits of the IPv6 address can be learned by using one of these algorithms:

- EUI-64, which is based on the MAC address of the interface
- Private addresses that are randomly generated

Figure 31: Address Assignment Using SLAAC



These Cisco IOS configuration commands from a Cisco-capable IPv6 router are used to enable SLAAC addressing and router advertisements:

```
ipv6 unicast-routing
    interface Vlan20
    description IPv6-SLAAC
    ip address 192.168.20.1 255.255.255.0
    ipv6 address FE80:DB8:0:20::1 linklocal
    ipv6 address 2001:DB8:0:20::1/64
    ipv6 enable
    end
```

Stateful DHCPv6 address assignment

This topic explains how stateful DHCPv6 assigns IPv6 addresses to clients and its operational differences compared to stateless DHCPv6.

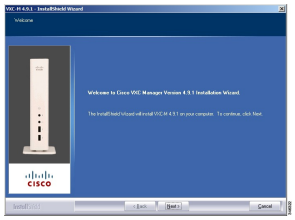
A stateful DHCPv6 address assignment is an IPv6 address management method that

- assigns unique IPv6 addresses to clients from a DHCPv6 pool
- enables the router or external DHCPv6 server to manage address state and lease information, and
- provides additional configuration details, including domain names and DNS servers.

There are two modes of operation for DHCPv6: Stateless and Stateful.

The DHCPv6 Stateless mode is used to provide clients with additional network information that is not available in the router advertisement, but not an IPv6 address, becuase this is already provided by SLAAC. This information includes the DNS domain name, DNS servers, and other DHCP vendor-specific options.

Figure 32: Stateful DHCPv6 Address Assignment



Configure the interface as shown to implement stateless DHCPv6 with SLAAC enabled on your Cisco IOS IPv6 router.

```
ipv6 unicast-routing
ipv6 dhcp pool IPV6_DHCPPOOL
address prefix 2001:db8:5:10::/64
domain-name cisco.com
dns-server 2001:db8:6:6::1
interface Vlan20
description IPv6-DHCP-Stateless
ip address 192.168.20.1 255.255.255.0
ipv6 nd other-config-flag
ipv6 dhcp server IPV6_DHCPPOOL
ipv6 address 2001:DB8:0:20::1/64
end
```

Configure DHCP pool on switch (GUI)

Configure a DHCP pool on a switch by specifying IP address ranges, subnet mask, lease times, and other related settings using the GUI.

Set up DHCP pools to automatically assign IP addresses to devices on your network using the GUI.

Procedure

1. Choose **Administration** > **DHCP**. Click the **Add** button. The **Create DHCP Pool** dialog box appears.
2. Enter a pool name in the **DHCP Pool Name** field. The pool name must not exceed 236 characters.
3. Select either **IPv4** or **IPv6** from the **IP Type** drop-down list.
4. Enter an IP address in the **Network** field.
5. Select a subnet mask from the **Subnet Mask** drop-down list.
6. Enter an IP address in the **Starting ip** field amd enter an IP address in the **Ending ip** field.
7. Optional: Set the status of the **Reserved Only** field to **Enabled** to reserve the DHCP pool. Select the desired option from the **Lease** drop-down list.
8. Selecting the **User Defined** option from the **Lease** drop-down list enables the **(0-365 days)**, **(0-23 hours)**, and **(0-59 minutes)** fields. Enter appropriate values. Click the **Save & Apply to Device** button.
9. For IPv6, enter the **DNS Server, DNS Domain Name, and Ipv6 Address Allocation**.

Configure DHCP pool on Switch (CLI)

Configure a DHCP pool on a switch using commands to enable automatic IP address assignment to connected devices.

Set up a DHCP pool so that devices connected to the switch automatically receive IP addresses and network configuration using commands.

Follow the procedure given below to configure DHCP Pool on an interface:

Procedure

1. Enable the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Enter the configuration mode and configure the IPv6 DHCP pool on the VLAN.

   Example

   ```
   Device(config)# ipv6 dhcp pool vlan-id
   ```

4. Enter the configuration-DHCP mode and configure the address pool and its lifetime on a VLAN.

Example

```
Device(config-dhcpv6)# address prefix 2001:DB8:0:1:FFFF:1234::/64 lifetime
300 10
```

5. Configure the DNS servers for the DHCP pool.

   Example

   ```
   Device(config-dhcpv6)# dns-server 2001:20:21::1
   ```

6. Configure the domain name to complete unqualified host names.

   Example

   ```
   Device(config-dhcpv6)# domain-name example.com
   ```

7. Return to the privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   Alternatively, you can also press ***Ctrl-Z*** to exit global configuration mode.

Configure stateless auto address configuration without DHCP on Switch (CLI)

Use this task to configure IPv6 stateless address autoconfiguration (SLAAC) on a switch interface without using DHCP.

Configure IPv6 addresses using stateless auto address configuration without enabling DHCP on the switch using commands.

Procedure

---

1. Enable the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Create an interface and enter the interface configuration mode.

   Example

   ```
   Device# interface vlan 1
   ```

4. Configure IPv6 address on the GigabitEthernet interface using the link-local option.

   Example

   ```
   Device(config-if)# ip address 198.51.100.1 255.255.255.0
   Device(config-if)# ipv6 address fe80::1 link-local
   Device(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64
   Device(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64
   ```

5. Optional: Enable IPv6 on the GigabitEthernet interface.

   Example

   ```
   Device(config)# ipv6 enable
   ```

6. Ensure the attached hosts do not use stateful autoconfiguration to obtain addresses.

   Example

   ```
   Device(config)# interface vlan 1
   Device(config-if)# no ipv6 nd managed-config-flag
   ```

7. Ensure the attached hosts do not use stateful autoconfiguration to obtain non-address options from DHCP (domain etc).

   Example

   ```
   Device(config-if)# no ipv6 nd other-config-flag
   ```

8. Return to the privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   Alternatively, you can also press ***Ctrl-Z*** to exit global configuration mode.

Configure stateless auto address configuration with DHCP on Switch

Configure IPv6 stateless auto addressing with DHCP on a switch using the commands.

Enable IPv6 stateless auto address configuration so attached hosts can receive IPv6 addresses and additional information from DHCP without stateful address assignment using commands.

Procedure

---

1. Enable the privileged EXEC mode and enter the global configuration mode.

Example

```
Device# enable
```

```
Device# configure terminal
```

Enter your password if prompted.

2. Create an interface and enter the interface configuration mode.

Example

```
Device(config)# interface vlan 1
```

3. Configure IPv6 address on the GigabitEthernet interface using the link-local option.

Example

```
Device(config-if)# ip address 198.51.100.1 255.255.255.0
Device(config-if)# ipv6 address fe80::1 link-local
Device(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64
Devicec(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64
```

4. Configure IPv6 address on the GigabitEthernet interface using the link-local option.

Example

```
Device(config-if)# ip address 198.51.100.1 255.255.255.0
Device(config-if)# ipv6 address fe80::1 link-local
Device(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64
Device(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64
```

5. Optional: Enable IPv6 on the GigabitEthernet interface.

Example

```
Device(config)# ipv6 enable
```

6. Specify a subnet prefix.

Example

```
Device(config)# ipv6 nd prefix 2001:9:3:54::/64 no-advertise
```

7. Ensure the attached hosts do not use stateful autoconfiguration to obtain addresses.

Example

```
Device(config)# interface vlan 1
Device(config-if)# no ipv6 nd managed-config-flag
```

8. Ensure the attached hosts do not use stateful autoconfiguration to obtain non-address options from DHCP (domain etc).

Example

```
Device(config-if)# no ipv6 nd other-config-flag
```

9. Display the configuration parameters and exit the interface mode.

Example

```
Device(config)# ipv6 dhcp server servername
```

```
Device(config)# end
```

Configure stateless address auto configuration without DHCP on Switch (CLI)

Configure IPv6 stateless address autoconfiguration (SLAAC) without DHCP for hosts on a switch using commands.

Set up IPv6 addresses for hosts on a switch using SLAAC, ensuring they do not rely on DHCP for address or configuration using commands.

Procedure

---

1. Enable the privileged EXEC mode.

Example

```
Device# enable
```

Enter your password if prompted.

2. Enter the global configuration mode.

Example

```
Device# configure terminal
```

3. Create an interface and enter the interface configuration mode.

Example

```
Device(config)# interface vlan 1
```

4. Configure IPv6 address on the GigabitEthernet interface using the link-local option.

Example

```
Device(config-if)# ip address 198.51.100.1 255.255.255.0
Device(config-if)# ipv6 address fe80::1 link-local
Device(config-if)# ipv6 address 2001:DB8:0:1:FFFF:1234::5/64
Device(config-if)# ipv6 address 2001:DB8:0:0:E000::F/64
```

5. Optional: Enable IPv6 on the GigabitEthernet interface.

Example

```
Device(config)# ipv6 enable
```

6. Ensure the attached hosts do not use stateful autoconfiguration to obtain addresses.

   Example

   ```
   Device(config)# interface vlan 1
   Device(config-if)# no ipv6 nd managed-config-flag
   ```

7. Ensure the attached hosts do not use stateful autoconfiguration to obtain non-address options from DHCP (domain etc).

   Example

   ```
   Device(config-if)# no ipv6 nd other-config-flag
   ```

8. Return to the privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   Alternatively, you can also press ***Ctrl-Z*** to exit global configuration mode.

Router solicitation

This topic explains how router solicitation messages prompt routers to advertise configuration information for local routing and address auto-configuration.

A router solicitation message is a network communication signal that

- is issued by a host controller to locate local routers
- prompts routers to transmit Router Advertisement messages, and
- facilitates the host in obtaining routing information or performing stateless auto-configuration.

Router Advertisements are transmitted periodically. The host triggers an immediate Router Advertisement using a Router Solicitation, for example, when it boots or after a restart operation.

Router advertisement

A Router Advertisement message is issued periodically by a router or in response to a Router Solicitation message from a host.

A Router advertisement is a type of message

- that is sent by IPv6 routers either periodically or in response to Router Solicitation messages from hosts, and
- communicates essential information such as network prefixes, default gateway availability, and configuration flags.

Router advertisement messages enable hosts to perform stateless autoconfiguration and update their routing tables with current network settings.

Router advertisement guard

This topic explains how router advertisement guards increase IPv6 network security by blocking rogue router advertisements from wireless clients.

A router advertisement guard is a security feature that

- examines incoming router advertisement messages from IPv6 devices
- drops unwanted or rogue router advertisements coming from wireless clients, and
- protects legitimate IPv6 routers and the network from misconfigured or malicious IPv6 clients.

By default, RA guard is always enabled on the controller.

- Port on which the frame is received
- IPv6 source address
- Prefix list
- Trusted or Untrusted ports for receiving the router advertisement guard messages
- Trusted/Untrusted IPv6 source addresses of the router advertisement sender
- Trusted/Untrusted Prefix list and Prefix ranges
- Router preference

Router advertisement throttling

This topic explains how router advertisement throttling limits the frequency of RA packets and maintains IPv6 connectivity for clients.

A router advertisement throttle is a network control mechanism that

- enforces limits on the frequency of router advertisement (RA) packets sent toward the wireless network
- ensures routers that send multiple RA packets are restricted to a minimum frequency required for IPv6 client connectivity, and
- maintains uninterrupted IPv6 connectivity for new or roaming clients by allowing RA packets in response to Router Solicitation (RS) packets.

If a client sends an RS packet, the router sends an RA back to the client. The controller allows this RA to pass and delivers it as a unicast message to the client. This process ensures that new or roaming clients are not affected by RA throttling.

Configure RA throttle policy (CLI)

Configure Router Advertisement (RA) throttle policies to enforce limits for IPv6 multicast RA traffic to wireless clients using commands

Restrict the frequency and quantity of multicast Router Advertisements (RAs) sent to wireless clients for improved network performance and client behavior using commands.

Configure RA Throttle policy to allow the enforce the limits

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Define the router advertisement (RA) throttler policy name and enter IPv6 RA throttle policy configuration mode.

   Example

   ```
   Device(config)# ipv6 nd ra-throttler policy ra-throttler1
   ```

3. Configure the throttle period in an IPv6 RA throttler policy.

   Example

   ```
   Device(config-nd-ra-throttle)# throttle-period 500
   ```

   The throttle period is measured in seconds. It is the duration during which the controller does not forward RA to wireless clients.

4. Limit multicast RAs per VLAN per throttle period.

   Example

   ```
   Device(config-nd-ra-throttle)# max-through 15
   ```

5. Limit the number of multicast RAs per device per throttle period in an RA throttler policy.

   Example

   ```
   Deviceconfig-nd-ra-throttle)# allow at-least 5 at-most 10
   ```

Apply RA throttle policy on VLAN (GUI)

Apply a Router Advertisement (RA) throttle policy to a VLAN through the GUI to control the rate of RA packets on your network.

Apply a Router Advertisement (RA) throttle policy to a VLAN through the GUI to control the rate of RA packets on your network.

Procedure

---

1. Choose **Configuration** > **Services** > **RA Throttle Policy**.
2. Click **Add**. The **Add RA Throttle Policy** dialog box appears.
3. Enter a name for the policy in the **Name** field.
4. Select the desired option from the **Medium Type** drop-down list.
5. Enter a value in the **Throttle Period** field. RA throttling takes place only after the **Max Through** limit is reached for the VLAN or the **Allow At-Most** value is reached for a particular router.
6. Enter a value for the **Max Through** field, which is the maximum number of RA packets on a VLAN that can be sent before throttling takes place. The **No Limit** option allows an unlimited number of RA packets through with no throttling.
7. Choose an **Interval Option**. The device acts differently based on the RFC 3775 value set in IPv6 RA packets. Options include:

- Ignore: Causes the RA throttle to treat packets with the interval option as a regular RA and subjects them to throttling if in effect.
- Passthrough: Allows any RA messages with the RFC 3775 interval option to go through without throttling.
- Throttle: Causes the RA packets with the interval option to always be subject to rate limiting.

8. Enter the minimum number of RA packets per router that can be sent as multicast before throttling takes place in the **At Least Multicast RAs** field.
9. Enter the maximum number of RA packets per router that can be sent as multicast before throttling takes place in the **At Most Multicast RAs** field. The **No Limit** option allows an unlimited number of RA packets through the router. Click the **Add & Apply to Device** button.

Apply RA throttle policy on a VLAN (CLI)

Configure an RA throttle policy for a VLAN using commands to limit Router Advertisement (RA) packet frequency.

Control and optimize IPv6 Router Advertisement (RA) packet frequency within a VLAN, ensuring efficient IPv6 client connectivity using commands.

Applying the RA Throttle policy on a VLAN. Enabling RA throttling reduces the frequency of router advertisement packets while still maintaining IPv6 client connectivity.

Procedure

---

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a VLAN or a collection of VLANs and enter the VLAN configuration mode.

   Example

   ```
   Device(config)# vlan configuration 1
   ```

3. Attach an IPv6 RA throttler policy to a VLAN or a collection of VLANs.

   Example

   ```
   Device(config-vlan)# ipv6 nd ra throttler attach-policy ra-throttler1
   ```

Neighbor discovery

IPv6 Neighbor Discovery is a protocol that enables IPv6 nodes to discover other nodes on the local network, determine their link-layer addresses, and maintain reachability information.

IPv6 Neighbor Discovery (ND) is a suite of messages and processes

- that allows devices on an IPv6 network to discover neighbors, determine their link-layer addresses, identify routers, and

- maintain information about routes and reachability replacing several IPv4 mechanisms, including the Address Resolution Protocol (ARP), ICMP Router Discovery, and ICMP Redirect.

IPv6 neighbor discovery inspection analyzes neighbor discovery messages to build a trusted binding table database. IPv6 Neighbor Discovery packets that do not comply with requirements are dropped. The neighbor binding table in the tracks each IPv6 address and its associated MAC address. The table removes clients according to the neighbor-binding timers.

Neighbor discovery suppression

This topic explains how neighbor discovery suppression improves wireless device efficiency by selectively responding to multicast requests.

A neighbor discovery suppression mechanism is a network optimization feature that

- caches IPv6 addresses of wireless clients when they reach RUN state
- selectively converts multicast Neighbor Solicitation (NS) requests to unicast or replies directly, depending on configuration, and
- forwards NS multicast requests to the wired side when the target address is not in the device's cache.

The same behavior applies to ARP requests for IPv4 addresses, where the device maintains the IPv4 address of the wireless client in the cache.

If neither configuration is enabled, and the device receives a Non-DAD or DAD NS multicast searching for an IPv6 address, and the target address is known to the device and belongs to one of its clients, the device converts the multicast NS to a unicast NS. It replaces the destination MAC address with the client's MAC address and forwards the unicast packet to the client.

If full proxy is enabled, and the device receives a Non-DAD or DAD NS multicast searching for an IPv6 address, and the target address is known to the device and belongs to one of its clients, the device replies with an NA message on behalf of the client.

Use the **ipv6 nd proxy** command to enable or disable DAD or full proxy.

When the device receives an DAD-NS multicast looking for an IPv6 address, and if the target address is known to the device and belongs to one of its clients, the device will reply with an NA message on behalf of the client.

If the device receives a DAD-NS multicast searching for an IPv6 address, and the target address is known to the device and belongs to one of its clients, the device replies with an NA message on behalf of the client.

If the device does not have the IPv6 address of a wireless client, it does not respond with NA; instead, it forwards the NS packet to the wired side. This is because all wireless client IPv6 addresses and their corresponding MAC addresses should be available in the controller. If an IPv6 address required by the NS is not available, the address is assumed not to belong to a wireless client, so the request is forwarded to the wired side.

IPv6 address tracking for wireless clients

Explains how Cisco wireless controllers track IPv6 addresses for wireless clients, the operational limits, and the impact of new software releases on address tracking and feature compatibility.

A wireless client's IPv6 addresses are network configuration parameters that

- enable a controller to identify and monitor multiple IPv6 addresses used by a single wireless device
- are stored and tracked by the controller to enforce policy and facilitate client management, and
- are subject to a fixed limit that influences feature availability and network behavior.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 161: Feature history for IPv6 Address Tracking for Wireless Clients

Feature Name	Release Information	Feature Description
IPv6 Address Tracking for Wireless Clients	Cisco IOS XE Cupertino 17.9.2	In this release, the controller allows data traffic from wireless clients with new IPv6 source addresses, even after eight addresses have been learned for each client. The controller continues to learn new IPv6 addresses from the clients' control traffic (IPv6 NS/NA and DHCPv6) and keeps track of only a maximum of eight addresses per wireless client.
IPv6 Address Tracking for Wireless Clients	Cisco IOS XE Cupertino 17.9.1	Until this release, the controller supported a maximum of eight IPv6 addresses per wireless client. After eight IPv6 addresses were learned for a wireless client, the controller dropped any data traffic from that client with new IPv6 source addresses.

To forward multicast neighbor solicitation (NS) queries for unknown IPv6 target addresses those not tracked by the controller to wireless clients, you must enable the **wireless ipv6 nd ns-forward** configuration.

Important

Configure IPv6 Multicast over Multicast (MoM) tunnel along with the **wireless ipv6 nd ns-forward** configuration.

Note

In Cisco IOS XE 17.9.2, because the controller allows IPv6 traffic without address tracking beyond the eight-address limit, some features such as User Defined Network, iPSK Peer-to-Peer Blocking, Management over Wireless, Neighbor Discovery Suppression, and IP Theft Detection may not work for clients using more than eight addresses. You can disable the new behavior by enabling the IP Source Guard feature https://www.cisco.com/c/en/us/td/docs/wireless/controller/9800/17-9/config-guide/b_wl_17_9_cg/m_ipsg_ewlc.html?bookSearch=true.

Configure unknown address multicast neighbor solicitation forwarding (CLI)

Configure your wireless controller to forward multicast neighbor solicitation (NS) messages for unknown IPv6 client addresses.

Enable the controller to forward multicast NS queries for wireless clients whose IPv6 addresses are not tracked, ensuring proper neighbor discovery on the network using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable forwarding of the multicast neighbor solicitation (NS) messages for unknown IPv6 target addresses, specifically, the addresses of wireless clients that are not tracked by the controller to wireless clients.

   Example

   ```
   Device (config)# wireless ipv6 nd ns-forward
   ```

   Note

   We recommend configuring IPv6 Multicast over Multicast (MoM) tunnel together with this command.

Native IPv6

Enables seamless communication by using IPv6 as the exclusive protocol throughout a network, eliminating the need for IPv4 translation or tunneling mechanisms.

IPv6

IPv6 is a packet-based protocol used to exchange data, voice, and video traffic over digital networks.

IPv6 (Internet Protocol version 6) is a

- packet-based communications protocol developed to exchange data, voice, and video traffic over digital networks, and
- successor to IPv4, offering a much larger address space and several functional enhancements to support the growing requirements of modern networks.

IPv6 is based on IP, but adds a much larger address space and improvements such as a simplified main header and extension headers. The architecture of IPv6 helps existing IPv4 users transition to IPv6. It allows continued use of services such as end-to-end security, quality of service (QoS), and globally unique addresses. The larger IPv6 address space allows networks to scale and provide global reachability.

> **Note**
>
> Features and functions that work on IPv4 networks with IPv4 addresses also work on IPv6 networks with IPv6 addresses.

General guidelines and unsupported features of IPv6

This topic outlines the recommended guidelines for configuring IPv6 on controller or embedded wireless controllers and highlights unsupported IPv6 features you should avoid.

- For IPv6 functionality, disable IPv6 multicast routing.
- The Wireless Management interface must have only one static IPv6 address.
- Router advertisements should be suppressed on the wireless management interface and on client VLANs, if IPv6 is configured on the client VLAN.
- The preferred mode is a setting within the AP join profile. When you configure the preferred mode as IPv6, an AP attempts to join over IPv6 first. If it fails, the AP falls back to IPv4.
- You should use MAC addresses for RA tracing of APs and clients.
- APs can join IPv6 controllers only with a static IPv6 address. If you have a controller with auto configurations and multiple IPv6 addresses, APs cannot join those IPv6 controllers.

Unsupported Feature

- UDP Lite is not supported.
- AP sniffer over IPv6 is not supported.
- IPv6 is not supported for the HA port interface.
- Auto RF grouping over IPv6 is not supported. Only static RF grouping is supported.

Configure IPv6 addressing (CLI)

Configure IPv6 addressing for network interfaces using commands.

Enable IPv6 communication and routing capabilities on physical and VLAN interfaces using commands.

> **Note**
>
> All features and functions that operate on IPv4 networks with IPv4 addresses also function on IPv6 networks with IPv6 addresses.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure IPv6 for unicasting.

Example

```
Device(config)# ipv6 unicast-routing
```

3. Create an interface and enter the interface configuration mode.

Example

```
Device(config)# interface vlan 1
```

4. Specify a global IPv6 address.

Example

```
Device(config-if)# ipv6 address FD09:9:2:49::53/64
```

5. Enable IPv6 on the interface.

Example

```
Device(config-if)# ipv6 enable
```

6. Suppress IPv6 router advertisement transmissions on the interface.

Example

```
Device(config-if)# ipv6 nd ra suppress all
```

7. Return to the global configuration mode and configure the ports that are connected to the supported APs with the wireless management interface.

Example

```
Device(config-if)# exit
```

```
Device(config)# wireless management interface gigabitEthernet vlan 64
```

8. Specify IPv6 static routes.

Example

```
Device(config)# ipv6 route ::/0 FD09:9:2:49::1
```

Create an AP join profile (GUI)

Configure an AP join profile through the GUI to set parameters for AP joining.

Set up an AP join profile that controls AP joining behavior and preferences using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. On the **AP Join Profile** window, click the **General** tab and click **Add**.
3. In the **Name** field, enter a name for the AP join profile.
4. Optional: Enter a description for the AP join profile.
5. Choose **CAPWAP** > **Advanced**.
6. Select **IPv6** from the **Preferred Mode** drop-down list under the **Advanced** tab to set the preferred mode of APs to IPv6.
7. Click **Save & Apply to Device**.

Create an AP join profile (CLI)

Create an AP join profile using commands.

Create an AP join profile using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile ap-profile
   ```

3. Add a description for the AP profile.

   Example

   ```
   Device(config-ap-profile)# description ap-profile-name
   ```

4. Set the preferred mode of APs as IPv6.

   Example

   ```
   Device(config-ap-profile)# preferred-mode ipv6
   ```

Configure the primary and backup Controller (GUI)

Use this task to configure both the primary and backup embedded wireless controllers in the GUI for high availability.

Set up a reliable wireless network by designating primary and backup controllers to ensure continuous connectivity in case of failures using the GUI.

Before you begin

Ensure that you have configured an AP join profile prior to configuring the primary and backup controller s.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. On the **AP Join Profile** window, click the AP join profile name.
3. In the **Edit AP Join Profile** window, click the **CAPWAP** tab.
4. In the **High Availability** tab, under **Backup Controller Configuration**, check the **Enable Fallback** check box.
5. Enter the primary and secondary controller names and IP addresses.
6. Click **Update & Apply to Device**.

Configure primary and backup controller (CLI)

Use this procedure to configure the primary and backup controllers for a selected AP using commands.

Set up failover by designating primary and backup controllers on an AP to ensure continuous operation if the primary controller becomes unavailable using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile profile-name
   ```

3. Configure AP CAPWAP parameters with the primary backup controller's name.

   Example

   ```
   Device(config)# capwap backup primary primary-controller-name
   primary-controller-ip
   ```

> **Note**
>
> Enable fast heartbeat so that **capwap backup primary** and **capwap backup secondary** work.
>
> The AP may disconnect if the link between the controller and the AP is unreliable and fast heartbeat is enabled.

4. Configure AP CAPWAP parameters with the secondary backup controller's name.

   Example

   ```
   Device(config)# capwap backup secondary secondary-controller-name
   secondary-controller-ip
   ```

5. Configure the system logging settings for the APs.

   Example

   ```
   Device(config)# syslog host ip-address
   ```

6. Initiate AP image downgrade from a TFTP server for all the APs.

   Example

   ```
   Device(config)# tftp-downgrade tftp-server-ip imagename
   ```

Verify IPv6 configuration

This section lists the commands and interface summaries used to verify IPv6 configuration on wireless devices.

Use these **show** command to verify the IPv6 configuration:

```
Device# show wireless interface summary
Wireless Interface Summary
Interface Name Interface Type VLAN ID IP Address     IP Netmask    MAC Address
-------------------------------------------------------------------------------
Vlan49         Management     49      0.0.0.0       255.255.255.0
001e.f64c.1eff
                                      fd09:9:2:49::54/64
```

181 IPv6 ACL

Topics:

- IPv6 ACL
- Configure IPv6 ACLs
- Verify IPv6 ACL

This chapter explains how to configure, apply, and verify IPv6 access control lists on Cisco devices, detailing ACL types, CLI/GUI procedures, interface application, RA/ND protections, and troubleshooting commands.

IPv6 ACL

Introduces IPv6 access control lists (ACLs), describing their purpose, configuration, and how they compare to IPv4 ACLs in managing network access and data traffic.

An access control list (ACL) is a set of rules used to control or restrict access to a network interface that

- are configured on the device and can be applied to various interfaces such as the management interface, AP-manager interface, dynamic interfaces, or on WLANs, and
- regulate data traffic to and from wireless clients or to protect the controller's central processing unit (CPU).

You can also create a preauthentication ACL for web authentication. Such an ACL is used to allow certain types of traffic before authentication is complete.

IPv6 ACLs support the same options as IPv4 ACLs including source, destination, source and destination ports.

> **Note**
>
> To enable only IPv4 traffic in your network, block all IPv6 traffic. Configure an IPv6 ACL to deny all IPv6 traffic, and apply it to specific WLANs or to all WLANs.

Understand IPv6 ACLs

Understanding IPv6 ACLs.

Types of ACL

Types of ACL.

Per user IPv6 ACL

Per user IPv6 ACL.

- For the per-user ACL, the complete access control entries (ACEs) are configured as text strings on the RADIUS server.
- The ACCESS-Accept attribute contains the ACE, which is then sent to the device and applied directly to the client. When a wireless client roams to a foreign device, the foreign device receives the ACEs as an AAA attribute in the mobility handoff message. Output direction using per-user ACL is not supported.

Filter ID IPv6 ACL

Filter ID IPv6 ACL.

- For the filter-Id ACL, the complete ACEs and the ACL name `filter-id` are configured on the Cisco 9800 controller. Only the filter-id is configured on the RADIUS server.
- The `filter-id` is sent to the device in the ACCESS-Accept attribute. The device looks up the `filter-id` to find the ACEs, and then applies the ACEs to the client. When the client roams at Layer 2 to the foreign device, only the filter-id is sent to the foreign device in the mobility handoff message. Using output filtered ACLs with per-user ACLs is not supported.

Prerequisites for configuring IPv6 ACL

Prerequisites for configuring IPv6 ACL.

- To filter IP Version 6 (IPv6) traffic, create IPv6 access control lists (ACLs) and apply them to interfaces, using the same process as for IP Version 4 (IPv4) named ACLs.
- You can also create and apply input router ACLs to filter Layer 3 management traffic when the switch is running the Network Essentials license.

Restrictions for configuring IPv6 ACL

Restrictions for configuring IPv6 ACL.

With IPv4, you can configure standard and extended numbered IP ACLs, named IP ACLs, and MAC ACLs. IPv6 supports only named ACLs. The IPv6 ACL does not support FlexConnect mode.
The device supports most of the Cisco IOS-supported IPv6 ACLs, with some exceptions:

- The controller does not support routing, and only inbound ACLs are supported for wireless clients.
- The device does not support matching on these keywords: flowlabel, routing header, and undetermined-transport.
- The device does not support reflexive ACLs (the reflect keyword).
- The device does not apply MAC-based ACLs on IPv6 frames.
- There is no restriction on the keywords entered in the ACL, even if they are not supported on the platform. When you apply the ACL to an interface that requires hardware forwarding (physical ports or SVIs), the device checks if the ACL is supported on the interface. If not, attaching the ACL is rejected.
- If not, the device rejects attaching the ACL. If an ACL is already applied to an interface and you attempt to add an access control entry (ACE) with an unsupported keyword, the device does not allow the ACE to be added to the ACL currently attached to the interface.

Configure IPv6 ACLs

Use this task to filter IPv6 traffic by configuring and applying IPv6 access control lists (ACLs) to router interfaces.

Block or permit IPv6 traffic based on policies using ACLs.

To filter IPv6 traffic, perform these steps:

Before you begin

Procedure

1. Create an IPv6 ACL, and enter the IPv6 access list configuration mode.
2. Configure the IPv6 ACL to block (deny) or pass (permit) traffic.
3. Apply the IPv6 ACL to the interface where traffic will be filtered.
4. For router ACLs, assign an IPv6 address to the Layer 3 interface where the ACL is applied.

Default IPv6 ACL configuration

This topic provides the default configuration settings for IPv6 access control lists (ACLs).

There are no IPv6 ACLs configured or applied.

Interact with other features and switches

This topic provides details on how access control lists (ACLs) interact with other features and switch hardware, including supported protocols, error conditions, and software fallback behaviors.

Access control lists (ACLs) on switches interact with other features and platform behaviors in these ways:

- If a port ACL is configured to drop a bridged frame, the frame is not bridged.
- You can create IPv4 and IPv6 ACLs, and apply both to the same interface.
- Each ACL must have a unique name; if not, you receive an error message.
- Different commands are required to create and attach IPv4 and IPv6 ACLs. Using the incorrect command results in an error.
- MAC ACLs cannot filter IPv6 frames; MAC ACLs are for non-IP frames.
- When hardware memory or TCAM is full, additional ACLs are processed in software by the CPU. A console message indicates the ACL has been unloaded and packets will be handled in software.
- Only packets matching the ACL type (IPv4, IPv6, or MAC) that could not be added will be processed in software.
- If the TCAM is full, additional configured ACLs result in packets being forwarded to the CPU, and the ACLs are applied in software.

Create an IPv6 ACL (GUI)

Configure a new IPv6 access control list (ACL) using the GUI for secure network traffic filtering.

Define and apply an IPv6 ACL to control permitted or denied network traffic based on source, destination, protocol, and other parameters using the GUI.

Procedure

1. Choose **Configuration** > **Security** > **ACL**.
2. Click **Add**.
3. In the **Add ACL Setup** dialog box, enter these parameters.
   - ACL Name: Enter the name for the ACL
   - ACL Type: IPv6
   - Sequence: The valid range is between 100 and 199 or 2000 and 26991.
   - Action: Select **Permit** or **Deny** the packet flow from the drop-down list.
   - Source Type: Choose **any**, **Host** or **Network** from which the packet is sent.
   - Destination Type: Choose **any**, **Host** or **Network** to which the packet is sent.
   - Protocol: Select a protocol from the drop-down list.
   - Log: Enable or disable logging.
   - DSCP: Enter to match packets with the DSCP value.
4. Click **Add**.
5. Add the rest of the rules and click **Apply to Device**.

Create an IPv6 ACL (CLI)

Configure an IPv6 Access Control List (ACL) in Cisco IOS using commands to filter IPv6 traffic by protocol, address, port, and conditions.

Define rules to permit or deny IPv6 packets based on specific criteria, enhancing network security and control using commands.

Procedure

1. Enable the privileged EXEC mode and enter the global configuration mode.

   Example

   ```
   Device# enable
   ```

   ```
   Device# configure terminal
   ```

   Enter your password if prompted.

2. Use a name to define an IPv6 access list and enter the IPv6 access-list configuration mode.

   Example

   ```
   Device# ipv6 access-list acl_name
   ```

3. Enter deny or permit to specify whether to deny or permit the packet if conditions are matched.

   Example

   ```
   {deny | permit} protocol {source-ipv6-prefix/prefix-length | any | host
   source-ipv6-address}
   [operator [port-number]]{destination-ipv6-prefix/prefix-length | any |host
   destination-ipv6-address}
   [operator [port-number]][dscp value] [fragments][log] [log-input]
   [routing][sequence value]
   [time-range name]
   ```

   These are the conditions:

   - To set the protocol, enter the name or number of an Internet protocol (ahp, esp, icmp, ipv6, pcp, stcp, tcp, or udp), or enter an integer from 0 to 255 to represent an IPv6 protocol number.
   - Specify the source-ipv6-prefix/prefix-length or destination-ipv6-prefix/prefix-length as the source or destination IPv6 network or network class to set deny or permit conditions. Enter values in hexadecimal, using 16-bit segments separated by colons, as described in RFC 2373.
   - Enter any as an abbreviation for the IPv6 prefix ::/0.
   - For host source-ipv6-address or destination-ipv6-address, enter the specific IPv6 host address. Enter values in hexadecimal, using 16-bit segments separated by colons, to set deny or permit conditions.
   - Specify an operator to compare the source or destination ports of the protocol. Valid operators are lt (less than), gt (greater than), eq (equal), neq (not equal), and range.

   When the operator is placed after the source-ipv6-prefix/prefix-length argument, it matches the source port. When the operator is placed after the destination-ipv6-prefix/prefix-length argument, it matches the destination port.

- The port number can be a decimal value from 0 to 65535 or a TCP or UDP port name. Use TCP port names only for TCP filtering, and UDP port names only for UDP filtering.
- Enter a DSCP value to match a differentiated services code point against the traffic class value in each IPv6 packet header. The valid range is 0 to 63.
- To check noninitial fragments, enter the keyword fragments. This option is available only when the protocol is IPv6.
- To send a log message to the console for matching packets, enter log. To include the input interface in the log entry, enter log-input. Logging is supported only for router ACLs.
- Enter routing to enable IPv6 packet routing.
- Enter a sequence value to assign a sequence number to the access list statement. Valid values range from 1 to 4,294,967,295.
- Enter a time-range name to apply a specific time range to the deny or permit statement.

4. Define a TCP access list and the access conditions.

Example

```
{deny | permit} tcp {source-ipv6-prefix/prefix-length | any |
hostsource-ipv6-address}
[operator [port-number]]{destination-ipv6-prefix/prefix-length | any
|hostdestination-ipv6-address}
[operator [port-number]][ack] [dscp value][established] [fin]
[log][log-input] [neq {port |protocol}] [psh] [range{port | protocol}]
[rst][routing] [sequence value]
[syn] [time-range name][urg]
```

Enter tcp for Transmission Control Protocol. The parameters are the same as those described in Step 3, with these additional optional parameters:

- ack: Acknowledgment bit set.
- established: An established connection. A match occurs if the TCP datagram has the ACK or RST bits set.
- fin: Finished bit set; no more data from sender.
- neq {port | protocol}: Matches only packets that are not on a given port number.
- psh: Push function bit set.
- range {port | protocol}: Matches only packets in the port number range.
- rst: Reset bit set.
- syn: Synchronize bit set.
- urg: Urgent pointer bit set.

5. Define a UDP access list and the access conditions.

Example

```
{deny | permit} udp {source-ipv6-prefix/prefix-length | any |
hostsource-ipv6-address}
[operator [port-number]]{destination-ipv6-prefix/prefix-length | any |
hostdestination-ipv6-address}
[operator [port-number]][dscp value] [log][log-input]
```

```
[neq {port |protocol}] [range {port |protocol}] [routing][sequence
value][time-range name]
```

Enter udp for the User Datagram Protocol. The UDP parameters are similar to those for TCP. However, the operator [port] must specify a UDP port number or name, and the established parameter cannot be used with UDP.

6. Define an ICMP access list and the access conditions.

   Example

   ```
   {deny | permit} icmp {source-ipv6-prefix/prefix-length | any |
   hostsource-ipv6-address}
   [operator [port-number]] {destination-ipv6-prefix/prefix-length | any |
   hostdestination-ipv6-address}
   [operator [port-number]][icmp-type [icmp-code] |icmp-message] [dscpvalue]
   [log] [log-input]
   [routing] [sequence value][time-range name]
   ```

   Enter icmp for Internet Control Message Protocol. The ICMP parameters are the same as those for most IP protocols in Step 3a. However, ICMP includes additional parameters for message type and code. These optional keywords have these meanings:

   - icmp-type: Enter to filter by ICMP message type, a number from 0 to 255.
   - icmp-code: Enter to filter ICMP packets by the ICMP message code type, a number from 0 to 255.
   - icmp-message: Enter to filter ICMP packets by the ICMP message type name or the ICMP message type and code name. To view a list of ICMP message type names and code names, use the ? key or consult the command reference for this release.

7. Returns to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

8. Verify the access list configuration.

   Example

   ```
   show ipv6 access-list
   ```

9. Optional: Save your entries in the configuration file.

   Example

   ```
   copy running-config startup-config
   ```

Create WLAN IPv6 ACL (GUI)

Create a WLAN IPv6 Access Control List (ACL) using the GUI to apply IPv6 traffic controls to a wireless LAN.

Create a WLAN IPv6 Access Control List (ACL) using the GUI to apply IPv6 traffic controls to a wireless LAN.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name** , the **SSID** and the **WLAN ID**.
4. Choose **Security** > **Layer3** tab. Click **Show Advanced Settings**. Under the **Preauthenticated ACL** settings, select the ACL from the **IPv6** drop-down list.
5. Click **Apply to Device**.

Create WLAN IPv6 ACL (CLI)

Configure IPv6 ACLs on a WLAN using commands to secure and control network traffic.

Create and apply a WLAN IPv6 Access Control List (ACL) to define traffic rules and enforce security for wireless clients using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create policy profile for the WLAN.

   Example

   ```
   Device(config)# wireless profile policy profile-name
   ```

   The *profile-name* is the profile name of the policy profile.

3. Create a named WLAN ACL.

   Example

   ```
   Device(config-wireless-policy)# ipv6 acl acl_name
   ```

4. Create a pre-authentication ACL for web authentication.

   Example

   ```
   Device(config-wlan)# ipv6 traffic-filter web acl_name-preauth
   ```

Verify IPv6 ACL

Verify IPv6 ACL.

Display IPv6 ACLs (CLI)

Learn how to use commands to view IPv6 access control lists configured on a network device.

View all IPv6 access lists or a specific access list by name for verification and troubleshooting using commands.

Procedure

1. Enable the privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

   Enter your password if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Display all access lists configured on the device

   Example

   ```
   Device# show access-lists
   ```

4. Display all configured IPv6 access list or the access list specified by name.

   Example

   ```
   Device# show ipv6 access-list acl_name
   ```

Example: Create an IPv6 ACL

Example for creating an IPv6 ACL.

This example configures the IPv6 access list named CISCO. The first deny entry denies all packets with a destination TCP port number greater than 5000. The second deny entry denies packets with a source UDP port number less than 5000. The second deny entry also logs all matches to the console. The first permit entry permits all ICMP packets. The second permit entry permits all other traffic. The second permit entry is necessary because each IPv6 access list ends with an implicit deny all condition.

> **Note**
>
> Logging is supported only on Layer 3 interfaces.

```
Device(config)# ipv6 access-list CISCO
```

```
Device(config-ipv6-acl)# deny tcp any any gt 5000
Device(config-ipv6-acl)# deny ::/0 lt 5000 ::/0 log
Device(config-ipv6-acl)# permit icmp any any
Device(config-ipv6-acl)# permit any any
```

Example: Apply an IPv6 ACL to a policy profile in a wireless environment

This topic explains the examples for applying an IPv6 ACL to a policy profile.

These example shows how to apply an IPv6 ACL to a policy profile in a wireless environment.

> **Note**
>
> All IPv6 ACLs must be associated to a policy profile.

1. Create an IPv6 ACL.

```
Device
                              (config)# ipv6 access-list <acl-name>

Device
                              (config-ipv6-acl)# permit tcp 2001:DB8::/32 any

Device
                              (config-ipv6-acl)# permit udp 2001:DB8::/32 any
```

2. Apply the IPv6 ACL to a policy profile.

```
Device
                               (config)# wireless profile policy
<policy-profile-name>
Device
                              (config-wireless-policy)# shutdown

Device
                              (config-wireless-policy)# ipv6 acl <acl-name>

Device
                          (config-wireless-policy)# no shutdown
```

Display IPv6 ACLs (CLI)

This topic shows you how to display all configured IPv6 access control lists (ACLs) or a specific ACL by using commands on your device.

This topic shows you how to display all configured IPv6 access control lists (ACLs) or a specific ACL by using commands on your device.

To display IPv6 ACLs, perform this procedure:

Procedure

1. Display all access lists configured on the device.

   Example

   ```
   Device# show access-lists
   ```

2. Display all configured IPv6 access list or the access list specified by name.

   Example

   ```
   Device# show ipv6 access-list acl_name
   ```

Example: Display IPv6 ACLs

Example for displaying IPv6 ACLs.

This is an example of the output from the **show access-lists** privileged EXEC command. The output shows all access lists that are configured on the switch or switch stack .

```
Device# show access-lists
Extended IP access list hello
10 permit ip any any
IPv6 access list ipv6
permit ipv6 any any sequence 10
```

This is an example of the output from the **show ipv6 access-lists** privileged EXEC command. The output shows only IPv6 access lists configured on the switch or switch stack.

```
Device# show ipv6 access-list
IPv6 access list inbound
permit tcp any any eq bgp (8 matches) sequence 10
permit tcp any any eq telnet (15 matches) sequence 20
permit udp any any sequence 30
IPv6 access list outbound
deny udp any any sequence 10
deny tcp any any eq telnet sequence 20
```

Example: Configure RA throttling

This task describes how to create and apply an RA throttle policy using the controller.

Configure router advertisement (RA) throttling and neighbor solicitation (NS) suppression to prevent power-saving wireless clients from being disturbed by frequent unsolicited RA messages using commands.

This task describes how to create an RA throttle policy in order to help the power-saving wireless clients from being disturbed by frequent unsolicited periodic RA's. The unsolicited multicast RA is throttled by the controller .

Before you begin

Enable IPv6 on the client machine.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a RA throttler policy called Mythrottle.

   Example

   ```
   Device(config)# ipv6 nd ra-throttler policy Mythrottle
   ```

3. Determine the time interval segment during which throttling applies.

   Example

   ```
   Device(config-nd-ra-throttle)# throttle-period 20
   ```

4. Determine how many initial RA's are allowed.

   Example

   ```
   Device(config-nd-ra-throttle)# max-through 5
   ```

5. Determine how many RA's are allowed after the initial RAs have been transmitted, until the end of the interval segment.

   Example

   ```
   Device(config-nd-ra-throttle)# allow at-least 3 at-most 5
   ```

6. Create a per VLAN configuration.

   Example

   ```
   Device(config)# vlan configuration 100
   ```

7. Enable the router advertisement throttling.

   Example

   ```
   Device(config)# ipv6 nd ra-throttle attach-policy attach-policy_name
   ```

8. Return to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

182 Configuring IPv6 Mobility

Topics:

- IPv6 client mobility

IPv6 client mobility

Explains how IPv6 client mobility enables seamless Layer 2 and Layer 3 roaming for IPv6-only and dual-stack wireless nodes, providing consistent connectivity and correct IP address management.

A client mobility is a wireless network feature that

- supports seamless Layer 2 and Layer 3 roaming for IPv6-only and dual-stack clients
- manages IPv6 client addresses and Neighbor Discovery (NDP) packet handling across controllers, and
- retains the client's IP address and application connectivity as they move between access points and switches in the mobility group.

Link layer mobility is not sufficient for wireless client Layer 3 applications to work seamlessly while roaming. Cisco IOSd's wireless mobility module uses mobility tunneling to retain seamless connectivity for the client's Layer 3 PoP (point of presence) when the client roams across different subnets on different switches.

IPv6 is the next-generation network layer Internet protocol intended to replace IPv4 in the TCP/IP suite of protocols. This new version increases the global internet address space to accommodate users and applications that require unique global IP addresses. IPv6 incorporates 128-bit source and destination addresses, which provide significantly more addresses than the 32-bit IPv4 addresses.

To support IPv6 clients across controllers, ICMPv6 messages must be handled in a specific way to ensure the IPv6 client remains on the same Layer 3 network. The device keeps track of IPv6 clients by intercepting ICMPv6 messages to provide seamless mobility and protect the network from network attacks. Neighbor Discovery Protocol (NDP) packets are converted from multicast to unicast and delivered individually to each client. This unique solution ensures that Neighbor Discovery and Router Advertisement packets are not leaked across VLANs. Clients receive specific Neighbor Discovery and Router Advertisement packets, ensuring correct IPv6 addressing and avoiding unnecessary multicast traffic.

The configuration for IPv6 mobility is the same as for IPv4 mobility and does not require separate software on the client side to achieve seamless roaming. The device must be part of the same mobility group. Both IPv4 and IPv6 client mobility are enabled by default.

IPv6 client mobility provides these features:

- Retains multiple client IPv6 addresses during Layer-2 and Layer-3 roaming.
- Manages IPv6 Neighbor Discovery Protocol (NDP) packets.
- Learns client IPv6 addresses.

> **Note**
>
> The configuration for IPv6 mobility in SDA wireless and Local mode is the same as for IPv4 mobility and does not require different software configuration on the client side to achieve seamless roaming. Refer to the IPv4 mobility section for configuration information.

> **Note**
>
> f an IPv6 address is configured on the SVI, configure the **ipv6 nd ra suppress all** command on all client VLAN SVI interfaces on the controller. This prevents multiple devices from advertising themselves as routers.

Using router advertisement

This topic describes how Router Advertisement (RA) messages in the IPv6 Neighbor Discovery Protocol (NDP) enable wireless client mobility and IPv6 address acquisition in converged access network environments.

The Neighbor Discovery Protocol (NDP) is a protocol that

- operates at the link layer and manages the discovery of other nodes on the network link
- determination of their link-layer addresses, identification of available routers, and
- the maintenance of reachability information for neighboring nodes.

A Router Advertisement (RA) is an IPv6 Neighbor Discovery Protocol (NDP) message

- sent by routers to communicate network parameters such as available routers
- network prefixes for IPv6 address configuration, and
- link MTU information to hosts on the link.

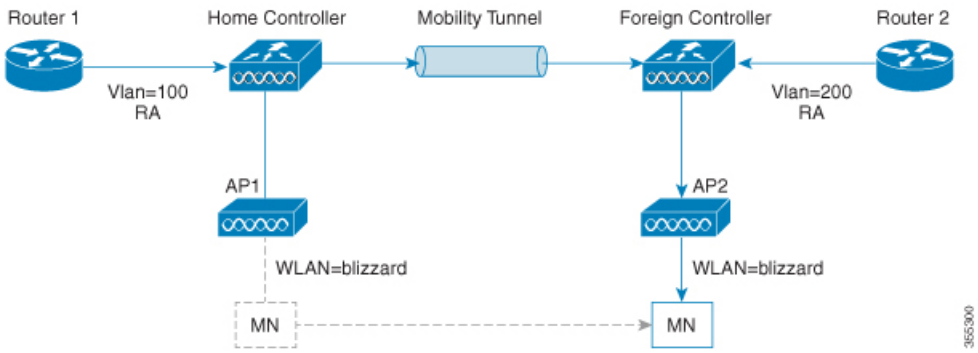
RAs play a vital role in this process by allowing hosts to detect routers, generate IPv6 addresses, and configure other connection parameters.

Routers send RAs either on a regular basis or in response to Router Solicitation messages from hosts.

IPv6 wireless client mobility manages the IPv6 RA packet. The device forwards the link-local all-nodes multicast RA packets to both local and roaming wireless nodes mapped on the same VLAN where the RA was received.

Figure 1 illustrates how a roaming client “MN” receives RA from VLAN 200 in a foreign controller and how it acquires an new IP address and breaks into L3 mobility's point of presence.

Figure 33: Roaming Client Receives Valid RA from Router 1



Router advertisement throttling

This topic explains how router advertisement throttling limits the frequency of RA packets and maintains IPv6 connectivity for clients.

A router advertisement throttle is a network control mechanism that

- enforces limits on the frequency of router advertisement (RA) packets sent toward the wireless network
- ensures routers that send multiple RA packets are restricted to a minimum frequency required for IPv6 client connectivity, and
- maintains uninterrupted IPv6 connectivity for new or roaming clients by allowing RA packets in response to Router Solicitation (RS) packets.

If a client sends an RS packet, the router sends an RA back to the client. The controller allows this RA to pass and delivers it as a unicast message to the client. This process ensures that new or roaming clients are not affected by RA throttling.

IPv6 address learning

This topic explains how IPv6 clients obtain their addresses and how network devices detect and record these address assignments.

IPv6 address learning is the process

- by which IPv6 clients acquire IP addresses, and
- network devices such as switches or controllers, observe and record these assignments.

There are three ways for an IPv6 client to acquire IPv6 addresses:

- Stateless Address Auto-Configuration (SLAAC)
- Stateful DHCPv6
- Static configuration

For these methods, the IPv6 client always sends NS DAD (duplicate address detection) to ensure that there is no duplicated IP address on the network. The device snoops the client's NDP and DHCPv6 packets to learn about the client's IP addresses. It then updates the controller's database. The database then informs the controller about the client's new IP address.

Handling multiple IP addresses

This concept explains how IP addresses are managed for network clients, including their assignment, update, and provisioning to external systems.

An IP address is a network identifier that

- uniquely distinguishes a device on a network
- supports both IPv4 and IPv6 formats, and
- can be dynamically assigned, updated, or displayed for client devices.

If a new IP address is received after the RUN state, whether it is an addition or removal, the controller updates its local database with the new IP addresses for display purposes. IPv6 uses the same PEM state machine code flow as IPv4. When external entities, such as Prime Infrastructure, request IP addresses, the controller includes all available IPv4 and IPv6 addresses in the API or SPI interface response.

An IPv6 client can acquire multiple IP addresses from the stack for different purposes. For example, a client may have a link-local address for local traffic and a routable unique local or global address.

When the client is in the DHCP request state and the controller receives the first IP address notification from the database, for either an IPv4 or IPv6 address, the PEM moves the client into the RUN state.

If a new IP address is received after the RUN state, whether it is an addition or removal, the controller updates its local database with the new IP addresses for display purposes.

When external entities, such as Prime Infrastructure, request IP addresses, the controller provides all available IPv4 and IPv6 addresses.

IPv6 configuration

This topic explains the key attributes of IPv6 configurations in a switch controller device, including client support, VLAN setup, and packet handling

A IPv6 configuration is a network setup that

- enables seamless support for IPv6 clients alongside IPv4 clients
- requires manual VLAN configuration by the administrator to activate IPv6 features, and
- implements snooping and throttling functionality to manage NDP packets between the switch controller device and its clients.

Prerequisites for IPv6 client mobility

This topic explains the prerequisites for IPv6 client mobility.

Wireless IPv6 client mobility requires the wired and wireless network infrastructure to support IPv6 routing, suitable address assignment mechanisms, and compatible device configurations.

- To enable wireless IPv6 client connectivity, the underlying wired network must support IPv6 routing and an address assignment mechanism, such as SLAAC or DHCPv6. The device must have Layer 2 adjacency to the IPv6 router. The VLAN should be tagged when packets enter the device. APs do not require connectivity on an IPv6 network because all traffic is encapsulated inside the IPv4 CAPWAP tunnel between the AP and the device.
- Ensure that all clients support IPv6 address assignment using either stateless autoconfiguration, such as Windows XP clients, or stateful DHCPv6 IP addressing, such as Windows 7 clients.

  Verify that your network switch or router supports the DHCP for IPv6 feature and is configured as a DHCPv6 server, such as a Cisco Catalyst 3850 or 5700 switch. Alternatively, use a dedicated server, such as a Windows 2008 server with a built-in DHCPv6 server.

Monitor IPv6 client mobility

This section provides a list of commands used to monitor IPv6 client mobility on the device, including their functions and descriptions.

- This table lists the commands available for monitoring IPv6 client mobility and their corresponding descriptions:

Commands	Description
show wireless client summary	Displays the wireless specific configuration of active clients.
show wireless client mac-address	Displays the wireless specific configuration of active clients based on their MAC address.

183 IPv6 Support on Flex and Mesh

Topics:

- IPv6 support on FlexConnect + mesh deployment

This chapter shows how to enable and configure IPv6 for FlexConnect with mesh on Cisco wireless controllers, covering CLI and GUI steps to set IPv6 addresses, AP preferred-mode, PIM dense-mode, and verification.

IPv6 support on FlexConnect + mesh deployment

Explains IPv6 support and traffic handling when using FlexConnect and mesh deployments with Cisco Catalyst Wireless Controllers.

A FlexConnect and mesh deployment is a wireless network configuration that

- enables IPv6 as the primary backhaul transport for service provider
- supports IPv6 clients connecting via WLANs on the Cisco Catalyst 9800 series wireless controller and Cisco Embedded Wireless controller on Catalyst APs, and
- forwards IPv6 client traffic efficiently to the network core.

Configure IPv6 support for FlexConnect + Mesh (CLI)

This topic enables IPv6 routing and multicast features on controller-based FlexConnect and Mesh interfaces by guiding you through the required commands.

Follow these steps to enable IPv6 routing and multicast features for FlexConnect and Mesh on a controller using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create an interface and enter the interface configuration mode.

   Example

   ```
   Device(config)#interface vlan vlan-interface-number
   ```

3. Disable the interface configuration.

   Example

   ```
   Device(config-if)#shutdown
   ```

4. Optional: Enable IPv6 on the interface.

   Example

   ```
   Device(config-if)#ipv6 enable
   ```

5. Configure IPv6 address on the interface using the IPv6 prefix option.

   Example

   ```
   Device(config-if)#ipv6 address X:X:X:X::X/<0-128>
   ```

6. Enable the IPv6 address.

   Example

   ```
   Device(config-if)#no shutdown
   ```

7. Enable the PIM dense-mode operation.

   Example

   ```
   Device(config-if)#no shutdown
   ```

8. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-if)#end
   ```

9. Verify your entries and check the gateway connectivity.

   Example

   ```
   Device# show ipv6 interface brief
   ```

   ```
   Device# ping ipv6 destination-address or hostname
   ```

Configure preferred IP address as IPv6 (GUI)

Configure your AP join profile to use IPv6 as the preferred IP address mode in Cisco Wireless Controller GUI.

Set the preferred IP address mode to IPv6 for APs joining the controller using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Join**.
2. Click the AP Join Profile Name. The **Edit AP Join Profile** window is displayed.
3. Choose **CAPWAP** > **Advanced**.
4. Select **IPV6** from the **Preferred Mode** drop-down list.
5. Click **Update & Apply to Device**.

Configure preferred IP address as IPv6 (CLI)

Configure an AP profile so that the preferred IP address is set to IPv6 on a controller using commands.

Set the AP to use IPv6 as its preferred IP address when joining the controller using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile default-ap-profile
   ```

3. Use IPv6 to join the controller .

   Example

   ```
   Device(config-ap-profile)# preferred-mode ipv6
   ```

4. Exit the configuration mode and return to the privileged EXEC mode.

   Example

   ```
   Device(config-ap-profile)# end
   ```

Verify IPv6 on FlexConnect + Mesh

This topic provides commands and output examples for verifying IPv6 configuration and interface status on a controller embedded wireless controller operating with FlexConnect and Mesh.

To verify the IPv6 configuration on the controller , use this **show** command:

```
Device# show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet2       unassigned      YES unset  up                    up

GigabitEthernet0       unassigned      YES NVRAM  administratively down down

Capwap1                unassigned      YES unset  up                    up

Capwap2                unassigned      YES unset  up                    up

Vlan1                  unassigned      YES NVRAM  administratively down down

Vlan89                 9.10.89.90      YES NVRAM  up                    up

Ewlc-9.10.89.90#show running-config interface vlan 89
Building configuration...
Current configuration : 120 bytes
!
interface Vlan89
 ip address 9.10.89.90 255.255.255.0
 ip helper-address 9.1.0.100
```

```
 no mop enabled
 no mop sysid
end
```

184 IPv6 CAPWAP UDP Lite Support

Topics:

- UDP Lite

This chapter describes how to enable and verify UDP Lite for IPv6 CAPWAP on controllers and APs using commands, explains firewall considerations, and notes limitations such as lack of support for mobility IPv6 tunnels.

UDP Lite

Describes UDP Lite support, its purpose in IPv6 environments for controllers and APs, and important operational impacts and restrictions.

UDP Lite Support is an enhancement to the IPv6 functionality on controllers and APs

- that enables support for the UDP Lite protocol, and
- minimizes performance impact by restricting the checksum calculation coverage for the UDP Lite header to 8 bytes rather than requiring a complete payload checksum as standard UDP over IPv6 does.

This feature applies only to the IPv6 addresses of the controller and APs. IPv6 mandates a complete payload checksum for UDP.

Intermediate firewalls must allow UDP Lite protocol (protocol ID 136) packets when using the UDP Lite Support feature. Some firewalls might not provide the option to open specific ports on UDP Lite protocol. In these cases, the administrator must open all ports on UDP Lite.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 162: Feature history for IPv6 CAPWAP UDP Lite support

Feature Name	Release Information	Feature Description
IPv6 CAPWAP UDP Lite support	Cisco IOS XE Cupertino 17.1.1s	From this release, the UDP Lite support feature is an enhancement to the existing IPv6 functionality to support the UDP Lite protocol.

Restrictions for UDP Lite support

- Mobility IPv6 tunnels do not support the UDP Lite support feature.

Enable UDP lite support (CLI)

Use this procedure to enable UDP Lite for IPv6 CAPWAP data packets on an AP profile using commands.

Enable UDP Lite, allowing APs to handle IPv6 CAPWAP data packets using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter the AP profile configuration mode.

Example

```
Device(config)# ap profile ap-profile
```

3. Enable IPv6 CAPWAP UDP lite on the AP.

Example

```
Device(config-ap-profile)# capwap udplite
```

> **Note**
>
> This message is displayed after the configuration:
>
> ```
> This feature is supported only for IPv6 data packets, APs will be rebooted.
> ```

4. Return to the privileged EXEC mode.

Example

```
Device(config-ap-profile)# end
```

Alternatively, you can also press *Ctrl-Z* to exit global configuration mode.

Verify UDP lite support configuration

This section lists the command and output used to verify UDP Lite support configuration on a device.

To verify the CAPWAP UDP Lite status, use this command:

```
Device# show ap profile name default-ap-profile detailed

CAPWAP UDP-Lite                 : ENABLED
Lawful-Interception             : ENABLED
LI timer                      : 60
AWIPS                           : DISABLED
AWIPS Forensic                  : Unknown
Client RSSI Statistics
Reporting                     : ENABLED
Reporting Interval            : 30 seconds
```

185 Neighbor Discovery Proxy

Topics:

- Neighbor discovery

This chapter explains configuring a controller's IPv6 neighbor discovery and duplicate address detection proxies in central switching mode, detailing behavior, use cases, and commands to enable full-proxy and DAD-proxy functions.

Neighbor discovery

Explains how the Neighbor Discovery Protocol enables IPv6 nodes to signal their presence on a network link and obtain information about nearby systems.

A neighbor discovery protocol (NDP) is a communication protocol that

- uses ICMPv6 messages to allow nodes within the same IPv6 network link to detect and signal their presence
- employs solicited-node multicast addresses to track and discover other IPv6 hosts connected to network interfaces, and
- facilitates address resolution and reachability checks between nodes using neighbor solicitation (NS) and neighbor advertisement (NA) messages.

Neighbor discovery proxy

This concept explains how neighbor discovery proxies operate in wireless controller environments to manage IPv6 neighbor solicitation messages and reduce multicast traffic to wireless clients.

A neighbor discovery proxy is a controller feature that

- intercepts IPv6 neighbor solicitation packets destined for wireless client
- responds on behalf of the clients to reduce unnecessary multicast traffic, and
- operates only in central switching mode for efficient network management.

If neighbor discovery proxy is not enabled, the multicast neighbor solicitation is converted into unicast neighbor solicitation with the MAC address of the target client and is forwarded to that client.

> **Note**
>
> A controller does not proxy the neighbor solicitation packet if the destination address is not that of a wireless client.

Configure neighbor discovery proxy (CLI)

This topic explains IPv6 neighbor discovery (ND) proxy functionality on a device using commands.

Follow these steps to configure the neighbor discovery proxy using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

Example

```
Device(config)# wireless profile policy policy policy-name
```

3. Enable ND proxy.

Example

```
Device(config-wireless-policy)# ipv6 nd proxy full-proxy
```

Duplicate address detection proxy

The IPv6 duplicate address detection (DAD) proxy feature ensures that IP addresses assigned to wireless devices are unique by intercepting and responding to DAD requests on their behalf.

IPv6 duplicate address detection (DAD) is a mechanism that is

- used to verify that an IPv6 address is unique on a network segment before it is assigned to a device
- acts on behalf of wireless clients to ensure no duplicate addresses are assigned, and
- the proxy responds to DAD neighbor solicitation (NS) messages, preventing address conflicts among wireless clients.

DAD verifies whether a host address is unique. The IPv6 DAD Proxy feature responds on behalf of the address owner when an address is in use.

However, in a scenario where nodes are restricted from talking to each other at Layer 2, DAD cannot detect a duplicate address. If the DAD proxy is disabled, the multicast packet is converted into a unicast packet and sent to the target client.

Note

- DAD proxy is applicable only in central switching mode.
- A controller does not proxy the DAD NS packet if the destination address is not that of a wireless client.

Configure duplicate address detection proxy (CLI)

Configure a duplicate address detection (DAD) proxy for IPv6 networks using commands.

Enable DAD proxy to prevent duplicate IPv6 address assignment by monitoring and responding to neighbor discovery requests using commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a WLAN policy profile and enter the wireless policy configuration mode.

Example

```
Device(config)# wireless profile policy policy-profile-name
```

3. Enable DAD proxy.

Example

```
Device(config-wireless-policy)# ipv6 nd proxy dad-proxy
```

> **Note**
>
> Full proxy configuration includes both ND proxy and DAD proxy configurations. To enable DAD proxy, use the **ipv6 nd proxy full-proxy** command.

186 Address Resolution Protocol Proxy

Topics:

- Address resolution protocol

This chapter explains how a wireless controller handles ARP requests for wireless clients, describing ARP proxy behavior, broadcast suppression, and command steps to enable ipv4 arp-proxy in a WLAN policy profile.

Address resolution protocol

Explains how the Address Resolution Protocol (ARP) maps IP network addresses to hardware addresses used by data link protocols.

An address resolution protocol is a network protocol that

- maps IP network addresses to hardware (MAC) addresses used by data link protocols
- enables devices in a local network to discover the physical address associated with an IP address, and
- facilitates communication by resolving addresses required for packet delivery on the network.

The Address Resolution Protocol (ARP) is used by the Internet Protocol (IP) [RFC826], specifically in IPv4 networks. When a wireless client sends an ARP request for an IP address of interest, the controller performs a search for that address in its database. If an entry is found in the controller database, then the ARP is converted to unicast and forwarded to that particular client. If there is no entry in the controller's database, the ARP request is flooded out to the VLAN wired ports.

Address resolution protocol proxy

This topic explains how proxy ARP enables a controller to respond to ARP requests for wireless clients, helping manage broadcast traffic and enhance network efficiency.

A proxy ARP operation is a wireless controller feature that

- responds to ARP request packets on destined for wireless clients
- minimizes broadcast traffic by limiting ARP requests to appropriate clients, and
- is applicable only in central switching mode for wireless client destinations.

ARP Proxy enables the controller to respond to ARP request packets destined for wireless clients. During broadcast suppression, the controller checks if proxy is enabled for the destined wireless clients. If proxy is enabled, the controller drops the ARP request packet and generates a response to the source so that the packet appears to come from the wireless client. This helps in limiting the traffic to the wireless clients.

If ARP Proxy is not enabled, the broadcast ARP request is converted into a unicast ARP request with the MAC address of the target client and forwarded only to that client.

> **Note**
>
> A device will not proxy the ARP request if the destination address does not belong to a wireless client.

Configure address resolution protocol proxy (CLI)

Learn how to configure the Address Resolution Protocol (ARP) proxy on a wireless policy profile using commands.

Enable ARP proxying for client devices on a WLAN by updating the relevant policy profile settings using commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a WLAN policy profile and enter the wireless policy configuration mode.

Example

```
Device(config)# wireless profile policy policy-profile-name
```

3. Enable ARP proxy.

Example

```
Device(config-wireless-policy)# ipv4 arp-proxy
```

187 IPv6 Ready Certification

Topics:

- IPv6 ready certification

This chapter describes the IPv6 RFC-compliance features added in Cisco IOS XE Bengaluru 17.6.1, packet fragmentation rules, neighbor discovery handling, MTU behavior, hop-by-hop processing, and route information option configuration and verification to enable secure, standards-based IPv6 deployment.

IPv6 ready certification

Explains the IPv6 features required for IPv6 Ready Certification compliance in Cisco IOS XE Bengaluru, 17.6.1.

A functionality is a network capability that

- enables compliance with the latest RFC specifications for IPv6
- supports key IPv6 packet and protocol operations, and
- ensures robust and secure IPv6 communication in modern network environments

Starting with Cisco IOS XE Bengaluru, 17.6.1, these IPv6 functionalities are implemented:

- **Fragment Processing and Reassembly (RFC8200)**: The first fragment must contain the mandatory extension header up to the first upper level protocol (ULP) header as specified in RFC 8200.
- **Handling Atomic Fragments in Neighbor Discovery (RFC6980)**: Fragmented neighbor discovery packets must be dropped.
- **Packet too Big (RFC8201)**: Atomic fragmentation is not supported. Packets that do not meet the IPv6 MTU requirement of 1280 bytes are dropped.
- **Route Information Options (RIO) in IPv6 Router Advertisements (RFC4191)**: A new RIO is added to the IPv6 Router Advertisement message to communicate specific routes from routers to hosts. Explicit route configuration ensures that only necessary routes are advertised.
- **IPv6 Hop-by-Hop Processing (RFC 8200)**: This enhancement allows explicit configuration of nodes along the delivery path for packets that require hop-by-hop options header processing.

Feature History for IPv6 ready certification

This table provides release and related information for the feature explained in this module.

This feature is available in all the releases subsequent to the one in which it is introduced in, unless noted otherwise.

Table 163: Feature History for IPv6 ready certification

Release	Feature	Feature Information
Cisco IOS XE Bengaluru 17.6.1	IPv6 ready certification	This feature is enhanced with the implementation of various IPv6 functionalities that are required to comply with the latest RFC specifications.

Configure IPv6 route information (CLI)

Configure the route information option (RIO) in IPv6 router advertisement messages so hosts can receive specific routes and select the optimal default router.

Ensure that IPv6 hosts, especially those on multihomed networks, receive required explicit routes to improve their ability to select the appropriate default router.

The Route Information Option (RIO) in the IPv6 router advertisement messages helps in communicating specific routes from routers to hosts. This improvement enables a host to select an appropriate default router when it is multihomed and the routers are on different links. Only necessary routes are advertised to the hosts because of the explicit route configuration.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the interface and enter the interface configuration mode.

   Example

   ```
   Device(config)# interface gigabitethernet1.1
   ```

3. Configure RIO in IPv6 router advertisement messages.

   Example

   ```
   Device(config-if)# ipv6 nd ra specific-route prefix/length lifetime
   lifetime/infinity [ preference preference]
   ```

   For more information, see the ipv6 nd ra specific route command.

Verify IPv6 route information

This section list the command you can use to view IPv6-specific routes in router advertisements and to examine hop-by-hop processing in an IPv6 node.

To identify the specific routes that are sent in the router advertisements, use this command:

```
Device# show ipv6 nd ra specific-route
IPv6 Prefix/Length Lifetime Preference Interface
---------------------------------------- -------- ---------- ---------
1234::12/127 1000 High GigabitEthernet2
```

Part XV

CleanAir

Topics:

- Cisco CleanAir
- Bluetooth Low Energy
- Persistent Device Avoidance
- Spectrum Intelligence
- Spectrum Analysis

188 Cisco CleanAir

Topics:

- Cisco CleanAir
- Configure CleanAir
- CleanAir Pro scanning
- Verify CleanAir parameters
- CleanAir FAQs

Cisco CleanAir

Introduces Cisco CleanAir technology, describing key terminology, system components, and comparisons with Wi-Fi chip-based RF management. Explains persistent device handling, spontaneous interference, device merging, EDRRM and AQR update modes, high availability, and important operational considerations.

A Cisco CleanAir system is a wireless spectrum management solution that

- identifies and monitors all users and interferers in the industrial, scientific, and medical (ISM) bands
- provides real-time radio frequency (RF) visibility and actionable interference information, and
- enables the network to automatically or manually respond to interference, improving wireless performance.

Table 164: Feature history for Cisco CleanAir

Release	Feature	Feature Information
Cisco IOS XE 17.13.1	EDRRM Support for 6 GHz Band Radio	The Event-Driven Radio Resource Management (EDRRM) is enabled in the 6 GHz band radio of AP.

Cisco CleanAir systems consist of CleanAir-enabled APs and a Cisco Catalyst 9800 Series Wireless Controller. The APs gather device information across the ISM bands, detect and evaluate interference sources, and forward this information to the controller. The controller manages the APs and displays detected interference devices.

For devices operating in the unlicensed band, Cisco CleanAir systems provide detailed information about device identity, interference impact, and recommended actions to mitigate issues. CleanAir simplifies RF management, allowing users to maintain optimal wireless performance without requiring deep RF expertise.

Components of Cisco CleanAir systems

Cisco CleanAir systems are comprised of several key hardware and software elements:

- **Cisco CleanAir-enabled APs**: These devices actively scan the wireless spectrum for all active transmitters.
- **Wireless controllers**: Cisco Catalyst 9800 Series Wireless Controller.
- **Management platforms**: Provides centralized management and visualization of spectrum and interference data forwarded by the controllers.

How Cisco CleanAir systems detect and respond to interference

1. **Always-On spectrum awareness**: Cisco CleanAir-enabled APs scan the industrial, scientific, and medical (ISM) bands, identifying and evaluating all radio devices that may cause interference.
2. **Unified control and interference intelligence**: The controller controls the access points and displays the interference devices.
3. **Actionable insight for every interferer**: For every device operating in the unlicensed band, Cisco CleanAir provides information about what it is, how it is impacting your wireless network, and what actions you or your network should take. It simplifies RF so that you do not have to be an RF expert.

Benefits of Cisco CleanAir systems

- **Simplifies spectrum management**: Provides visibility into interference without requiring advanced RF expertise.
- **Improves network reliability**: Automatically detects and resolves issues affecting Wi-Fi performance.

- **Supports advanced WLAN services**: Ensures quality for voice over wireless and IEEE 802.11 operations by minimizing RF interference.

Detecting and mitigating interference

In a busy office environment, multiple devices such as microwave ovens, cordless phones, and Bluetooth devices operate in the 2.4-GHz and 5-GHz ISM bands. These can disrupt Wi-Fi communication. Cisco CleanAir systems detect these interferers, identify their impact, and automatically adjust wireless channels or prompt users to take corrective action.

A basic wireless LAN without CleanAir functionality cannot detect, identify, or respond to environmental interference in real time, meaning performance issues can go unnoticed and unresolved.

Cisco CleanAir systems function like air quality sensors in a smart building—they continuously monitor the environment, detect harmful agents, and trigger responses to ensure optimal conditions for all users.

Cisco CleanAir-related terms

Cisco CleanAir-related terms.

Table 165: Cisco CleanAir-related terms

Term	Description
AQI	Air Quality Index. The AQI indicates air quality based on RF interference. An AQI of zero is considered poor, and an AQI greater than eighty-five is considered good.
AQR	Air Quality Report. An AQR shows total interference from all sources, including major categories. The system sends an AQR every 15 minutes and every 30 seconds in Rapid mode.
DC	Duty Cycle. Indicates the percentage of time your device uses the channel.
EDRRM	Event-Driven RRM. When needed, EDRRM enables your access point to skip normal RRM intervals and immediately change channels.
IDR	Interference Device Reports that an AP sends to the controller .
ISI	Interference Severity Index. The ISI is an indicator of the severity of the interference.
RSSI	Received Signal Strength Indicator. RSSI measures the power in a received signal. It indicates the level detected by an access point from an interfering device.

Cisco CleanAir components

This process shows how the main Cisco CleanAir components cooperate to monitor, analyze, and mitigate RF interference in wireless networks. It outlines each component's role and the sequence of operations in the CleanAir system.

Summary

Cisco CleanAir technology uses a coordinated set of hardware and software components to detect, report, and address wireless interference. Each part of the system contributes to maintaining optimal air quality and reliable wireless performance.

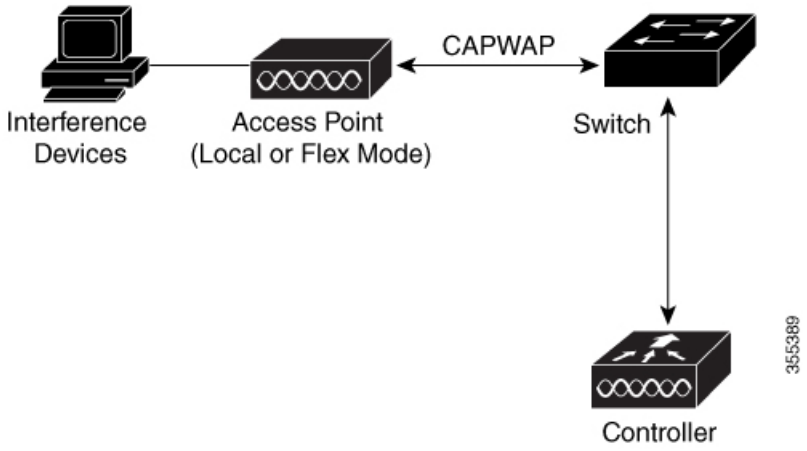
The key components involved in the process are:

- Cisco CleanAir-enabled AP: Collects information about Wi-Fi interference sources and processes it. The AP collects and sends the Air Quality Report (AQR) and Interference Device Report (IDR) to the controller .
- Controller: Controls and configures CleanAir-capable access points, collects and processes spectrum data, provides user interfaces (GUI and CLI), and detects, merges, and mitigates interference devices. The controller also detects, merges, and mitigates interference devices using RRM TPC and DCA For details, see Interference Device Merging.

CleanAir technology ensures wireless networks remain reliable by automatically detecting sources of interference and providing tools for administrators to analyze and respond to RF threats.

Workflow

Figure 34: Cisco CleanAir Solution



These stages describe the operation of Cisco CleanAir components in the system.

1. Configuration: Configures Cisco CleanAir capabilities on the APs.
2. Detection: Cisco CleanAir-enabled APs continuously scan the wireless environment for interference sources and assess air quality.
3. Reporting: APs generate reports such as Air Quality Reports (AQRs) and Interference Device Reports (IDRs), which summarize detected interference and air quality indexes.
4. Aggregation: Wireless controllers collect AQRs and IDRs from all APs, process the data, and store it in relevant databases.
5. Analysis and Response: Controllers use built-in mechanisms (like Radio Resource Management) to mitigate interference and maintain optimal RF conditions.
6. Advanced Services: Cisco Spaces or CMX tracks interference devices over time and location, merges data from multiple controllers, and provides comprehensive threat detection with adaptive Wireless Intrusion Prevention System (WIPS).
7. Specialized Troubleshooting (optional): For in-depth RF analysis, an administrator can connect a CleanAir-enabled AP directly to a PC running Cisco Spectrum Expert.

Result

The Cisco CleanAir system delivers end-to-end automated detection, analysis, and mitigation of wireless interference, enabling proactive network management and high-quality wireless performance.

Prerequisites for CleanAir

Details the prerequisites required for deploying Cisco CleanAir, outlining supported configurations and hardware, and specifies relevant operational restrictions for successful implementation.

You can configure Cisco CleanAir only on CleanAir-enabled access points.

Only Cisco CleanAir-enabled access points in these access point modes can perform Cisco CleanAir spectrum monitoring:

- **Local**: In this mode, each Cisco CleanAir-enabled access point radio provides air quality and interference detection reports for the current operating channel only. An AP measures air quality and interference only when it is not transmitting Wi-Fi frames. When channel utilization is high, CleanAir detects less interference.
- **FlexConnect**: When a FlexConnect access point is connected to the controller, its Cisco CleanAir functionality is identical to local mode.

- **Monitor**: When Cisco CleanAir is enabled in monitor mode, the access point provides air quality and interference detection reports for all monitored channels.
  - **All**: All channels
  - **DCA**: Channel selection governed by the DCA list
  - **Country**: All channels are legal within a regulatory domain

Restrictions for CleanAir

Restrictions for CleanAir.

Access points in monitor mode do not transmit Wi-Fi traffic or 802.11 packets. They are excluded from radio resource management (RRM) planning and are not included in the neighbor AP list. IDR clustering depends on the device 's ability to detect neighboring in-network APs. Correlating interference device detections from multiple APs is limited between monitor-mode APs.

CleanAir is not supported where the channel width is 160 MHz.

Difference between Spectrum Intelligence and Cisco CleanAir

Difference betwen

Feature	Spectrum Intelligence	Cisco CleanAir
Noise identification	Reports broad, often vague RF noise without pinpointing sources	Precisely identifies interference type, source, potential impact to a WLAN
Noise measurement method	Relies on long-term averaging, which smooths out important details and reduces resolution of measurements	Captures real-time events with full granularity
Response style	Primarily reacts after interference is noticed	Enables proactive, context-aware mitigation allowing for intelligent, proactive decisions.
Transient interference handling	Averaging measurements reduces resolution of measurements, and disruptive signals do not appear to need mitigation.	Flags brief, spontaneous interference events immediately

Cisco CleanAir access points can detect and report severity of the interference .

Spectrum event-driven RRM is one such mitigation strategy.

> **Note**
>
> Spectrum event-driven RRM can be triggered only by Cisco CleanAir-enabled access points in local mode.

Persistence interference

This section explains persistent device avoidance, a feature in Cisco CleanAir that ensures sustained interference mitigation from devices like microwave ovens.

A persistent device avoidance is a wireless interference mitigation feature that

- identifies and continuously tracks intermittent interference sources such as microwave ovens,
- maintains channel avoidance for affected APs even if the interference is not constantly present, and
- automatically updates Radio Resource Management (RRM) whenever the interfering device or related access points are moved.

For example, if a microwave oven operates in a break room only for a few minutes at a time, Cisco CleanAir identifies the microwave as an interference source, locates the affected portion of the band and APs most severely affected, and directs RRM to avoid those channels persistently. Even if the microwave is not active all day, the system maintains avoidance as long as the device is periodically detected. If you move either the microwave oven or the surrounding access points, the algorithm updates RRM automatically

Classes of persistent interference sources

Certain devices, once detected, are likely to remain in place and may cause ongoing interference. Cisco CleanAir can bias the affected channel in Radio Resource Management (RRM) so that CleanAir "remembers" the potential for interference. See Radio Resource Management White Paper.

These device classes qualify as persistent interference sources:

- Microwave ovens
- Outdoor Ethernet bridges
- WiMax fixed and mobile devices
- Motorola Canopy systems

Special case: Bluetooth devices

Bluetooth devices are detected and reported as interference only when actively transmitting. They often enter power-save modes, so interference may only be noticed during active streaming of data or voice.

Persistent devices detection

Persistent devices detection.

CleanAir-capable Monitor Mode AP collects information about persistent devices on all configured channels and store the information in controller . Local or Bridge mode AP detects interference devices on the serving channels only.

Persistent device avoidance

Persistent device avoidance.

When a Persistent Device (PD) is detected in the CleanAir module, it is reported to the RRM module on the MA. This information is used in the channel selection by the subsequent EDRRM Event Driven RRM (ED-RRM) signal sent to the RRM module.

Spontaneous interference

Spontaneous interference is interference that appears suddenly on a network and may jam one or more channels. With Cisco CleanAir spectrum, event-driven RRM, you can set a threshold for air quality, abbreviated as AQ. If air quality drops below this threshold, the system immediately changes the affected AP's channel. Although most RF management systems avoid interference, they require time for information to spread before channels update. Cisco CleanAir uses AQ measurements to continually check the spectrum and can switch channels within **30 seconds**.

For example, if your AP detects interference from a device such as a video camera, it switches channels within **30 seconds** after the interference begins. Cisco CleanAir also helps you identify and locate the interference source so you can take permanent steps to resolve the issue later.

EDRRM and AQR update mode

This section explains EDRRM and AQR update modes, which allow wireless APs to respond rapidly to interference by bypassing standard Radio Resource Management intervals..

An EDRRM and AQR update mode is a wireless network feature set that

- enables APs experiencing severe interference (“in distress”) to bypass normal RRM intervals and immediately change channels,
- requires CleanAir technology for monitoring and reporting classified interference devices, and
- provides fast action and remote configuration through control messages and local spectrum management.
- **EDRRM (Event Driven Radio Resource Management):** A function that allows an AP to react instantly to excessive interference by changing channels immediately instead of waiting for scheduled RRM intervals.
- **AQR (Air Quality Reporting) update mode**: A process by which CleanAir APs monitor air quality and report classified interference devices every 15 minutes; when interference causes substantial air quality degradation, EDRRM is triggered.

If a classified interference device severely degrades the air quality on an active channel, EDRRM triggers an immediate channel change so that no clients are stranded on an unusable channel.

Think of a city’s scheduled traffic lights as the normal RRM intervals, allowing cars (data) to move in an orderly, timed way. Normally, everyone waits for the green light, moving only at the set intervals. EDRRM is like having an emergency override for ambulances (APs in distress): if there’s a medical emergency (severe interference), the control center can change the lights to green immediately—regardless of the normal schedule—so the ambulance can take an alternate, clear route (new channel). Air Quality Reporting (AQR) acts like real-time traffic cameras detecting and reporting sudden blockages (classified interference devices) so that the emergency override can be triggered if road conditions (channel quality) deteriorate unexpectedly.

Restrictions for EDRRM and AQR update mode

- EDRRM is not enabled by default; you must first enable CleanAir and then enable EDRRM.
- If an interfering device is operating on an active channel and causes enough AQ degradation to trigger an EDRRM, then no clients will be able to use that channel or the AP. You must remove the AP from the channel.

Configure CleanAir

Explains step-by-step procedures to enable and configure CleanAir on 2.4-GHz and 5-GHz bands using both GUI and CLI, set up air quality alarms, configure interference reporting, EDRRM, Persistent Device Avoidance, and manage CleanAir on controllers and access points.

Enable CleanAir for the 2.4-GHz band (GUI)

Learn how to activate CleanAir spectrum intelligence for 2.4-GHz radios using the web interface to monitor and mitigate wireless interference.

Enable CleanAir on 2.4-GHz band radios for real-time interference detection.

Use this procedure to activate Cisco CleanAir technology on APs operating in the 2.4-GHz spectrum using the GUI.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **CleanAir**.

2. On the **CleanAir** window, click the **2.4 GHz Band** > **General** tab.
3. Check the **Enable CleanAir** checkbox.
4. Click **Apply**.

CleanAir is enabled on all selected 2.4-GHz radios, and spectrum analysis begins for those devices.

Enable CleanAir for the 2.4-GHz band (CLI)

This topic provides step-by-step procedure to configure CleanAir on the controller using commands.

Enable CleanAir technology on APs operating in the 2.4-GHz band radios for real-time interference detection using commands.

Procedure

Enable the CleanAir feature on the 802.11b network.

Example

```
Device(config)#ap dot11 24ghz cleanair
Device(config)#no ap dot11 24ghz cleanair
```

Run the **no** form of this command to disable CleanAir on the 802.11b network.

CleanAir is enabled on all selected 2.4-GHz radios, and spectrum analysis begins for those devices.

Configure a CleanAir alarm for 2.4-GHz air-quality and devices (CLI)

Configure CleanAir alarms for specific air quality thresholds and device interference types on 2.4-GHz radios using the commands.

Set up alarms that notify you when air quality drops or when certain device types are detected on the 2.4-GHz band using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the alarm for the threshold value for air-quality for all the 2.4-GHz devices.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair alarm air-quality threshold
   threshold_value
   ```

   Add the **no** form of this command to disable the alarm.

3. Configure the alarm for the 2.4-GHz devices using this command.

Example

```
Device(config)# ap dot11 24ghz cleanair alarm device {bt-discovery | bt-link
 | canopy | cont-tx | dect-like | fh | inv | jammer | mw-oven | nonstd |
report | superag | tdd-tx | video | wimax-fixed | wimax-mobile | xbox | zigbee}
```

Add the **no** form command to disable the alarm.

- bt-discovery: Bluetooth Discovery and bt-link: Bluetooth Link.
- canopy: Canopy devices and cont-tx: Continuous Transmitter.
- dect-like: Digital Enhanced Cordless Communication (DECT)-like phone and fh: 802.11 frequency hopping devices.
- inv: Devices using spectrally inverted WiFi signals and jammer: Jammer.
- mw-oven: Microwave oven and nonstd: Devices using non standard Wi-Fi channels.
- report: Interference device reporting and superag: 802.11 SuperAG devices.
- tdd-tx: TDD Transmitter and video: Video cameras.
- wimax-fixed: WiMax Fixed and wimax-mobile: WiMax Mobile.
- xbox: Xbox and zigbee: 802.15.4 devices.

4. Return to the privileged EXEC mode.

Example

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure interference reporting for a 2.4-GHz device (GUI)

Configure interference reporting for a 2.4-GHz device using GUI.

Configure which types of interference are detected by a 2.4-GHz device using the CleanAir GUI. This task guides you through selecting and applying specific interference types to monitor.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **CleanAir**.
2. Click the **2.4 GHz Band** tab.
3. Choose the interference types and add them to the **Interference Types to detect** section.

   These interference types are available:

   - BLE Beacon: Bluetooth low energy beacon and Bluetooth Discovery.
   - Bluetooth Link and Canopy.
   - Continuous Transmitter and DECT-like Phone: Digital Enhanced Cordless Technology phone.
   - 802.11 FH: 802.11 frequency hopping device and WiFi Inverted: Device using spectrally inverted Wi-Fi signals.
   - Jammer and Microwave Oven.

- WiFi Invalid Channel: Device using nonstandard Wi-Fi channels and TDD Transmitter.
- Video Camera and SuperAG: 802.11 SuperAG device.
- WiMax Mobile and WiMax Fixed.
- 802.15.4, Microsoft Device and SI_FHSS.

4. Click **Apply**.

Configure interference reporting for a 2.4-GHz device (CLI)

Configure interference reporting for a 2.4-GHz device using commands.

Configure a 2.4-GHz device to report interference using commands. This process enables devices to detect and provide information about various sources of wireless interference.

Procedure

1. Configure the 2.4-GHz interference devices to report to the device using the command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair device {ble-beacon | bt-discovery |
   bt-link | canopy | cont-tx | dect-like | fh | inv | jammer | mw-oven | nonstd
    | report | superag | tdd-tx | video | wimax-fixed | wimax-mobile | xbox |
   zigbee | alarm}
   ```

   Run the **no** form of this command to disable the configuration.

   These are the list of the keyword descriptions:

   - ble-beacon: Bluetooth low energy beacon.
   - bt-discovery: Bluetooth discovery and bt-link: Bluetooth link.
   - canopy: Canopy device and cont-tx: Continuous transmitter.
   - dect-like: Digital Enhanced Cordless Communication-like phone and fh: 802.11-frequency hopping device.
   - inv: Device using spectrally inverted Wi-Fi signals and jammer: Jammer.
   - mw-oven: Microwave oven, nonstd: Device using nonstandard Wi-Fi channels and report: Interference device reporting.
   - superag: 802.11 SuperAG device, tdd-tx: TDD transmitter and video: Video camera.
   - wimax-fixed: WiMax Fixed and wimax-mobile: WiMax Mobile.
   - microsoft xbox: Microsoft Xbox device and zigbee: 802.15.4 device.

2. Return to the privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Enable CleanAir for the 5-GHz band (GUI)

Learn how to enable CleanAir for the 5-GHz band to improve wireless interference detection and spectrum analysis.

Enable CleanAir to improve 5-GHz wireless interference detection and spectrum analysis using the GUI.

Perform this task when you need enhanced monitoring and automated mitigation of interference on 5-GHz wireless networks.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **CleanAir**.
2. On the **CleanAir** page, choose the **5 GHz Band** > **General** tab.
3. Check the **Enable CleanAir** checkbox.
4. Click **Apply**.

CleanAir is enabled for the 5-GHz band, enhancing the detection and mitigation of wireless interference.

Enable CleanAir for the 5-GHz band (CLI)

Configure CleanAir on a 5-GHz 802.11a network to enhance wireless interference detection and mitigation using the commands.

Enable the CleanAir feature to monitor and mitigate wireless interference on 5-GHz wireless networks using commands.

Procedure

Enable the CleanAir feature on a 802.11a network.

Example

```
Device(config)# ap dot11 5ghz cleanair
```

Run the **no** form of this command to disable CleanAir on the 802.11a network.

CleanAir is enabled for the 5-GHz band, enhancing the detection and mitigation of wireless interference.

Configure interference reporting for a 5-GHz device (GUI)

Learn how to set up interference reporting on your 5-GHz device to identify and detect various wireless interference sources.

Enable detection and reporting of selected interference types for better network performance.

Use this procedure when you need to monitor wireless interference on 5-GHz band devices using CleanAir technology.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **CleanAir**.

2. Click the **5 GHz Band** tab.
3. Choose the interference types and add them to the **Interference Types to detect** section.

   These interference types are available:

   - Canopy
   - Continuous Transmitter
   - DECT-like Phone–Digital Enhanced Cordless Technology phone
   - 802.11 FH–802.11 frequency hopping device
   - WiFi Inverted–Device using spectrally inverted Wi-Fi signals
   - Jammer
   - WiFi Invalid Channel–Device using nonstandard Wi-Fi channels
   - SuperAG–802.11 SuperAG device
   - TDD Transmitter
   - WiMax Mobile
   - WiMax Fixed
   - Video Camera

4. Click **Apply**.

Your 5-GHz device now detects and reports the selected interference types.

Configure interference reporting for a 5-GHz device (CLI)

Configure interference reporting for a 5-GHz device using commands.

Procedure

---

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a 5-GHz interference device to report to the **ap dot11 5ghz cleanair device{canopy | cont-tx | dect-like | inv | jammer | nonstd | report | superag | tdd-tx | video | wimax-fixed | wimax-mobile}** command.

   Example

   ```
   Device(config)#ap dot11 5ghz cleanair device canopy
   Device(config)#ap dot11 5ghz cleanair device cont-tx
   Device(config)#ap dot11 5ghz cleanair device dect-like
   Device(config)#ap dot11 5ghz cleanair device inv
   ```

```
Device(config)#ap dot11 5ghz cleanair device jammer
Device(config)#ap dot11 5ghz cleanair device nonstd

Device(config)#ap dot11 5ghz cleanair device report
Device(config)#ap dot11 5ghz cleanair device superag
Device(config)#ap dot11 5ghz cleanair device tdd-tx
Device(config)#ap dot11 5ghz cleanair device video
Device(config)#ap dot11 5ghz cleanair device wimax-fixed
Device(config)#ap dot11 5ghz cleanair device wimax-mobile
Device(config)#ap dot11 5ghz cleanair device si_fhss
Device(config)#ap dot11 5ghz cleanair device alarm
```

Device. Run the **no** form of this command to disable interference device reporting.

This is a list of the keyword descriptions:

- **canopy**–Canopy device
- **cont-tx**–Continuous transmitter
- **dect-like**–Digital Enhanced Cordless Communication-like phone
- **fh**–802.11-frequency hopping device
- **inv**–Device using spectrally-inverted Wi-Fi signals
- **jammer**–Jammer
- **nonstd**–Device using nonstandard Wi-Fi channels
- **superag**–802.11 SuperAG device
- **tdd-tx**–TDD transmitter
- **video**–Video camera
- **wimax-fixed**–WiMax fixed
- **wimax-mobile**–WiMax mobile

3. Return to the privileged EXEC mode.

Example

```
Device(config)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure event driven RRM for a CleanAir event (GUI)

Configure event driven Radio Resource Management (RRM) to automatically adjust wireless channels in response to interference detected by CleanAir-enabled access points.

Configure event driven Radio Resource Management (RRM) to respond to CleanAir-detected interference.

Use this feature to optimize channel assignment automatically when a CleanAir-enabled access point detects significant interference.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **RRM**.
   The **Radio Resource Management** window is displayed.
2. Click the **DCA** tab.
3. In the **Event Driven RRM** section, check the **EDRRM** check box to run RRM when CleanAir-enabled AP detects a significant level of interference.
4. Configure the **Sensitivity Threshold** level at which RRM has to be invoked from the following options:
   - **Low**: Represents a decreased sensitivity to changes in the environment and its value is set at 35.
   - **Medium**: Represents medium sensitivity to changes in the environment at its value is set at 50.
   - **High**: Represents increased sensitivity to changes in the environment at its value is set at 60.
   - **Custom**: If you choose this option, you must specify a custom value in the **Custom Threshold** box.
5. To configure rogue duty cycle, check the **Rogue Contribution** check box and then specify the **Rogue Duty-Cycle** in terms of percentage. The default value of rogue duty cycle is 80 percent.

   Note

   Rogue Contribution is a new component in ED-RRM functionality. It enables ED-RRM to respond to Rogue Channel Utilization, which operates independently from CleanAir metrics. Rogue Duty Cycle is derived from standard off-channel RRM metrics and triggers a channel change when neighboring rogue interference is detected. Since this process relies on RRM metrics rather than CleanAir, the timing depends on normal 180-second off-channel intervals, with a maximum delay of three minutes. Rogue Contribution is configured separately from CleanAir ED-RRM and is disabled by default. This feature allows each AP to respond to Wi-Fi interference originating outside the local network, with measurements taken at every individual AP.

   Rogue Contribution is a new component in ED-RRM functionality. It enables ED-RRM to respond to detected Rogue Channel Utilization, which functions separately from CleanAir metrics.

6. Save the configuration.

The controller now automatically triggers RRM based on CleanAir or optionally, rogue channel utilization events.

Configure EDRRM for a CleanAir event (CLI)

Use this procedure to enable and configure Event-Driven Radio Resource Management (EDRRM) CleanAir event detection on an access point through the CLI.

Enable automated, event-driven dynamic RRM adjustments in response to CleanAir events.

EDRRM allows the AP to adjust its channel management dynamically in response to detected interference or CleanAir events. This ensures better wireless performance and mitigation of interference.

Before you begin

Verify the AP supports CleanAir and EDRRM features.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# conf t
   ```

   Run the **no** form of this command to disable EDRRM.

2. Enable EDRRM CleanAir event.

   Example

   ```
   Device (config)#ap dot11 24ghz rrm channel cleanair-event
   Device(config)#no ap dot11 24ghz rrm channel cleanair-event
   ```

   Run the **no** form of this command to disable EDRRM.

3. Configure the EDRRM sensitivity of the CleanAir event.

   Example

   ```
   Device(config)#ap dot11 24ghz rrm channel cleanair-event sensitivity high
   ```

   This a list of the keyword descriptions:

   - Custom–Specifies custom sensitivity to non-Wi-Fi interference as indicated by the AQ value.
   - High–Specifies the most sensitivity to non-Wi-Fi interference as indicated by the AQ value.
   - Low–Specifies the least sensitivity to non-Wi-Fi interference as indicated by the AQ value.
   - Medium–Specifies medium sensitivity to non-Wi-Fi interference as indicated by the AQ value.

EDRRM CleanAir event detection and response is enabled with your selected sensitivity, improving dynamic channel management during interference events.

Verify CleanAir support on connected APs

Lists the methods for determining CleanAir support status on connected access points.

Provide information and methods to help users identify which connected access points support Cisco CleanAir technology, enabling informed network management and planning.

To verify CleanAir support on connected APs, use this command:

```
Device# show ap dot11 24ghz cleanair summary
                                                Spectrum  Spectrum     Spectrum
    Spectrum
```

```
AP Name             MAC Address         Slot  AP Type   Admin State  Oper State
  Analysis
-------------------------------------------------------------------------------
ap9177               1057.252a.04e0         0  CA-Pro    Enabled      Up
    Disabled
ap9174               940d.4b9d.6e20         0  CA-Pro    Enabled      Up
    Disabled
ap9176               ecf4.0caf.4140         0  CA-Pro    Enabled      Up
    Disabled
ap9176-desk          ecf4.0caf.8200         0  CA-Pro    Enabled      Up
    Disabled
ap9130               087b.87fc.7f40         0  CA        Enabled      Up
    Disabled
ap9105               e4a4.1c4b.39c0         0  SI        Disabled     Down
    NA
```

CleanAir Pro scanning

Describes CleanAir Pro scanning features, detailing how to enable scanning via CLI, monitor scanning statistics through GUI, and verify operational details for enhanced RF interference detection and analysis.

A CleanAir Pro feature is a wireless monitoring capability that

- monitors and reports different categories of non-Wi-Fi interference across supported 2.4-GHz, 5-GHz, and 6-GHz band frequency bands
- generates detailed Interference Device Reports (IDRs) including interferer type, severity, and impacted channels, and
- calculates air quality (AQ) metrics for each dynamic channel assignment (DCA) channel to support intelligent channel management by the controller.

Table 166: Feature history for CleanAir Pro scanning

Release	Feature	Feature Information
Cisco IOS XE Dublin 17.10.1	CleanAir Support for 6-GHz	The CleanAir is enabled in 6-GHz band radio of AP only if CleanAir is enabled globally in 6-GHz band in the controller and 6-GHz radio of individual AP. Cisco IOS XE Cupertino 17.9.1 supports 6-GHz only for spectral analysis on Cisco Catalyst Center. IDR and AQ are not supported for the 6-GHz band in Cisco IOS XE Cupertino 17.9.1.
Cisco IOS XE Cupertino 17.9.1	CleanAir Pro Scanning	The CleanAir Pro Scanning feature monitors and reports the different categories of non-Wi-Fi interference in the 2.4-GHz and 5-GHz bands.

Controller behavior for interferer and channel monitoring

The controller manages interferer information and channel selection using these methods:

- **Interferer database management**: The controller maintains a database of active interferers reported by each access point (AP) on the network.

- **Interferer merging**: If the same interferer is detected by multiple APs, the controller merges these records so the interferer is tracked as a single entity across the network.
- **Air quality tracking**: The controller monitors air quality metrics for each channel. These metrics are used during dynamic channel assignment (DCA) to select optimal channels for each AP.

> **Note**
>
> The CleanAir Pro Scanning feature is applicable only for APs with CleanAir Pro-supported radios.

Channel-list configuration command keywords for CleanAir Pro scanning

CleanAir Pro Scanning scans channels in the slots or bands currently enabled on an AP. Here are certain commands used to enable scannig.

- **ap dot11 rrm monitor channel-list dca**: Scans only channels enabled under DCA.
- **ap dot11 rrm monitor channel-list country**: Scans all regulatory channels permitted in the site's country setting.
- **ap dot11 rrm monitor channel-list country all**: Scans every channel available to the radio, without regard to regulatory constraints.

Contents and event types in IDR

APs detect non-Wi-Fi interferers and report this data to the controller through Interference Device Report (IDR) such as

- duty cycle,
- Received Signal Strength Indicator (RSSI) in dBm, and
- a calculated Severity metric.

The controller receives IDR messages and uses event types to track the status of each interferer:

- **UP**: The interferer is first detected.
- **UPDT**: The interferer's status is updated.
- **DOWN**: The interferer is no longer observed.

The controller maintains a list of interferers along with the channels affected for each AP. Within an AP, interferers are merged if they share the same channel, RSSI, and device signature. Across multiple APs, a controller merges interferers reported as the same type.

These processes enable accurate tracking and consolidation of interferer data throughout the network.

Air quality index reports

The Air Quality Index (AQ) metric quantifies interference levels detected by each access point (AP) in a Cisco wireless network.

AQ calculation: AQ starts at 100 (optimal) for each AP and is decremented by the cumulative severity values of all detected interference sources.

Air Quality Metric Calculation

For example, if three Bluetooth devices are reported by an AP, each with a severity of two, this results in an overall cell AQ of 94 (2x3 BT = 6, 100 – 6 = AQ of 94).

Enable CleanAir Pro scan features (CLI)

Learn how to enable CleanAir Pro scan features using the CLI for 2.4, 5, and 6 GHz radios.

Enable spectrum monitoring and interference detection across all supported wireless radios.

Use this procedure when you want to configure CleanAir Pro scanning and alarms on Cisco wireless devices using CLI.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the CleanAir features for the 2.4-GHz , 5-GHz, or 6-GHz radios using the **ap dot11** {**24ghz** | **5ghz** | **6ghz**} **cleanair** command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair
   ```

3. Configure CleanAir alarm for air quality in the 2.4-GHz, 5-GHz, or 6-GHz radiosusing the **ap dot11** {**24ghz** | **5ghz** | **6ghz**} **cleanair alarm air-quality** command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair alarm air-quality
   ```

4. Configure the air quality threshold value of CleanAir alarm in the 2.4-GHz , 5-GHz, or 6-GHz radios using the **ap dot11** {**24ghz** | **5ghz** | **6ghz**} **cleanair alarm air-quality threshold** *threshold-value* command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair alarm air-quality threshold 25
   ```

   The valid range is between 1 and 100.

5. Configure the continuous transmitter as the interference device CleanAir alarm in the 2.4-GHz , 5-GHz, or 6-GHz radios using the **ap dot11** {**24ghz** | **5ghz** | **6ghz**} **cleanair alarm device cont-tx** command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair alarm device cont-tx
   ```

6. Configure the air quality alarm on exceeding unclassified category severity using the **ap dot11** {**24ghz** | **5ghz** | **6ghz**} **cleanair alarm unclassified** command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair alarm unclassified
   ```

7. Configure the air quality alarm on exceeding unclassified category severity threshold using the **ap dot11** {**24ghz** | **5ghz** | **6ghz**} **cleanair alarm unclassified threshold** *threshold-value*command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair alarm unclassified threshold 15
   ```

   The valid range is between 1 and 100.

8. Configure continuous transmitter as the CleanAir interference device type using the **ap dot11** {**24ghz** | **5ghz** | **6ghz**} **cleanair device cont-tx** command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair device cont-tx
   ```

CleanAir Pro scan features are active, with alarms configured according to your settings.

Monitor CleanAir Pro statistics (GUI)

Use this procedure to monitor CleanAir Pro statistics and interference reports using the graphical user interface.

Review CleanAir Pro statistics for air quality, interference, and radio channels across your wireless network.

This task helps you assess wireless spectrum health, identify interference sources, and check air quality metrics for monitored access points.

Procedure

---

1. Choose **Monitoring** > **Wireless** > **CleanAir Statistics**.
   The **CleanAir Statistics** window is displayed.
2. Click the **2.4 GHz Band**, **5 GHz Band**, or **6 GHz Band** tab.
   The **CleanAir Interference Devices** window is selected by default. You can monitor and detect the cluster IDs, the interferer type, severity, the affected channels, and so on, for the listed APs.
3. Click the listed devices under the **CleanAir Interference Devices** tab to view the **CleanAir Interference Charts** that displays the **AQ Graph** and the **Interference Power**.
4. Click the **Air Quality** tab to monitor the channel, the average and minimum AQ, number of interferers, the time at which the interference was detected, and the spectrum AP type.
5. Click the **Worst Air Quality Report** tab to view the AQ report, with details of the AP that reported the worst AQ, the radio channel number with the worst-reported air quality, the minimum and the average AQ index, the interference device count, and the spectrum AP type.

You have reviewed CleanAir Pro statistics to monitor air quality, interference, and device-specific spectrum data across selected wireless bands.

Verify CleanAir Pro scanning details

Verify CleanAir Pro scanning details using GUI.

To view the CleanAir Air Quality (AQ) data, run this command:

```
Device# show ap dot11 {24ghz | 5ghz | 6ghz} cleanair air-quality summary
```

To view the CleanAir Air Quality (AQ) worst data, run this command:

```
Device# show ap dot11 {24ghz | 5ghz | 6ghz} cleanair air-quality worst
```

To view the CleanAir device cluster information, run this command:

```
Device# show ap dot11 {24ghz | 5ghz | 6ghz} cleanair device cluster cluster-id
```

To view the CleanAir interferers of a device type, run this command:

```
Device# show ap dot11 {24ghz | 5ghz | 6ghz} cleanair device type
```

To view the CleanAir configuration for a specific AP, run this command:

```
Device# show ap name ap-name dot11 {24ghz | 5ghz | 6ghz} cleanair air-quality
Device# show ap name ap-name dot11 {24ghz | 5ghz | 6ghz} cleanair device
```

To view the continuous transmitter as the CleanAir interference device type, run this command:

```
Device# show ap dot11 6ghz cleanair device type cont-tx
```

Configuration examples for CleanAir

Configuration examples for CleanAir.

This example shows how to enable CleanAir on the 2.4-GHz band and an AP operating in the channel:

```
Device#configure terminal
Device(config)#ap dot11 24ghz cleanair
Device(config)#exit
Device#ap name TAP1 dot11 24ghz cleanair
Device#end
```

This example shows how to enable an EDRRM CleanAir event in the 2.4-GHz band and configure high sensitivity to non-Wi-Fi interference:

```
Device#configure terminal
Device(config)#ap dot11 24ghz rrm channel cleanair-event
Device(config)#ap dot11 24ghz rrm channel cleanair-event sensitivity high
Device(config)#end
```

This example shows how to enable an AP in the monitor mode:

```
Device#ap name <ap-name> mode monitor
```

CleanAir Pro scanning

Describes CleanAir Pro scanning features, detailing how to enable scanning via CLI, monitor scanning statistics through GUI, and verify operational details for enhanced RF interference detection and analysis.

A CleanAir Pro feature is a wireless monitoring capability that

- monitors and reports different categories of non-Wi-Fi interference across supported 2.4-GHz, 5-GHz, and 6-GHz band frequency bands
- generates detailed Interference Device Reports (IDRs) including interferer type, severity, and impacted channels, and

- calculates air quality (AQ) metrics for each dynamic channel assignment (DCA) channel to support intelligent channel management by the controller.

Table 167: Feature history for CleanAir Pro scanning

Release	Feature	Feature Information
Cisco IOS XE Dublin 17.10.1	CleanAir Support for 6-GHz	The CleanAir is enabled in 6-GHz band radio of AP only if CleanAir is enabled globally in 6-GHz band in the controller and 6-GHz radio of individual AP. Cisco IOS XE Cupertino 17.9.1 supports 6-GHz only for spectral analysis on Cisco Catalyst Center. IDR and AQ are not supported for the 6-GHz band in Cisco IOS XE Cupertino 17.9.1.
Cisco IOS XE Cupertino 17.9.1	CleanAir Pro Scanning	The CleanAir Pro Scanning feature monitors and reports the different categories of non-Wi-Fi interference in the 2.4-GHz and 5-GHz bands.

Controller behavior for interferer and channel monitoring

The controller manages interferer information and channel selection using these methods:

- **Interferer database management**: The controller maintains a database of active interferers reported by each access point (AP) on the network.
- **Interferer merging**: If the same interferer is detected by multiple APs, the controller merges these records so the interferer is tracked as a single entity across the network.
- **Air quality tracking**: The controller monitors air quality metrics for each channel. These metrics are used during dynamic channel assignment (DCA) to select optimal channels for each AP.

Note

The CleanAir Pro Scanning feature is applicable only for APs with CleanAir Pro-supported radios.

Channel-list configuration command keywords for CleanAir Pro scanning

CleanAir Pro Scanning scans channels in the slots or bands currently enabled on an AP. Here are certain commands used to enable scannig.

- **ap dot11 rrm monitor channel-list dca**: Scans only channels enabled under DCA.
- **ap dot11 rrm monitor channel-list country**: Scans all regulatory channels permitted in the site's country setting.
- **ap dot11 rrm monitor channel-list country all**: Scans every channel available to the radio, without regard to regulatory constraints.

Contents and event types in IDR

APs detect non-Wi-Fi interferers and report this data to the controller through Interference Device Report (IDR) such as

- duty cycle,
- Received Signal Strength Indicator (RSSI) in dBm, and
- a calculated Severity metric.

The controller receives IDR messages and uses event types to track the status of each interferer:

- **UP**: The interferer is first detected.
- **UPDT**: The interferer's status is updated.
- **DOWN**: The interferer is no longer observed.

The controller maintains a list of interferers along with the channels affected for each AP. Within an AP, interferers are merged if they share the same channel, RSSI, and device signature. Across multiple APs, a controller merges interferers reported as the same type.

These processes enable accurate tracking and consolidation of interferer data throughout the network.

Air quality index reports

The Air Quality Index (AQ) metric quantifies interference levels detected by each access point (AP) in a Cisco wireless network.

AQ calculation: AQ starts at 100 (optimal) for each AP and is decremented by the cumulative severity values of all detected interference sources.

Air Quality Metric Calculation

For example, if three Bluetooth devices are reported by an AP, each with a severity of two, this results in an overall cell AQ of 94 (2x3 BT = 6, 100 – 6 = AQ of 94).

Enable CleanAir Pro scan features (CLI)

Learn how to enable CleanAir Pro scan features using the CLI for 2.4, 5, and 6 GHz radios.

Enable spectrum monitoring and interference detection across all supported wireless radios.

Use this procedure when you want to configure CleanAir Pro scanning and alarms on Cisco wireless devices using CLI.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the CleanAir features for the 2.4-GHz , 5-GHz, or 6-GHz radios using the **ap dot11** {**24ghz** | **5ghz** | **6ghz**} **cleanair** command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair
   ```

3. Configure CleanAir alarm for air quality in the 2.4-GHz, 5-GHz, or 6-GHz radiosusing the **ap dot11 {24ghz | 5ghz | 6ghz} cleanair alarm air-quality** command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair alarm air-quality
   ```

4. Configure the air quality threshold value of CleanAir alarm in the 2.4-GHz , 5-GHz, or 6-GHz radios using the **ap dot11 {24ghz | 5ghz | 6ghz} cleanair alarm air-quality threshold** *threshold-value* command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair alarm air-quality threshold 25
   ```

   The valid range is between 1 and 100.

5. Configure the continuous transmitter as the interference device CleanAir alarm in the 2.4-GHz , 5-GHz, or 6-GHz radios using the **ap dot11 {24ghz | 5ghz | 6ghz} cleanair alarm device cont-tx** command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair alarm device cont-tx
   ```

6. Configure the air quality alarm on exceeding unclassified category severity using the **ap dot11 {24ghz | 5ghz | 6ghz} cleanair alarm unclassified** command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair alarm unclassified
   ```

7. Configure the air quality alarm on exceeding unclassified category severity threshold using the **ap dot11 {24ghz | 5ghz | 6ghz} cleanair alarm unclassified threshold** *threshold-value*command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair alarm unclassified threshold 15
   ```

   The valid range is between 1 and 100.

8. Configure continuous transmitter as the CleanAir interference device type using the **ap dot11 {24ghz | 5ghz | 6ghz} cleanair device cont-tx** command.

   Example

   ```
   Device(config)# ap dot11 24ghz cleanair device cont-tx
   ```

CleanAir Pro scan features are active, with alarms configured according to your settings.

Monitor CleanAir Pro statistics (GUI)

Use this procedure to monitor CleanAir Pro statistics and interference reports using the graphical user interface.

Review CleanAir Pro statistics for air quality, interference, and radio channels across your wireless network.

This task helps you assess wireless spectrum health, identify interference sources, and check air quality metrics for monitored access points.

Procedure

1. Choose **Monitoring** > **Wireless** > **CleanAir Statistics**.
   The **CleanAir Statistics** window is displayed.
2. Click the **2.4 GHz Band**, **5 GHz Band**, or **6 GHz Band** tab.
   The **CleanAir Interference Devices** window is selected by default. You can monitor and detect the cluster IDs, the interferer type, severity, the affected channels, and so on, for the listed APs.
3. Click the listed devices under the **CleanAir Interference Devices** tab to view the **CleanAir Interference Charts** that displays the **AQ Graph** and the **Interference Power**.
4. Click the **Air Quality** tab to monitor the channel, the average and minimum AQ, number of interferers, the time at which the interference was detected, and the spectrum AP type.
5. Click the **Worst Air Quality Report** tab to view the AQ report, with details of the AP that reported the worst AQ, the radio channel number with the worst-reported air quality, the minimum and the average AQ index, the interference device count, and the spectrum AP type.

You have reviewed CleanAir Pro statistics to monitor air quality, interference, and device-specific spectrum data across selected wireless bands.

Verify CleanAir Pro scanning details

Verify CleanAir Pro scanning details using GUI.

To view the CleanAir Air Quality (AQ) data, run this command:

```
Device# show ap dot11 {24ghz | 5ghz | 6ghz} cleanair air-quality summary
```

To view the CleanAir Air Quality (AQ) worst data, run this command:

```
Device# show ap dot11 {24ghz | 5ghz | 6ghz} cleanair air-quality worst
```

To view the CleanAir device cluster information, run this command:

```
Device# show ap dot11 {24ghz | 5ghz | 6ghz} cleanair device cluster cluster-id
```

To view the CleanAir interferers of a device type, run this command:

```
Device# show ap dot11 {24ghz | 5ghz | 6ghz} cleanair device type
```

To view the CleanAir configuration for a specific AP, run this command:

```
Device# show ap name ap-name dot11 {24ghz | 5ghz | 6ghz} cleanair air-quality
Device# show ap name ap-name dot11 {24ghz | 5ghz | 6ghz} cleanair device
```

To view the continuous transmitter as the CleanAir interference device type, run this command:

```
Device# show ap dot11 6ghz cleanair device type cont-tx
```

Verify CleanAir parameters

Outlines methods for monitoring CleanAir deployment, including configuring air quality traps, tracking interference devices, viewing worst air quality metrics by radio band via GUI, and referencing practical configuration examples.

You can verify CleanAir parameters using these commands:

Table 168: Commands for verifying CleanAir

Command Name	Description
show ap dot11 24ghz cleanair device type all	Displays all the CleanAir interferers for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type ble-beacon	Displays all the Bluetooth Low Energy (BLE) beacons for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type bt-discovery	Displays CleanAir interferers of type BT Discovery for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type bt-link	Displays CleanAir interferers of type BT Link for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type canopy	Displays CleanAir interferers of type Canopy for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type cont-tx	Displays CleanAir interferers of type Continuous transmitter for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type dect-like	Displays CleanAir interferers of type Digital Enhanced Cordless Telecommunications (DECT) Like for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type fh	Displays CleanAir interferers of type 802.11FH for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type inv	Displays CleanAir interferers of type Wi-Fi Inverted for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type jammer	Displays CleanAir interferers of type Jammer for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type mw-oven	Displays CleanAir interferers of type MW Oven for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type nonstd	Displays CleanAir interferers of type Wi-Fi inverted channel for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type superag	Displays CleanAir interferers of type SuperAG for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type tdd-tx	Displays CleanAir interferers of type TDD Transmit for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type video	Displays CleanAir interferers of type Video Camera for the 2.4-GHz band.
show ap dot11 24ghz cleanair device type wimax-fixed	Displays CleanAir interferers of type WiMax Fixed for the 2.4-GHz band.

Configure air quality traps (CLI)

Configure SNMP traps to monitor air quality index (AQI) and generate alerts when thresholds are exceeded.

Set up SNMP traps so that you receive alerts when the air quality falls below a specified threshold.

Procedure

1. Configure the trap **ciscoLwappSiAqLowSeverityHigh** to generate an alert when air quality goes beyond a given threshold.

   On the controller GUI, navigate to **Configuration** > **Radio Configurations** > **Configuration** > **CleanAir** > **Trap configuration**.

   The trap **ciscoLwappSiAqLowSeverityHigh** (OID: 1.3.6.1.4.1.9.9.513.1.6.1.0.5) is used to monitor air quality and generate alerts when the air quality index exceeds a specified threshold.

2. Send the trap from the controller CLI. From the global configuration mode, enable SNMP traps for air quality index (AQI).

   Example

   ```
   Device# snmp-server enable traps wireless SI
   ```

SNMP traps are configured, and the device sends alerts when air quality index thresholds are exceeded.

What to do next
Monitor SNMP alerts to ensure notifications are received as expected.

Monitor with CleanAir device clusters

This section explains CleanAir device clusters and how they manage detections of interferer devices by merging multiple sensor reports into unified records

A CleanAir device cluster is a spectrum monitoring feature that

- merges detections of the same interference device from multiple sensors into a single cluster,
- assigns and maintains a unique cluster ID for each device, and
- preserves device history by keeping cluster IDs active during temporary lapses in detection.

How CleanAir handles interference device monitoring

Some devices conserve power by limiting the transmit time until actually needed, which results in the spectrum sensor to stop detecting the device temporarily. This device is then correctly marked as down and removed from the spectrum database.

In cases when all the interferer detections for a specific device are reported, the cluster ID is kept alive for an extended period of time to prevent possible device-detection bouncing. If the same device is detected again, it is merged with the original cluster ID and the device-detection history is preserved.

> **Note**
>
> You can configure Cisco CleanAir only on CleanAir-enabled APs.

: Bluetooth headset power management

Some Bluetooth headsets operate on battery power and employ methods to reduce power consumption, such as turning off the transmitter when not needed. Such devices can appear to come and go from the classification. To manage these devices, CleanAir keeps the cluster IDs for longer and they are remerged into a single record upon detection. This process smoothens the user records and accurately represents the device history.

CleanAir FAQs

Provides answers to frequently asked questions about Cisco CleanAir, addressing common operational scenarios, troubleshooting steps, and best practices for efficient RF interference management.

Multiple access points detect the same interference device. However, the device shows them as separate clusters or different suspected devices clustered together. Why does this happen?

Access points must be RF neighbors for the device to consider merging the devices that are detected by these access points. An access point takes time to establish neighbor relationships. A few minutes after the device reboots or after there is a change in the RF group, and similar events, clustering will not be very accurate.

How do I view neighbor access points?

To view neighbor access points, use the **show ap** *ap_name* **auto-rf dot11 24ghz 5ghz** command.

This example shows how to display the neighbor access points:

```
Device#show ap name AS-5508-5-AP3 auto-rf dot11 24ghz

<snippet>
Nearby APs
  AP 0C85.259E.C350 slot 0                          :  -12 dBm on   1
(10.10.0.5)
  AP 0C85.25AB.CCA0 slot 0                          :  -24 dBm on   6
(10.10.0.5)
  AP 0C85.25C7.B7A0 slot 0                          :  -26 dBm on  11
(10.10.0.5)
  AP 0C85.25DE.2C10 slot 0                          :  -24 dBm on   6
(10.10.0.5)
  AP 0C85.25DE.C8E0 slot 0                          :  -14 dBm on  11
(10.10.0.5)
  AP 0C85.25DF.3280 slot 0                          :  -31 dBm on   6
(10.10.0.5)
  AP 0CD9.96BA.5600 slot 0                          :  -44 dBm on   6
(10.0.0.2)
  AP 24B6.5734.C570 slot 0                          :  -48 dBm on  11
(10.0.0.2)
<snippet>
```

What are the AP debug commands available for CleanAir?

The AP debug commands for CleanAir are:

- **debug cleanair bringup event logdebug low major nsi offchan**
- **debug rrm neighbor off-channel reports**

189 Bluetooth Low Energy

Topics:

- Bluetooth low energy

Explains how Bluetooth Low Energy beacons work with Cisco CleanAir to enable location-based services, asset tracking, and device detection.

Bluetooth low energy

Details core concepts and characteristics of Bluetooth Low Energy, outlining device roles, communication models, and application scenarios for low-power wireless connectivity.

A Bluetooth low energy device is a wireless personal area network technology that

- enables enhanced location services for mobile devices
- transmits universally unique identifiers (UUIDs), major and minor fields for device identity and context, and
- allows smartphones and APs to detect BLE signals for various use cases, including interferer identification, asset tracking, and targeted notifications.

These details are picked up by Bluetooth-enabled smartphones and devices. The location information of these devices is sent to the corresponding back-end server. Relevant advertisements and other important information are then delivered to the devices using this location-specific information.

Additionally, information about missing tags can be obtained. This feature can determine rogue and malicious tags using the unique identifier associated with each tag (or family of tags) against a predetermined allowed list from a customer. The management function can display or send email alerts based on rogue tags, missing tags, or moved tags.

Note

This feature is not related to the Indoor IoT Services feature set that is part of Cisco Spaces.

It describes how APs and Catalyst 9800 detect BLE devices as wireless interferers using CleanAir, rather than the BLE radio available on some AP models. This feature is not intended for BLE-based asset tracking, environmental monitoring, or tag management use cases. These use cases are powered by Cisco Spaces.

For details about the full functionality of BLE-related use cases in the Cisco solution, refer to the Cisco Spaces configuration guides for Indoor IoT services.

Limitations of BLE Feature

- The wireless infrastructure must support Cisco CleanAir.
- The feature supports a maximum of only 250 unique BLE beacons (cluster entries) and 1,000 device entries.
- The Cisco CleanAir feature is supported only on Cisco Aironet 3700 Series APs with Hyperlocation module RM3010. On Wave 2 and Wi-Fi 6 APs, the BLE feature works differently (through cloud beacon center) and is not covered by this feature.

Areas of Use

The BLE feature provides granular location details of devices, such as smartphones or Bluetooth-enabled devices, which helps deliver context-sensitive advertising and other information to users. Possible areas of application include retail stores, museums, zoos, healthcare, fitness centers, security, and advertising.

Enable Bluetooth low energy beacon (GUI)

Learn how to enable the Bluetooth low energy beacon detection using the CleanAir feature in the GUI.

Enable detection of Bluetooth low energy (BLE) beacons in your wireless network for enhanced interference management using the GUI.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **CleanAir** > **2.4 GHz Band** > **General**.
2. Check the **Enable CleanAir** check box.
3. From the **Available Interference Types** list, select and move **BLE Beacon** to the **Interference Types to Detect** list.
4. Click **Apply**.

Enable Bluetooth low energy beacon (CLI)

Use this task to enable Bluetooth low energy beacon detection on your Cisco controller using commands.

Bluetooth low energy (BLE) detection is enabled by default. Use the procedure given below to enable BLE when it is disabled.

Before you begin

- The wireless infrastructure must support Cisco CleanAir.
- Cisco CleanAir configuration and show commands are available only in Mobility Controller (MC) mode.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Controller# configure terminal
   ```

2. Enable the BLE feature on the 802.11b network.

   Example

   ```
   Controller(config)# ap dot11 24ghz cleanair device ble-beacon
   ```

   Use the **no** form of the command to disable BLE feature on the 802.11b network.

3. Return to the privileged EXEC mode.

   Example

   ```
   Controller(config)# exit
   ```

4. Optional: Display the BLE beacon configuration.

   Example

   ```
   Controller# show ap dot11 24ghz cleanair config
         Interference Device Settings:
         Interference Device Reporting..............  : Enabled
         Bluetooth Link.........................  : Enabled
         Microwave Oven.........................  : Enabled
         BLE Beacon.............................  : Enabled
   ```

5. Optional: Display the BLE beacon device-type information.

Example

```
Controller# show ap dot11 24ghz cleanair device type ble-beacon
      DC    = Duty Cycle (%)
      ISI   = Interference Severity Index (1-Low Interference, 100-High
Interference)
      RSSI  = Received Signal Strength Index (dBm)
      DevID = Device ID
      No    ClusterID          DevID  Type           AP Name
ISI  RSSI  DC   Channel

-------------------------------------------------------------------------------------

      1   2c:92:80:00:00:22  0xa001 BLE Beacon     5508_3_AP3600_f839
--   -74   0    unknown
```

190 Persistent Device Avoidance

Topics:

- Cisco persistent device avoidance

This chapter explains how to enable and verify CleanAir's persistent device avoidance on supported Cisco APs, showing commands and controller output to detect, record, and mitigate recurring non-WiFi interference for optimized channel planning.

Cisco persistent device avoidance

Explains how persistent device avoidance improves WLAN performance by detecting and mitigating interference from periodic devices.

A persistent device avoidance feature is a spectrum management capability that

- detects persistent, periodic interference devices (such as outdoor bridges and microwave ovens) that are usually missed by other RF metrics
- stores device information on the controller to allow mitigation of affected channels, and
- dynamically alters channel plans to minimize interference and optimize WLAN performance because short-duration and periodic operations remain largely undetected by normal RF management metrics.

With Cisco CleanAir, the RRM dynamic channel allocation (DCA) algorithm can detect, measure, register, and remember the impact of interference. It then adjusts the RRM DCA algorithm accordingly. The PDA process minimizes the use of channels affected by persistent devices in the channel plan. This adjustment remains local to the interference source. CleanAir detects and stores persistent device information in the controller. This information is used to mitigate the interfering channels.

Persistent Devices Detection: CleanAir-capable monitor mode APs collect information about persistent devices on all the configured channels and store the information in the controller. Local or bridge mode APs detect interference devices only on the serving channels.

The PDA feature works seamlessly on all platforms. All AP models that support CleanAir and Spectrum Intelligence also support the PDA feature.

The supported platforms are:

- Cisco Aironet 1852 & 1832 APs
- Cisco Aironet 2700 Series APs
- Cisco Aironet 2800 Series APs
- Cisco Aironet 3700 Series APs
- Cisco Aironet 3800 & 4800 Series APs
- Cisco Catalyst 9115 & 9117 Series APs
- Cisco Catalyst 9120AX Series APs
- Cisco Catalyst 9124AX Series APs
- Cisco Catalyst 9130AX APs

Configure persistent device avoidance (GUI)

Configure persistent device avoidance to minimize interference from persistent non-WiFi sources during dynamic channel assignment in the network using the GUI..

Procedure

1. Choose **Configurations** > **Radio Configurations** > **RRM**.
2. Click the **5 GHz Band** tab or the **2.4 GHz Band**, and click the **DCA** tab.
3. In the **DCA** window, under the **Dynamic Channel Assignment Algorithm** section, check the **Avoid Persistent Non-WiFi Interference** check box to enable the device to ignore persistent non-WiFi interference.

4. Click **Apply**.

Configure persistent device avoidance (CLI)

Configure persistent device avoidance using commands.

Configure persistent device avoidance (PDA) on your device using commands.

You can enable and disable the PDA feature and PDA propagation configuration mode through the RRM Manager.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure persistent non-WiFi device avoidance in the 802.11a or 802.11b channel assignment.

   Example

   ```
   Device# [no] ap dot11 24ghz rrm channel device
   ```

   Use the **no** form of this command to negate the command or to set its defaults.

Verify persistent device avoidance

This topic provides instructions to verify that persistent interference devices are detected and avoided by the system using commands.

To verify the current state of **Device Aware** detail of the channel, use this command:

```
Device#show ap dot11 24ghz channel
Leader Automatic Channel Assignment
Channel Assignment Mode                    : AUTO
Channel Update Interval                    : 600 seconds
Anchor time (Hour of the day)              : 0
Channel Update Contribution
Noise                                    : Enable
Interference                             : Enable
Load                                     : Disable
Device Aware
                                             : Enable
CleanAir Event-driven RRM option           : Disabled
Channel Assignment Leader                  : cisco-vwlc (9.9.39.73)
Last Run                                   : 166 seconds ago
DCA Sensitivity Level                      : MEDIUM : 10 dB
DCA Minimum Energy Limit                   : -95 dBm
Channel Energy Levels
Minimum                                  : -82 dBm
Average                                  : -82 dBm
Maximum                                  : -82 dBm
Channel Dwell Times
```

```
Minimum                                  : 8 days 0 hour 43 minutes 13 seconds

Average                                  : 8 days 0 hour 43 minutes 13 seconds

Maximum                                  : 8 days 0 hour 43 minutes 13 seconds

802.11b 2.4 GHz Auto-RF Channel List
Allowed Channel List                      : 1,6,11
Unused Channel List                       : 2,3,4,5,7,8,9,10
```

```
Device#show ap dot11 24ghz cleanair device type all
DC    = Duty Cycle (%)
ISI   = Interference Severity Index (1-Low Interference, 100-High Interference)
RSSI  = Received Signal Strength Index (dBm)
DevID = Device ID
PD    = Persistent Device
ClusterID          Mac Address        DevID  Type           PD   AP Name
                     Version  ISI  RSSI  DC   Channel             Last Update
 Time
-------------------------------------------------------------------------------------------
9800.0000.0004     3400.0200.0001     0x0001 Jammer         No   RRM-TSIM-1
                     CA       100  -67   100  1,2                   12/16/2020
 18:32:42 UTC
9800.0000.0005     3400.0200.0004     0x0004 Xbox           No   RRM-TSIM-1
                     CA       45   -73   45   1                     12/16/2020
 18:32:42 UTC
9800.0000.0006     3400.0200.0006     0x0006 TDD Transmit   No   RRM-TSIM-1
                     CA       10   -75   10   1,2                   12/16/2020
 18:32:42 UTC
9800.0000.0006     3400.0200.0007     0x0007 Continuous TX  No   RRM-TSIM-1
                     CA       30   -77   30   1,2                   12/16/2020
 18:32:42 UTC
9800.0000.0007     3400.0200.0009     0x0009 802.15.4       No   RRM-TSIM-1
                     CA       10   -95   10   1,2                   12/16/2020
 18:32:42 UTC
```

```
Device# show ap dot11 5ghz cleanair device type all
DC    = Duty Cycle (%)
ISI   = Interference Severity Index (1-Low Interference, 100-High Interference)
RSSI  = Received Signal Strength Index (dBm)
DevID = Device ID
PD    = Persistent Device
ClusterID          Mac Address        DevID  Type           PD   AP Name
                     Version  ISI  RSSI  DC   Channel             Last Update
 Time
-------------------------------------------------------------------------------------------
9800.0000.0000     3400.0201.0002     0x0002 WiFi Inverted  No   RRM-TSIM-1
                     CA       45   -63   45   40                    12/16/2020
 18:32:42 UTC
9800.0000.0001     3400.0201.0004     0x0004 TDD Transmit   No   RRM-TSIM-1
                     CA       10   -76   10   40                    12/16/2020
 18:32:42 UTC
9800.0000.0001     3400.0201.0005     0x0005 Continuous TX  No   RRM-TSIM-1
                     CA       30   -77   30   40                    12/16/2020
 18:32:42 UTC
9800.0000.0002     3400.0201.0007     0x0007 BT Link        No   RRM-TSIM-1
                     CA       10   -88   10   40                    12/16/2020
 18:32:42 UTC
```

To verify all the reported interferers along with the class type, use this command:

```
Device# show ap dot11 24ghz cleanair device type wimax-mobile
DC    = Duty Cycle (%)
ISI   = Interference Severity Index (1-Low Interference, 100-High Interference)
RSSI  = Received Signal Strength Index (dBm)
DevID = Device ID
ClusterID          Mac Address        DevID  Type          AP Name
          ISI  RSSI  DC   Channel
_____________________________________________________________________________
1900.0000.0006     xxxx.xxxx.xxx1     0xc001 WiMax Mobile  Cisco-AP
            4   -88   1
1900.0000.0007     xxxx.xxxx.xxx2     0xc002 WiMax Mobile  Cisco-AP
            4   -88   1
```

To verify the persistent device information under Auto-RF, use this command:

```
Device#show ap auto-rf dot11 24ghz
Number of Slots            : 2
AP Name                    : VANC-AP
MAC Address                : d4c9.3ce5.c760
Slot ID                    : 0
Radio Type                 : 802.11n - 2.4 GHz
….…..…..….
Noise Information
……...
Persistent Interference Devices
Class Type                 Channel  DC (%%)  RSSI (dBm)  Last Update Time
-------------------------  -------  ------   ---------   ----------------
MW Oven                     11      NA        -71        08/22/2019 12:03:18
UTC
MW Oven                     11      NA        -24        08/22/2019 12:03:19
UTC
MW Oven                     11      NA        -17        08/22/2019 12:03:16
UTC
MW Oven                     11      NA        -22        08/22/2019 12:03:19
UTC
```

To verify the persistent device information under Auto-RF for specific Cisco APs, use this command:

```
Device# show ap name ap_name

auto-rf dot11 24ghz
Number of Slots            : 2
AP Name                    : VANC-AP
MAC Address                : d4c9.3ce5.c760
Slot ID                    : 0
Radio Type                 : 802.11n - 2.4 GHz
….…..…..….
Noise Information
……...
Persistent Interference Devices
Class Type                 Channel  DC (%%)  RSSI (dBm)  Last Update Time
-------------------------  -------  ------   ---------   ----------------
MW Oven                     11      NA        -71        08/22/2019 12:03:18
UTC
MW Oven                     11      NA        -24        08/22/2019 12:03:19
UTC
MW Oven                     11      NA        -17        08/22/2019 12:03:16
UTC
```

MW Oven	11	NA	-22	08/22/2019 12:03:19 UTC

191 Spectrum Intelligence

Topics:

- Spectrum Intelligence

This chapter explains how to enable and configure Spectrum Intelligence on compatible Cisco APs to detect and classify non-Wi-Fi interference across 2.4 and 5-GHz bands, and how to verify and debug related reports.

Spectrum Intelligence

Explains how Spectrum Intelligence features analyze radio frequencies to detect non-Wi-Fi interference on 2.4-GHz and 5-GHz bands.

A spectrum intelligence feature is a radio analysis tool that

- scans 2.4-GHz and 5-GHz bands for non-Wi-Fi radio interference
- detects and classifies interference types such as microwave, continuous wave (like video bridge and baby monitor), Wi-Fi, and frequency hopping (Bluetooth and frequency-hopping spread spectrum (FHSS) cordless phone), and
- provides detailed radio utilization, noise, air quality, and interference reports for compatible Cisco APs.

For information about APs that support this feature, see the relevant documentation. https://www.cisco.com/c/en/us/td/docs/wireless/access_point/feature-matrix/ap-feature-matrix.html.

Note

You must enable the Spectrum Intelligence feature on Cisco Aironet 1832 and 1852 series APs to view radio details such as noise, air quality, interference, and radio utilization in Cisco Catalyst Center Assurance AP health.

Restrictions

- SI APs report only a single interference type in Local mode.
- SI does not support high availability for air quality or interference reports. High availability is not supported because the interference report or device report will not be copied to standby after switchover. The AP is expected to send the report again if the interfering device is still present.
- Spectrum Intelligence detects only three types of devices:
  - Microwave
  - continuous wave (such as video recorder, baby monitor)
  - SI-FHSS (Bluetooth and frequency hopping Digital European Cordless Telecommunications [DECT] phones).

Configure spectrum intelligence (CLI)

Use this process to enable or disable the Spectrum Intelligence (SI) feature on a Cisco AP using commands. This procedure applies to both 2.4-GHz and 5-GHz radio networks.

Enable network administrators to monitor and analyze the radio frequency (RF) environment by configuring the Spectrum Intelligence (SI) feature on Cisco APs.

Follow the procedure given below to configure spectrum intelligence:

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the 2.4-GHz or 5-GHz spectrum intelligence feature on the 802.11a or 802.11b network.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz} SI
```

Add **no** form of the command to disable SI on the 802.11a or 802.11b network.

Verify spectrum intelligence information

Verify spectrum intelligence information by running key Cisco device commands to check CleanAir and SI status, device types, and interference details for both 2.4-GHz and 5-GHz bands.

Use these commands to verify spectrum intelligence information:

To display the SI information for a 2.4-GHz or 5-GHz band, use this command:

```
Device# show ap dot11 24ghz SI config

SI Solution..................................... : Enabled
Interference Device Settings:
SI_FHSS............................... : Enabled
Interference Device Types Triggering Alarms:
SI_FHSS............................... : Disabled
```

```
Device# show ap dot11 5ghz SI device type
RSSI  = Received Signal Strength Index (dBm)
DevID = Device ID
PD    = Persistent Device
Mac Address        DevID  Type          PD   AP Name
 RSSI  Channel              Last Update Time
------------------------------------------------------------------------------------------
3400.0401.0006     0x0006 BT Discovery   No   RRM-TSIM-3
  -88   40                 12/16/2020 18:11:28 UTC
```

To display SI interferers of type continuous transmitter for a 2.4-GHz band, use this command:

```
Device# show ap dot11 24ghz SI device type cont_tx

RSSI  = Received Signal Strength Index (dBm)
DevID = Device ID
Mac Address        DevID  Type          AP Name                          RSSI
  Channel
-------------------------------------------------------------------------------------
xxxx.xxxx.xxxx     0xf001 Continuous TX Cisco-AP                       -47
```

To display 802.11a interference devices information for the given AP for 5-GHz, use this command:

```
Device# show ap dot11 5ghz SI device type ap

DC    = Duty Cycle (%)
ISI   = Interference Severity Index (1-Low Interference, 100-High Interference)
```

```
RSSI  = Received Signal Strength Index (dBm)
DevID = Device ID
AP type = CA, clean air, SI spectrum intelligence
No  ClusterID/BSSID    DevID  Type    AP Type AP Name                  ISI
RSSI  DC   Channel
--- ------------------ ------ ------- ------  ------------------------ ----
----- ---- ----------
```

To display SI interferers of type continuous transmitter for a 5-GHz band, use this command:

```
Device# show ap dot11 5ghz SI device type cont_tx

RSSI  = Received Signal Strength Index (dBm)
DevID = Device ID
Mac Address        DevID  Type          AP Name                          RSSI
  Channel
-------------------------------------------------------------------------------
xxxx.xxxx.xxx1     0xf001 Continuous TX Cisco-AP                          -88

xxxx.xxxx.xxx2     0xf002 Continuous TX Cisco-AP                          -88
```

To display all Cisco CleanAir interferers for a 2.4-GHz band, use this command:

```
Device# show ap dot11 24ghz cleanair device type all
```

Debugg spectrum intelligence on supported APs (CLI)

Learn how to debug Spectrum Intelligence features on supported APs using commands.

Enable troubleshooting and log generation for Spectrum Intelligence on supported APs using commands.

Procedure

1. Generate major Spectrum Intelligence logs for an AP by entering this command.

   Example

   ```
   debug cleanair major
   ```

2. Verify the Spectrum Intelligence scan schedule of 5 seconds on an AP by entering this command

   Example

   ```
   debug cleanair event
   ```

3. Generate logs at 10-minute interval, when interference is not detected or reported by the AP, by entering this command.

   Example

   ```
   debug cleanair raw 10
   ```

   This command creates three files under **/tmp** directory from dev shell.

   - spectrum.fft
   - spectrum.dbg

- spectrum.int

4. View the Spectrum Intelligence detected interfering devices by entering this command.

   Example

   ```
   show cleanair interferers
   ```

5. View the Spectrum Intelligence configuration status by entering this command.

   Example

   ```
   show cleanair status
   ```

192 Spectrum Analysis

Topics:

- Spectrum analysis

This chapter explains how to enable and use AP radio monitoring GUI and commands and live captures in Cisco Catalyst Center to detect RF interference, interpret persistence, waterfall, and interference charts, and verify settings.

Spectrum analysis

Explains how spectrum analysis charts help network administrators identify RF violations and interpret spectrum data using the Cisco Catalyst Center.

A spectrum analysis chart is a graphical visualization tool that

- Cisco Catalyst Center receives and displays real-time spectrogram streams from APs
- aggregates spectrum data into distinct chart types for network administrators, and
- helps identify and analyze RF violations using color-coded, interactive visual outputs.

To analyze a violation, you should select the corresponding AP and analyze the spectrogram stream.

Each AP uses a specific channel to communicate with clients, depending on whether the setting is global or specific to a channel.

When many clients join the same AP, frames are more likely to be dropped. If clients disconnect quickly or cannot join, perform spectrum analysis to check if the channels are congested.

You can enable spectrum analysis on each AP listed in the web UI and view graphs for the selected AP. When spectrum analysis is enabled, APs send spectrum data to Cisco Catalyst Center, which aggregates it into three distinct charts.

You can view these charts while performing a spectrum analysis:

- Persistence charts plot the amplitude-to-power ratio of each signal at each channel for five minutes. The chart uses blue to represent one signal and red to represent many signals. It also uses opacity to show the age of signal data within the five-minute interval; older data appears more transparent.
- Waterfall charts plot all signals analyzed in the channel for five minutes, with intensity on the X-axis and time on the Y-axis. The chart uses blue for low values and red for high values.
- Interference and Duty charts plot the severity of detected interference for each channel band and list the interference type. Interference appears as a circle; the center indicates severity and the radius shows the affected section of the channel band. The impact is measured by severity values, ranging from 0 to 100. Cisco Catalyst Center determines the interference type from the RF signature using Cisco CleanAir technology.

Live spectrum analysis

This topic explains how live spectrum analyses enable real-time monitoring and visualization of radio frequency interference and usage on APs.

A live spectrum analysis is a radio monitoring feature that

- enables real-time inspection of the spectrum of frequencies generated by AP radios
- displays consolidated interference views across both the 2.4 GHz and 5 GHz ranges when using radio 2, and
- allows users to select an AP radio in the web UI and view its associated live spectrum for a configurable duration for 10 minutes.

You can perform a live spectrum analysis of the AP radios using the web UI, which allows you to monitor the spectrum of frequencies generated by the associated AP radios. The live spectrum capture uses radio 2 if it is available. Otherwise, both radio 0 and radio 1 are used. However, if the feature is enabled on radio 0 or radio 1, you can view only the part of the spectrum associated with those radios.

Perform AP spectrum analysis (GUI)

Learn how to analyze AP spectrum using the Cisco Catalyst Center GUI to evaluate wireless performance using the GUI.

Identify radio frequency interference and optimize wireless network performance using AP spectrum analysis using the GUI.

Before you begin

Use the Cisco Catalyst Center discovery functionality to locate an AP to perform a spectrum analysis.

Procedure

1. Choose **Provision > Inventory**.

   The **Inventory** window is displayed.

2. Click **AP Name**.

   The **360 degree Device** window is displayed.

3. Click **Intelligent Capture**.
4. Click **Spectrum Analysis** to view the graphs.
5. From the **Radio** drop-down list, select a radio.
6. Click **Start Spectrum Analysis**.

   The graphs are displayed on the Web UI for you to analyze.

   To stop the analysis, click **Stop Spectrum Analysis**.

Configure spectrum analysis (CLI)

Configure spectrum analysis on a wireless access point using commands.

Enable spectrum analysis to monitor and troubleshoot RF interference on wireless APs using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure spectrum analysis on the AP.

   Example

   ```
   Device# icap subscription ap rf spectrum enable
   ```

3. Select a radio slot to enable spectrum analysis.

   Example

   ```
   Device# icap subscription ap rf spectrum slot number
   ```

Verify spectrum analysis

Verify spectrum analysis.

This is a sample output of the command that verifies spectrum analysis on a selected AP:

```
Device# show ap icap subscription name
Subscription list
-----------------
Full Pkt Capture : Disabled
Partial Pkt Capture : Enabled
Anomaly Event : Enabled
Debug : Disabled
Stats : Disabled
Ap Operational Data : Disabled
Sensor Message : Enabled
RRM Operational Data : Disabled
Client Events : Disabled
aWIPS Forensic Pkts: Disabled
MAC and Filters subscription list
---------------------------------
Full-packet-trace: None
Partial-packet-trace: None
Filters: None
Anomaly Detection: None
Client Stats
------------
None
RF Spectrum
-----------
Radio Slot(s): 1
```

Part XVI

Mesh Access Points

Topics:

- Mesh Access Points
- Redundant Root Access Point (RAP) Ethernet Daisy Chaining

193 Mesh Access Points

Topics:

- Mesh network
- Mesh Authorization and Security
- Bridge group names
- Mesh background scanning
- Mesh serial backhaul
- Mesh backhaul RRM
- Mesh backhaul at 2.4 GHz and 5 GHz
- Ethernet bridging
- DHCP and NAT functionality on root AP (RAP)
- Mesh convergence
- Mesh daisy chaining
- FlexConnect + bridge mode
- Mesh AP Roles and Topology
- Link test
- Air time fairness on mesh
- Mesh call admission control
- Fast teardown for a mesh deployment

Mesh network

Introduces mesh architecture, RRM feature support, restrictions, country codes, controller interoperability, enabling mesh profiles, associating mesh features, and verifies mesh configuration for foundational understanding and initial setup.

A mesh network is a wireless network topology that

- uses interconnected Cisco Aironet outdoor mesh APs and indoor mesh APs to relay data
- provides scalability, central management, and seamless mobility between indoor and outdoor deployments, and
- is coordinated and provisioned using Cisco Wireless Controller and Cisco Prime Infrastructure.

The Control and Provisioning of Wireless Access Points (CAPWAP) protocol manages the connection of mesh access points to the network, ensuring secure and efficient network operations.

End-to-end security

End-to-end security within the mesh network is supported by employing Advanced Encryption Standard (AES) encryption between wireless mesh APs and Wi-Fi Protected Access 2 (WPA2) clients. For connections to a mesh AP (MAP) wireless client, such as MAP-to-MAP and MAP-to-root access point, WPA2 is applicable.

Wireless mesh termination points

The wireless mesh terminates at two points on the wired network. The first location is where the root access point (RAP) is attached and all bridged traffic connects to the wired network. The second location is where the CAPWAP controller connects to the wired network. At this location, the WLAN client traffic from the mesh network connects to the wired network. The WLAN client traffic from CAPWAP is tunneled to Layer 2. Matching WLANs should terminate on the same switch VLAN as the wireless controllers. The security and network configuration for each of the WLANs on the mesh depend on the security capabilities of the network to which the wireless controller is connected.

In the new configuration model, the controller has a default mesh profile. This profile is mapped to the default AP-join profile, which is in turn is mapped to the default site tag. If you are creating a named mesh profile, ensure that these mappings exist and that the corresponding AP is added to the appropriate site-tag.

Mesh scenarios in IRCM

Important

These are the mesh scenarios in IRCM from Cisco IOS XE Amsterdam 17.3 up to Cisco IOS XE Cupertino 17.9, for the Cisco Wave 1 APs that are not supported:

- Cisco Wave 1 APs are not supported in the releases after Cisco IOS XE Amsterdam 17.3. This includes mesh support as well. Therefore, it is not possible for a Cisco Wave 1 AP to join a Cisco Catalyst 9800 Series Wireless Controller (controller) with Cisco IOS XE Amsterdam 17.4 and later versions. We recommend this deployment mode for Cisco Wave 1 APs.
- In the case of Cisco mesh deployments, these are the deployment limitations to be aware of when the system is deployed:
  - MAP roaming is not allowed between Cisco Catalyst 9800 Series Wireless Controllers, if the controllers run different Cisco IOS XE versions (running on versions Cisco IOS XE Amsterdam 17.3 or Cisco IOS XE Cupertino 17.9) for any of the Cisco Wave 1 APs and Cisco Wave 2 APs.
  - You cannot have Cisco Wave 1 APs and Cisco Catalyst 9124 Series APs in the same mesh tree, in the releases post Cisco IOS XE Amsterdam 17.3.x. This can be achieved in 17.3.x, starting with the 17.3.6 (upcoming) release.

- The whole mesh tree containing Cisco Wave 1 APs must be joined to the 17.3 controller, by running the **strict-bgn** and **mac filtering** commands.

> **Note**
>
> The limitations mentioned above are not valid for the Cisco Industrial Wireless 3702 Series Access Points, which are supported until the Cisco IOS XE Cupertino 17.9 release.

Mesh RRM feature support and evolution across AP generations

Describes the Mesh RRM feature support and the evolution across AP generations.

This topic describes the mesh Radio Resource Management (RRM) feature support, its evolution across different generations of Cisco APs, differences between AP families, key features, and the progression of mesh RRM capabilities in various software releases.

Mesh AP families

The two families of mesh APs available are the Cisco Wave 1 APs and the Cisco Wave 2 APs and the AX APs. These families have different capabilities and features, and they are not fully compatible with each other. The Cisco Wave 1 APs do not support the same feature set as the Cisco Wave 2 and AX APs.

Both families of APs support background scanning. However, their implementations are not interoperable, so you cannot use both in a single mesh deployment.

Evolution in the mesh RRM feature across AP generations

Cisco Wave 1 APs have not received significant mesh RRM feature updates after the Cisco IOS XE Amsterdam 17.3 release. Mesh RRM features on Cisco Wave 2 and AX APs have gained many improvements in recent releases.

- Cisco IOS XE Cupertino 17.9.1 introduced mesh RRM DCA using radio information only from RAPs.
- Cisco IOS XE Dublin 17.11 introduced mesh background scanning and FastaAncestor find mode.
- Cisco IOS XE 17.14 introduced mesh RRM DCA that uses radio information from the entire mesh tree.

Restrictions for mesh access points

Outlines restrictions and considerations for mesh access points configuration and deployment.

Feature restrictions

For information about APs that support Mesh feature, see https://www.cisco.com/c/en/us/td/docs/wireless/access_point/feature-matrix/ap-feature-matrix.html.

- Serial backhaul AP support with separate backhaul radios for uplink and downlink.
- Public Safety channels (4.9-GHz band) support.
- Passive Beaconing (Anti-Stranding)

Configuration considerations

Only Root APs support SSO. MAPs will disconnect and rejoin after SSO.

- In a mixed regulatory domain mesh AP deployment, ensure that the Dynamic Channel Assignment (DCA) allowed channel list is supported by MAPs.
- When you disable the admin state on the 2.4-GHz radio of mesh APs, and the root AP (RAP) backhaul radio is switched to 2.4-GHz, RAP will still use 2.4-GHz radio to serve the mesh backhaul connections, inspite of 2.4-GHz radio being in the disabled state.

The AP Stateful Switch Over (SSO) feature allows the access point (AP) to establish a CAPWAP tunnel with the Active controller and share a mirror copy of the AP database with the Standby controller. The overall goal for the addition of AP SSO support to the controller is to reduce major downtime in wireless networks due to failure conditions that may occur due to box failover or network failover.

Country codes

This concept explains how country codes ensure that controllers and access points (APs) meet local regulatory compliance, including broadcast frequencies, channels, and power levels.

A country code is a regulatory compliance setting that

- allows specification of the intended country of operation for controllers and APs
- ensures adherence to local regulations about broadcast frequencies, channels, and power levels, and
- maintains correct assignment of regulatory domains for each device.

In certain countries, there is a difference for indoor and outdoor APs in these areas:

- Regulatory domain code
- Set of channels supported
- Transmit power level

Controllers and APs are designed for use in many countries with varying regulatory requirements. At the factory, the radios within the APs are assigned to a specific regulatory domain (such as -E for Europe). The country code then enables you to specify a particular country of operation (such as FR for France or ES for Spain). Configuring a country code ensures that the broadcast frequency bands, interfaces, channels, and transmit power levels of each radio are compliant with country-specific regulations.

Mesh interoperability between controllers

This concept explains mesh interoperabilities between controllers, detailing how mesh access points (MAPs) and parent controllers interact across AireOS and Catalyst 9800 networks.

Mesh interoperability between controllers is a wireless mesh network capability that

- enables MAPs to join an AireOS controller through a mesh network formed by APs connected to a Cisco Catalyst 9800 Series Wireless Controller
- allows MAPs to join a Cisco Catalyst 9800 Series Wireless Controller through a mesh network formed by APs connected to an AireOS controller, and
- supports MAP roaming between parent mesh APs connected to AireOS and Cisco Catalyst 9800 Series Wireless Controller using PMK cache.

> **Note**
>
> For seamless interoperability, the AireOS controller and the Cisco Catalyst 9800 Series Wireless Controller must be in the same mobility group. Both controllers should use image versions that support Inter-Release Controller Mobility (IRCM).

Indoor mesh interoperability with outdoor mesh

This concept explains how indoor and outdoor MAPs extend wireless coverage and interoperate in a mesh network.

A mesh network is a wireless connectivity architecture that

- enables both indoor and outdoor MAPs to connect and share coverage
- allows seamless mobility groups between the two environments, and
- supports management of all MAPs through a single controller.

The system supports interoperability between indoor MAPs and outdoor APs. By default, you should use indoor MAPs inside buildings. Deploy them outside only for temporary, short-haul coverage extensions, such as connecting a building to a nearby parking lot.

You can share mobility groups between indoor and outdoor networks, and you can manage all devices with a single controller. The same WLANs that you configure can be broadcast from both indoor and outdoor MAPs.

Enable wireless mesh profile (CLI)

Enable wireless mesh profile.

To enable and configure a wireless mesh profile to support mesh networking functionality on your device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a mesh profile and enter mesh profile configuration mode.

   Example

   ```
   Device(config)# wireless profile mesh profile-name
   ```

3. Enable the fast teardown of mesh network and configure the parameters of the feature.

   Example

   ```
   Device(config-wireless-profile-mesh)# fast-teardown
   ```

The wireless mesh profile is enabled and configured with the fast teardown feature.

```
Device# configure terminal
Device(config)# wireless profile mesh mesh1
Device(config-wireless-profile-mesh)# fast-teardown
```

Associate wireless mesh to an AP profile (CLI)

Learn how to associate a wireless mesh profile with an AP profile using CLI commands.

Associate a mesh profile with an AP profile to enable mesh capabilities on compatible APs through CLI configuration.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the AP profile to enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile default-ap-profile
   ```

3. In AP profile configuration mode, configure the mesh profile.

   Example

   ```
   Device(config-ap-profile)# mesh-profile mesh-profile-name
   ```

The AP profile now uses the specified mesh profile. All APs using this profile inherit mesh configuration parameters.

```
Device# configure terminal
Device(config)# ap profile default-ap-profile
Device(config-ap-profile)# mesh-profile test1
```

Verify mesh configuration

Verify mesh configuration.

Use these **show** commands to verify the various aspects of mesh configuration:

- **show wireless mesh stats** *ap-name*
- **show wireless mesh security-stats** {*all* | *ap-name*}
- **show wireless mesh queue-stats** {*all* | *ap-name*}
- **show wireless mesh per-stats summary** {*all* | *ap-name*}
- **show wireless mesh neighbor summary** {*all* | *ap-name*}
- **show wireless mesh neighbor detail** *ap-name*

- **show wireless mesh ap summary**
- **show wireless mesh ap tree**
- **show wireless mesh ap backhaul**
- **show wireless mesh config**
- **show wireless mesh convergence detail** *bridge-group-name*
- **show wireless mesh convergence subset-channels**
- **show wireless mesh neighbor**
- **show wireless profile mesh detailed** *mesh-profile-name*
- **show wireless stats mesh security**
- **show wireless stats mesh queue**
- **show wireless stats mesh packet error**
- **show wireless mesh ap summary**
- **show ap name** *ap-name* **mesh backhaul**
- **show ap name** *ap-name* **mesh neighbor detail**
- **show ap name** *ap-name* **mesh path**
- **show ap name** *ap-name* **mesh stats packet error**
- **show ap name** *ap-name* **mesh stats queue**
- **show ap name** *ap-name* **mesh stats security**
- **show ap name** *ap-name* **mesh stats**
- **show ap name** *ap-name* **mesh bhrate**
- **show ap name** *ap-name* **config ethernet**
- **show ap name** *ap-name* **cablemodem**
- **show ap name** *ap-name* **environment**
- **show ap name** *ap-name* **gps location**
- **show ap name** *ap-name* **environment**
- **show ap name** *ap-name* **mesh linktest data** *dest-mac*
- **show ap environment**
- **show ap gps location**

For details about these commands, see the *Cisco Catalyst 9800 Series Wireless Controller Command Reference* document.

MAC authorization

Use this **show** command to verify the MAC authorization configuration:

```
Device# show run aaa
aaa authentication dot1x CENTRAL_LOCAL local
aaa authorization credential-download CENTRAL_AUTHOR local
```

```
username 002cc8de4f31 mac
username 00425a0a53b1 mac

ewlc_eft#sh wireless profile mesh detailed madhu-mesh-profile

Mesh Profile Name             : abc-mesh-profile
-----------------------------------------------
Description                   :
Bridge Group Name             : bgn-abbc
Strict match BGN              : ENABLED
Amsdu                         : ENABLED
...
Battery State                 : ENABLED
Authorization Method          : CENTRAL_AUTHOR
Authentication Method         : CENTRAL_LOCAL
Backhaul tx rate(802.11bg)    : auto
Backhaul tx rate(802.11a)     : 802.11n mcs15
```

PSK provisioning

Use this **show** command to verify PSK provisioning configuration:

```
Device# show wireless mesh config
Mesh Config
  Backhaul RRM                                  : ENABLED
  Mesh CAC                                      : DISABLED
  Outdoor Ext. UNII B Domain channels(for BH)   : ENABLED
  Mesh Ethernet Bridging STP BPDU Allowed       : ENABLED
  Rap Channel Sync                              : ENABLED

Mesh Alarm Criteria
  Max Hop Count                                 :  4
  Recommended Max Children for MAP              : 10
  Recommended Max Children for RAP              : 20
  Low Link SNR                                  : 12
  High Link SNR                                 : 60
  Max Association Number                        : 10
  Parent Change Number                          :  3

Mesh PSK Config
  PSK Provisioning                              : ENABLED
  Default PSK                                   : ENABLED
  PSK In-use key number                         : 1
  Provisioned PSKs(Maximum 5)

  Index    Description
  ------   ------------
  1         key1
```

Bridge group name

Use this **show** command to verify the bridge group name configuration:

```
Device# show wireless profile mesh detailed abc-mesh-profile
Mesh Profile Name             : abc-mesh-profile
-----------------------------------------------
Description                   :
Bridge Group Name             : bgn-abc
Strict match BGN              : ENABLED
Amsdu                         : ENABLED
```

```
Background Scan                : ENABLED
Channel Change Notification    : DISABLED
Backhaul client access         : ENABLED
Ethernet Bridging              : ENABLED
Ethernet Vlan Transparent      : DISABLED
Full Sector DFS                : ENABLED
IDS                            : ENABLED
Multicast Mode                 : In-Out
Range in feet                  : 12000
Security Mode                  : EAP
Convergence Method             : Fast
LSC only Authentication        : DISABLED
Battery State                  : ENABLED
Authorization Method           : CENTRAL_AUTHOR
Authentication Method          : CENTRAL_LOCAL
Backhaul tx rate(802.11bg)     : auto
Backhaul tx rate(802.11a)      : 802.11n mcs15
```

Backhaul client access

Use this **show** command to verify the backhaul client access configuration:

```
Device# show wireless profile mesh detailed abc-mesh-profile
Mesh Profile Name              : abc-mesh-profile
-------------------------------------------------
Description                    :
Bridge Group Name              : bgn-abc
Strict match BGN               : ENABLED
Amsdu                          : ENABLED
Background Scan                : ENABLED
Channel Change Notification    : DISABLED
Backhaul client access         : ENABLED
Ethernet Bridging              : ENABLED
Ethernet Vlan Transparent      : DISABLED
...
Backhaul tx rate(802.11bg)     : auto
Backhaul tx rate(802.11a)      : 802.11n mcs15
```

Wireless backhaul data rate

Use this **show** command to verify the wireless backhaul data rate configuration:

```
Device# show wireless profile mesh detailed abc-mesh-profile
Mesh Profile Name              : abc-mesh-profile
-------------------------------------------------
Description                    :
Bridge Group Name              : bgn-abc
Strict match BGN               : ENABLED
...
Authorization Method           : CENTRAL_AUTHOR
Authentication Method          : CENTRAL_LOCAL
Backhaul tx rate(802.11bg)     : auto
Backhaul tx rate(802.11a)      : 802.11n mcs15
```

Dynamic frequency selection

Use this **show** command to verify the dynamic frequency selection configuration:

```
Device# show wireless profile mesh detailed abc-mesh-profile
Mesh Profile Name             : abc-mesh-profile
-------------------------------------------------
Description                   :
Bridge Group Name             : bgn-abc
Strict match BGN              : ENABLED
Amsdu                         : ENABLED
Background Scan               : ENABLED
Channel Change Notification   : DISABLED
Backhaul client access        : ENABLED
Ethernet Bridging             : ENABLED
Ethernet Vlan Transparent     : DISABLED
Full Sector DFS               : ENABLED
...
Backhaul tx rate(802.11a)     : 802.11n mcs15
```

Intrusion detection system

Use this **show** command to verify the wireless backhaul data rate configuration:

```
Device# show wireless profile mesh detailed abc-mesh-profile
Mesh Profile Name             : abc-mesh-profile
-------------------------------------------------
Description                   :
Bridge Group Name             : bgn-abc
Strict match BGN              : ENABLED
Amsdu                         : ENABLED
Background Scan               : ENABLED
Channel Change Notification   : DISABLED
Backhaul client access        : ENABLED
Ethernet Bridging             : ENABLED
Ethernet Vlan Transparent     : DISABLED
Full Sector DFS               : ENABLED
IDS                           : ENABLED
Multicast Mode                : In-Out
...
Backhaul tx rate(802.11a)     : 802.11n mcs15
```

Ethernet bridging

Use this **show** command to verify ethernet bridging configuration:

```
Device# show wireless profile mesh detailed abc-mesh-profile
Mesh Profile Name             : abc-mesh-profile
-------------------------------------------------
Description                   :
Bridge Group Name             : bgn-abc
Strict match BGN              : ENABLED
Amsdu                         : ENABLED
Background Scan               : ENABLED
Channel Change Notification   : DISABLED
Backhaul client access        : ENABLED
Ethernet Bridging             : ENABLED
Ethernet Vlan Transparent     : DISABLED
Full Sector DFS               : ENABLED
IDS                           : ENABLED
Multicast Mode                : In-Out
```

```
...
Backhaul tx rate(802.11a)     : 802.11n mcs15
```

Multicast over mesh

Use this **show** command to verify multicast over Mesh configuration:

```
Device# show wireless profile mesh detailed abc-mesh-profile
Mesh Profile Name             : abc-mesh-profile
-------------------------------------------------
Description                   :
Bridge Group Name             : bgn-abc
Strict match BGN              : ENABLED
Amsdu                         : ENABLED
Background Scan               : ENABLED
Channel Change Notification   : DISABLED
Backhaul client access        : ENABLED
Ethernet Bridging             : ENABLED
Ethernet Vlan Transparent     : DISABLED
Full Sector DFS               : ENABLED
IDS                           : ENABLED
Multicast Mode                : In-Out
...
Backhaul tx rate(802.11a)     : 802.11n mcs15
```

RRM on mesh backhaul

Use this **show** command to verify RRM on Mesh backhaul configuration:

```
Device# show wireless mesh config
Mesh Config
  Backhaul RRM                                    : ENABLED
  Mesh CAC                                        : DISABLED
  Outdoor Ext. UNII B Domain channels(for BH)     : ENABLED
  Mesh Ethernet Bridging STP BPDU Allowed         : ENABLED
  Rap Channel Sync                                : ENABLED

Mesh Alarm Criteria
  Max Hop Count                                   :  4
  Recommended Max Children for MAP                : 10
  Recommended Max Children for RAP                : 20
  Low Link SNR                                    : 12
  High Link SNR                                   : 60
  Max Association Number                          : 10
  Parent Change Number                            :  3

Mesh PSK Config
  PSK Provisioning                                : ENABLED
  Default PSK                                     : ENABLED
  PSK In-use key number                           : 1
  Provisioned PSKs(Maximum 5)

  Index    Description
  ------   ------------
  1         key1
```

Preferred parent selection

Use this **show** command to verify preferred parent configuration:

```
Device# show wireless mesh ap tree
=================================================================================
AP Name [Hop Ctr,Link SNR,BG Name,Channel,Pref Parent,Chan Util,Clients]
=================================================================================

[Sector 1]
-----------
1542-RAP [0, 0, bgn-madhu, (165), 0000.0000.0000, 1%, 0]
   |-MAP-2700 [1, 67, bgn-madhu, (165), 7070.8b7a.6fb8, 0%, 0]

Number of Bridge APs : 2
Number of RAPs : 1
Number of MAPs : 1

(*)  Wait for 3 minutes to update or Ethernet Connected Mesh AP.
(**) Not in this Controller
```

AP role change

Use this **show** command to verify AP role change configuration:

```
Device# show wireless mesh ap summary
AP Name                          AP Model BVI MAC        BGN        AP Role
-------                          -------- -------        ---        -------
1542-RAP                         1542D    002c.c8de.1338 bgn-abc  Root AP
MAP-2700                         2702I    500f.8095.01e4 bgn-abc  Mesh AP

Number of Bridge APs         :   2
Number of RAPs               :   1
Number of MAPs               :   1
Number of Flex+Bridge APs    :   0
Number of Flex+Bridge RAPs   :   0
Number of Flex+Bridge MAPs   :   0
```

Mesh leaf node

Use this **show** command to verify mesh leaf node configuration:

```
Device# show ap name MAP-2700 config general
Cisco AP Name   : MAP-2700
=================================================

Cisco AP Identifier                             : 7070.8bbc.d3e0
Country Code                                    : Multiple Countries :
IN,US,IO,J4
Regulatory Domain Allowed by Country            : 802.11bg:-AEJPQU
802.11a:-ABDJNPQU
AP Country Code                                 : IN  - India
AP Regulatory Domain
  Slot 0                                        : -A
  Slot 1                                        : -D
MAC Address                                     : 500f.8095.01e4
...
AP Mode                                         : Bridge
Mesh profile name                               : abc-mesh-profile
AP Role                                         : Mesh AP
Backhaul radio type                             : 802.11a
```

```
Backhaul slot id                                    : 1
Backhaul tx rate                                    : auto
Ethernet Bridging                                   : Enabled
Daisy Chaining                                      : Disabled
Strict Daisy Rap                                    : Disabled
Bridge Group Name                                   : bgn-abc
Strict-Matching BGN                                 : Enabled
Preferred Parent Address                            : 7070.8b7a.6fb8
Block child state                                   : Disabled
PSK Key Timestamp                                   : Not Configured
...
FIPS status                                         : Disabled
WLANCC status                                       : Disabled
GAS rate limit Admin status                         : Disabled
WPA3 Capability                                     : Disabled
EWC-AP Capability                                    : Disabled
AWIPS Capability                                    : Disabled
Proxy Hostname                                      : Not Configured
Proxy Port                                          : Not Configured
Proxy NO_PROXY list                                 : Not Configured
GRPC server status                                  : Disabled
```

Subset channel synchronization

Use this **show** command to verify the subset channel synchronization configuration:

```
Device# show wireless mesh config
Mesh Config
  Backhaul RRM                                      : ENABLED
  Mesh CAC                                          : DISABLED
  Outdoor Ext. UNII B Domain channels(for BH)       : ENABLED
  Mesh Ethernet Bridging STP BPDU Allowed           : ENABLED
  Rap Channel Sync                                  : ENABLED

Mesh Alarm Criteria
  Max Hop Count                                     :  4
  Recommended Max Children for MAP                  : 10
  Recommended Max Children for RAP                  : 20
  Low Link SNR                                      : 12
  High Link SNR                                     : 60
  Max Association Number                            : 10
  Parent Change Number                              :  3

Mesh PSK Config
  PSK Provisioning                                  : ENABLED
  Default PSK                                       : ENABLED
  PSK In-use key number                             : 1
  Provisioned PSKs(Maximum 5)

  Index    Description
  ------   ------------
  1         key1
```

Provisioning LSC for bridge-mode and mesh APs

Use this **show** command to verify the provisioning LSC for Bridge-Mode and Mesh AP configuration:

```
Device# show wireless profile mesh detailed default-mesh-profile
Mesh Profile Name             : default-mesh-profile
-------------------------------------------------
```

```
Description                    : default mesh profile
Bridge Group Name              : bgn-abc
Strict match BGN               : DISABLED
Amsdu                          : ENABLED
Background Scan                : ENABLED
Channel Change Notification    : ENABLED
Backhaul client access         : ENABLED
Ethernet Bridging              : DISABLED
Ethernet Vlan Transparent      : ENABLED
Full Sector DFS                : ENABLED
IDS                            : DISABLED
Multicast Mode                 : In-Out
Range in feet                  : 12000
Security Mode                  : EAP
Convergence Method             : Fast
LSC only Authentication        : DISABLED
Battery State                  : ENABLED
Authorization Method           : default
Authentication Method          : default
Backhaul tx rate(802.11bg)     : auto
Backhaul tx rate(802.11a)      : auto
```

Specify backhaul slot for root AP

Use this **show** command to verify the backhaul slot for the Root AP configuration:

```
Device# show ap name 1542-RAP mesh backhaul
MAC Address : 380e.4d85.5e60
  Current Backhaul Slot: 1
  Radio Type: 0
  Radio Subband: All
  Mesh Radio Role: DOWNLINK
  Administrative State: Enabled
  Operation State: Up
  Current Tx Power Level:
  Current Channel: (165)
  Antenna Type: N/A
  Internal Antenna Gain (in .5 dBm units): 18
```

Use a link test on mesh backhaul

Use this **show** command to verify the use of link test on mesh backhaul configuration:

```
Device# show ap name 1542-RAP mesh linktest data 7070.8bbc.d3ef
380e.4d85.5e60 ==> 7070.8bbc.d3ef

Started at : 05/11/2020 20:56:28
Status: In progress

Configuration:
==============
Data rate:  Mbps
Packets per sec: : 234
Packet Size: : 1200
Duration: : 200
```

Mesh CAC

Use this **show** command to verify mesh CAC configuration:

```
Device# show wireless mesh config
Mesh Config
  Backhaul RRM                                    : ENABLED
  Mesh CAC                                        : DISABLED
  Outdoor Ext. UNII B Domain channels(for BH)     : ENABLED
  Mesh Ethernet Bridging STP BPDU Allowed         : ENABLED
  Rap Channel Sync                                : ENABLED

Mesh Alarm Criteria
  Max Hop Count                                   :  4
  Recommended Max Children for MAP                : 10
  Recommended Max Children for RAP                : 20
  Low Link SNR                                    : 12
  High Link SNR                                   : 60
  Max Association Number                          : 10
  Parent Change Number                            :  3

Mesh PSK Config
  PSK Provisioning                                : ENABLED
  Default PSK                                     : ENABLED
  PSK In-use key number                           : 1
  Provisioned PSKs(Maximum 5)

  Index    Description
  ------   ------------
  1         key1
```

Mesh Authorization and Security

Covers MAC, EAP, and preshared-key authentication, certificate provisioning, and intrusion detection for mesh APs.

MAC authorization

MAC authorization controls whether a MAP can join a wireless controller by checking the AP's MAC address.

A MAC authorization is a security mechanism that

- restricts MAPs from joining a controller unless their MAC address is pre-approved
- ensures only authorized mesh APs participate in the wireless network, and
- can be handled using either an internal list or an external AAA server.

You must enter the MAC address of an AP in the controller for a MAP to join the controller. The controller responds only to CAPWAP requests from MAPs listed in its authorization list. Remember to use the MAC address provided on the back of the AP.

MAC authorization for MAPs connected to the controller over Ethernet occurs during the CAPWAP join process. For MAPs joining the controller over radio, MAC authorization occurs when the corresponding AP attempts to secure an adaptive wireless path protocol (AWPP) link with the parent MAP. AWPP operates as the protocol for Cisco mesh networks.

The Cisco Catalyst 9800 Series Wireless Controller supports MAC authorization internally as well as using an external AAA server.

Configure MAC authorization (GUI)

Configure MAC authorization using GUI.

Enable MAC-based authorization for wireless mesh networks. Specify which devices can connect by entering their MAC addresses.

Procedure

1. Choose **Configuration** > **Security** > **AAA** > **AAA Advanced** > **Device Authentication**.
2. Click **Add**.
   The **Quick Step: MAC Filtering** page is displayed.
3. In the **Quick Step: MAC Filtering** page, complete these steps:
   a) Enter the **MAC Address** . The MAC address can be in either *xx:xx:xx:xx:xx:xx* , *xx-xx-xx-xx-xx-xx* , or *xxxx.xxxx.xxxx* format.
   b) Choose the **Attribute List Name** from the drop-down list.
   c) Choose the **WLAN Profile Name** from the drop-down list.
   d) Click **Apply to Device**.

   Both WebUI and CLI support MAC user configuration in these formats: xxxxxxxxxxxx, xx:xx:xx:xx:xx:xx, xx-xx-xx-xx-xx-xx, or xxxx.xxxx.xxxx. The AP sends the default MAC address without any delimiter. If the MAC address is configured with a delimiter, AP authorization fails unless it uses the xxxxxxxxxxxx format.
4. Choose **Configuration** > **Security** > **AAA** > **AAA Method List** > **Authorization**.
5. Click **Add**.
   The **Quick Step: AAA Authorization** window is displayed.
6. In the **Quick Step: AAA Authorization** page, complete these actions:
   a) Enter the **Method List Name**.
   b) Choose the **Type** from the drop-down list.
   c) Choose the **Group Type** from the drop-down list.
   d) Check the **Fallback to Local** check box.
   e) Check the **Authenticated** check box.
   f) Move the required servers from the **Available Server Groups** to the **Assigned Server Groups.**
   g) Click **Apply to Device**.
7. Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**.
8. Click the mesh profile.
   The **Edit Mesh Profile** page is displayed.
9. Click the **Advanced** tab.
10. In the **Security** settings, from the **Method** drop-down list, choose **EAP**.
11. Choose the **Authentication Method** from the drop-down list.
12. Choose the **Authorization Method** from the drop-down list.
13. Click **Update & Apply to Device**.

Authorized devices with the specified MAC addresses can connect to the wireless mesh network.

Configure MAC authorization (CLI)

Use this procedure to configure MAC authorization for bridge mode access points (APs) using CLI commands on the controller.

Enable MAC authorization for bridge mode APs by configuring the credentials and authorization methods needed.

Add the MAC address of a bridge mode AP to the controller to allow authentication.

Before you begin

- MAC filtering for bridge mode APs is enabled by default on the controller. Configure only the MAC address. Find the MAC address on the back of the AP.
- MAC authorization is supported using both internal and external AAA servers.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure username authentication for MAC filtering, using the MAC address as the username.

   Example

   ```
   Device(config)# username user-name
   ```

3. Set the authorization method list to use local credentials.

   Example

   ```
   Device(config)# aaa authorization credential-download method-name local
   ```

4. Set the authorization method list to use a RADIUS server group. The command supports up to 14 lines.

   Example

   ```
   Device(config)# aaa authorization credential-download method-name radius group
    server-group-name
   ```

5. Configure a mesh profile to access mesh profile configuration mode.

   Example

   ```
   Device(config)# wireless profile mesh mesh-profile-name
   ```

6. Configure the authorization method for mesh APs.

   Example

   ```
   Device(config-wireless-mesh-profile)# method authorization method-name
   ```

You have successfully configured MAC authorization for bridge mode access points using the CLI on the controller.

```
Device# configure terminal
Device(config)# username username1
Device(config)# aaa authorization credential-download list1 local
Device(config)# aaa authorization credential-download auth1 radius group
radius-server-1
```

```
Device(config)# wireless profile mesh mesh-profile
Device(config-wireless-mesh-profile)# method authorization auth1
```

EAP authentication

EAP authentication methods are used to provide secure user and device authentication to wireless networks.

A EAP authentication is a wireless network authentication method that

- allows users and wireless clients to be authenticated locally on a controller
- removes dependence on external authentication servers, and
- supports additional modules such as LSC-based authentication for enhanced security.

Use Local EAP in remote offices to maintain connectivity during backend disruptions. The controller acts as both a server and a user database, retrieving user credentials locally or through LDAP. Local EAP supports the EAP-FAST method for MAP authentication.

An LDAP backend database allows the controller to query an LDAP server for the credentials (username and password) of a particular user. The controller uses these credentials to authenticate the user.

> **Note**
>
> If you configure RADIUS servers on the controller, it will first authenticate the wireless clients using those servers. Local EAP is attempted only if RADIUS servers are not found, timed out, or were not configured.

Locally significant certificate-based (LSC) EAP authentication

A locally significant certificate-based authentication is a WLAN security method that

- authenticates network devices (such as APs and controllers) using locally generated digital certificates
- allows both locally significant certificates (LSCs) and manufacturing installed certificates (MICs) to coexist (with LSCs taking precedence), and
- supports advanced security features, like mesh EAP authentication and master session key generation.

LSC-based EAP authentication requires a public key infrastructure (PKI). This establishes certification authorities, defines policies, and sets validity periods and restrictions.

Certificates (LSCs) must be generated and installed on Access Points and controllers. If an Access Point is provisioned with an LSC, the MIC certificate is not used during boot-up. Changes from LSC to MIC require the Access Point to reboot. LSC-based EAP authentication is supported for mesh Access Points (MAPs). The controller supports mesh security with EAP authentication to a designated server to authenticate mesh child Access Points and generate master session keys for packet encryption.

If a customer installs an LSC (certificate they generate), the AP will use that certificate for authentication and not the factory-installed MIC. Mesh child APs can use LSC EAP authentication to securely join the network and establish encrypted sessions.

The controller also supports mesh security with EAP authentication to a designated server in order to:

- Authenticate the mesh child AP
- Generate a master session key (MSK) for packet encryption.

Configure MAP authorization - EAP (GUI)

Configure MAP authorization - EAP using GUI.

Enable secure authentication for mesh access points by configuring MAP authorization to use EAP (Extensible Authentication Protocol) through the graphical user interface (GUI).

Procedure

1. Choose **Configuration** > **Security** > **AAA** > **AAA Method List** > **Device Authentication**.
2. Click **Add**.
3. Enter **Method List Name**.
4. Choose **Type** as dot1x and **Group Type** from the drop-down lists.
5. Check or uncheck the **Fallback to Local** check box.
6. Move the required servers from the **Available Server Groups** to the **Assigned Server Groups.**
7. Click **Apply to Device**.
8. Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**.
9. Click the mesh profile. The **Edit Mesh Profile** page is displayed.
10. Choose the **Advanced** tab.
11. In the **Security** settings, from the **Method** drop-down list, choose **EAP**.
12. Choose the options from the **Authentication Method** and **Authorization Method** drop-down lists.
13. Click **Update & Apply to Device**.

MAP authorization with EAP is now active for the selected mesh profile.

Configure MAP authorization (CLI)

Use this procedure to configure MAP authorization with EAP or PSK authentication for mesh access points using CLI commands.

Set up mesh access point (MAP) authentication for mesh access points using either Extensible Authentication Protocol (EAP) or Pre-Shared Key (PSK) methods.

Select and configure an authentication method—EAP or PSK—for MAP authentication.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Set an authentication method list to use a RADIUS server group. This is required for EAP authentication.

   Example

   ```
   Device(config)# aaa authentication dot1x method-name radius group
   server-group-name
   ```

For local authentication:

```
Device(config)# aaa authentication dot1x auth1 local
```

3. Set an authorization method list to use local credentials.

   Example

   ```
   Device(config)# wireless profile mesh profile-name local
   ```

4. Configure the mesh security EAP or PSK for mesh AP.

   Example

   ```
   Device(config-wireless-mesh-profile)# security eap server-group-name
   ```

5. Configure the authentication method for mesh AP authentication.

   Example

   ```
   Device(config-wireless-mesh-profile)# method authentication method-name
   ```

You have configured MAP authorization for mesh access points using your chosen authentication method (EAP or PSK).

```
Device# configure terminal
Device(config)# aaa authentication dot1x auth1 local
Device(config)# wireless profile mesh mesh1
Device(config-wireless-mesh-profile)# security eap / psk
Device(config-wireless-mesh-profile)# method authentication auth1
```

Preshared key provisioning

PSKs allow for controlled mesh deployments and enhance MAPs security beyond the default configuration.

A preshared key (PSK) is a mesh security credential that

- provides authentication between Mesh Access Points (MAPs), RAPs, and controllers
- allows for custom configuration to restrict network access, and
- enhances security compared to default or wildcard provisioning based on MAC addresses.

In mesh deployments, the MAPs might leave the network and join other mesh networks if both mesh deployments use AAA with wildcard MAC filtering to allow MAP association.

Since MAPs might use EAP-FAST, you cannot control this because EAP uses a combination of MAC address and AP type for authentication, and there are no controlled configuration options available. If you use the PSK option with a default pass phrase, you create a security risk.

This issue frequently occurs in overlapping deployments of two service providers when MAPs are used in moving vehicles, such as ferries, ships, or other public transportation.

In these scenarios, your MAPs are not restricted to your service provider mesh network. As a result, another service provider's network can hijack MAPs, preventing them from serving your customers.

The PSK key provisioning feature enables network administrators to assign unique preshared keys from the controller to MAPs. This ensures that only authorized MAPs can authenticate to specific RAPs and controllers. These measures prevent accidental or malicious association with unauthorized mesh networks and protect service provider deployments.

Configure PSK provisioning (CLI)

Use this procedure to configure a pre-shared key (PSK) for wireless mesh access points using CLI commands. This enables secure authentication for mesh APs, and streamlines the provisioning process.

When you enable PSK provisioning, your APs join using the default PSK. After you set the PSK provisioning key, the system pushes the configured key to each newly joined AP.

Complete these steps to configure a PSK:

Before you begin

Confirm that you have pushed the provisioned PSK to every AP configured with PSK as mesh security.

Note

- PSKs are saved across reboots in the controller as well as on the corresponding mesh AP.
- A controller can have a total of five PSKs and one default PSK.
- A mesh AP deletes the provisioned PSK only during a factory reset.
- A mesh AP never uses the default PSK after receiving the first provisioned PSK.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the security method for wireless as PSK.

   Example

   ```
   Device(config)# wireless mesh security psk provisioning
   ```

   Note

   The controller pushes the provisioned PSK only to APs configured with PSK as the mesh security method.

3. Configure a new PSK for mesh APs.

Example

```
Device(config)# wireless mesh security psk provisioning key index {0 | 8}
preshared-key description
```

4. Enable default PSK-based authentication.

Example

```
Device(config)# wireless mesh security psk provisioning default-psk
```

5. Specify the PSK to be actively used.

Example

```
Device(config)# wireless mesh security psk provisioning inuse index
```

> **Note**
>
> You should explicitly specify the in-use key index in the global configuration to point to the PSK index.

The controller provisions the specified pre-shared key on the mesh APs, sets the active key, and distributes it accordingly.

```
Device# configure terminal
Device(config)# wireless mesh security psk provisioning
Device(config)# wireless mesh security psk provisioning key 1 0 secret
secret-key
Device(config)# wireless mesh security psk provisioning default-psk
Device(config)# wireless mesh security psk provisioning inuse 1
```

Provision LSC for bridge-mode and mesh APs (GUI)

Use this procedure to provision Local Significant Certificates (LSC) for bridge-mode and mesh APs using the graphical user interface.

Provision LSCs. This action secures communication for bridge-mode and mesh APs.

Procedure

---

1. Choose **Configuration** > **Wireless** > **Access Points** > **LSC Provision**.
2. In the **Add APs to LSC Provision List** settings, click the **Select File** option to upload a CSV file that contains AP details.
3. Click **Upload File**.
4. You can also use the **AP MAC Address** field to search for APs by MAC address and add them. The system displays APs added to the provision list in the **APs in Provision List** list.
5. Click **Apply**.
6. Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**.
7. Click **Add**.

8. In the **General** tab, enter the **Name** of the mesh profile and check the **LSC** check box.
9. In the **Advanced** tab, under the **Security** settings, choose the authorization method from the **Authorization Method** drop-down list.
10. Click **Apply to Device**.

The system successfully applies LSC provisioning to the specified bridge-mode and mesh APs.

Provision LSC for bridge-mode and mesh APs

Provision Locally Significant Certificates (LSC) for bridge-mode and mesh Access Points (APs) using CLI commands.

- Configuring a Locally Significant Certificate (LSC) will not remove preexisting certificates from an AP.
- An AP can have both LSC and Message Integrity Check (MIC) certificates. However, when you provision an AP with LSC, the MIC certificate is not used when the AP starts. If you switch from LSC to MIC, reboot the AP.

To configure LSC for bridge-mode and mesh APs, use this procedure:

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure LSC provisioning for an AP.

   Example

   ```
   Device(config)# ap lsc-provision
   ```

   Note

   This step is applicable only for mesh APs.

3. Optional: Configure LSC provisioning for all APs in the provision list.

   Example

   ```
   Device(config)# ap lsc-provision provision-list
   ```

4. Configure a named authorization list to download EAP credentials from the RADIUS group server.

   Example

   ```
   Device(config)# aaa authentication dot1x auth-list radius group
   radius-server-grp
   ```

5. Create a mesh profile to enter mesh profile configuration mode.

   Example

   ```
   Device(config)# wireless profile mesh mesh-profile-name
   ```

6. Set mesh security for LSC-only MAP authentication.

   Example

   ```
   Device(config-wireless-mesh-profile)# lsc-only-auth
   ```

   All the mesh APs reboot after you run this command.

7. Configure an authorization method for mesh APs.

   Example

   ```
   Device(config-wireless-mesh-profile)# method authorization local
   ```

All bridge-mode and mesh APs are provisioned with Locally Significant Certificates (LSC).

```
Device# configure terminal
Device(config)# ap lsc-provision
Device(config)# ap lsc-provision provision-list
Device(config)# aaa authentication dot1x list1 radius group sg1
Device(config)# wireless profile mesh mesh1
Device(config-wireless-mesh-profile)# lsc-only-auth
Device(config-wireless-mesh-profile)# method authorization list1
```

Intrusion detection system

This concept explains how intrusion detection systems protect computer networks by monitoring, detecting, and responding to unauthorized or malicious activities to enhance network security.

Cisco Intrusion Detection System/Intrusion Prevention System (CIDS/CIPS) is a network security solution that

- monitors network traffic and system activities for signs of suspicious behavior
- detects and classifies potential security threats such as malware or unauthorized access, and
- responds to attacks or alerts administrators to enable prevention or mitigation actions.

CIDS can block specific clients from your wireless network if it detects attacks involving them in network layers 3 to 7. This feature helps you detect, classify, and stop threats such as worms, spyware, adware, network viruses, and application abuse.

Configure the intrusion detection system (CLI)

Learn how to configure intrusion detection system reporting for mesh APs using CLI commands.

Enable intrusion detection system monitoring and reporting on mesh APs to enhance network security.

When enabled, the intrusion detection system generates reports for all client access traffic. This feature does not apply to backhaul traffic.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a mesh profile and enter mesh profile configuration mode.

   Example

   ```
   Device(config)# wireless profile mesh profile-name
   ```

3. Configure intrusion detection system reporting for mesh APs.

   Example

   ```
   Device(config-wireless-mesh-profile)# ids
   ```

The device is now configured to report intrusion detection events for mesh APs.

```
Device# configure terminal
Device(config)# wireless profile mesh mesh-ids
Device(config-wireless-mesh-profile)# ids
```

Bridge group names

Explains how bridge group names organize mesh APs and manage parent-child associations, including procedures for configuring bridge group names through both GUI and CLI interfaces.

A bridge group name (BGN) is a mesh network configuration parameter that

- controls the association of Mesh Access Points (MAPs) to a parent mesh AP
- allows logical grouping of radios to isolate different networks on the same channel, and
- allows MAPs to join networks before you assign a custom BGN.

BGNs can logically group radios to avoid two networks on the same channel from communicating with each other. The setting is useful when your network contains multiple RAPs in the same sector (area). A BGN is a string containing up to 10 characters.

A BGN of *NULL VALUE* is assigned by default during manufacturing. This value is not visible to you, but it allows a MAP to join the network before you assign a network-specific BGN.

If you have two RAPs in your network in the same sector, we recommend that you configure the two RAPs with the same BGN, on different channels.

When Strict Match BGN is enabled on a MAP, the MAP scans ten times to find a matching BGN parent. If the AP does not find a parent with a matching BGN after ten scans, it connects to a nonmatching BGN and maintains the connection for 15 minutes. After 15 minutes, the AP scans ten times again, and this cycle continues. The default BGN functionality does not change when Strict Match BGN is enabled.

In Cisco Catalyst 9800 Series Wireless Controller, the BGN is configured on the mesh profile. When a MAP joins the controller, the system pushes the configured BGN on the mesh profile to the AP.

Preferred parent selection

A preferred parent selection is a mesh network configuration method that

- enables enforcement of linear topology in a mesh environment
- allows administrators to override the AWPP-defined (Adaptive Wireless Path Protocol) parent selection algorithm, and
- supports explicit specification of the uplink path for the MAP in mesh deployments.

For Cisco Wave 1 APs, when you configure a preferred parent, ensure that you specify the MAC address of the actual mesh neighbor for the desired parent. This MAC address is the base radio MAC address that has the letter "f" as the final character. For example, if the base radio MAC address is 00:24:13:0f:92:00, then you must specify 00:24:13:0f:92:0f as the preferred parent.

```
Device# ap name ap1 mesh parent preferred 00:24:13:0f:92:0f
```

For Cisco Wave 2 APs, when you configure a preferred parent, the MAC address is the base radio MAC address that has "0x11" added to the last two characters. For example, if the base radio MAC address is 00:24:13:0f:92:00, then you must specify 00:24:13:0f:92:11 as the preferred parent.

```
Device# ap name ap1 mesh parent preferred 00:24:13:0f:92:11
```

Configure a bridge group name (GUI)

Configure a bridge group name using GUI.

Configure a bridge group name to organize and manage wireless mesh network profiles efficiently.

Procedure

1. Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**.
2. Click **Add**.
3. In the **Advanced** tab, under the **Bridge Group** settings, enter the **Bridge Group Name**.
4. Under the **Bridge Group** settings, check the **Strict Match** check box to enable the feature. The MAP scans ten times to find a matching BGN parent when you enable **Strict Match** BGN.
5. Click **Apply to Device**.

The system assigns and applies a new bridge group name to the mesh profile you selected.

Configure a bridge group name (CLI)

Use this procedure to configure a bridge group name on an AP using CLI commands.

Configure a bridge group name (BGN) for a mesh profile on a wireless LAN controller using CLI commands.

- If a BGN is configured on a mesh profile, whenever a MAP joins the controller, it pushes the BGN configured on the mesh profile to the AP.
- Whenever a mesh AP moves from AireOS controller to the Cisco Catalyst 9800 Series Wireless Controller, the BGN configured on the mesh profile is pushed to that AP and stored there.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a mesh profile and enter mesh profile configuration mode.

   Example

   ```
   Device(config)# wireless profile mesh mesh-profile-name
   ```

3. Configure a bridge group name.

   Example

   ```
   Device(config-wireless-mesh-profile)# bridge-group name bridge-grp-name
   ```

4. Configure bridge group strict matching.

   Example

   ```
   Device(config-wireless-mesh-profile)# bridge-group strict-match
   ```

The bridge group name is assigned to the mesh profile and applied to all associated mesh APs.

```
Device# configure terminal
Device(config)# wireless profile mesh mesh-profile-bgn
Device(config-wireless-mesh-profile)# bridge-group name bgn-grp-name
Device(config-wireless-mesh-profile)# bridge-group strict-match
```

Mesh background scanning

Describes background channel scanning and MAP fast ancestor find mode for rapid failover, including configuration and verification procedures through both GUI and CLI.

A mesh background scan is a mesh networking feature that

- enables mesh access points (MAPs) to find and connect to better potential parents across channels
- maintains stable uplinks by proactively scanning for available parent nodes, and
- reduces reconnection delay after parent loss by using a neighbor list instead of scanning all channels.

When background scanning is disabled, a MAP has to scan all the channels of the regulatory domain after detecting a parent loss in order to find a new parent and go through the authentication process. This delays the time taken for the mesh AP to connect back to the controller.

When background scanning is enabled, a MAP can avoid scanning across the channels to find a parent after detecting a parent loss, and select a parent from the neighbor list and establish the Adaptive Wireless Path Protocol (AWPP) link.

Background scanning and MAP fast ancestor find mode

Describes mechanisms that optimize MAP connectivity and parent selection in Cisco mesh wireless networks.

Background scanning and MAP fast ancestor finding are features that

- optimize MAP connectivity and parent selection in Cisco mesh wireless networks
- minimize mesh tree convergence time when MAPs lose connectivity with their parents, and
- reduce the need for beacon transmission during network formation.

Background scanning and MAP fast ancestor finding details

Cisco mesh access points (MAPs) perform the following functions:

- Interconnects over wireless links in a tree topology,
- Uses Adaptive Wireless Path Protocol (AWPP) to create and maintain their topology, and
- Supports additional features: Background Scanning and MAP Fast Ancestor Finding.

When a MAP comes up, it tries to look for another MAP (parent) to join and reach the gateway through a RAP. The same happens when a MAP loses connectivity with its existing parent. This procedure is known as mesh tree convergence.

The Background scanning feature:

- Updates MAPs about neighboring channels and helps find new parents swiftly by scanning all available channels.
- Minimizes the time spent during scan-and-seek phases when a MAP loses its current parent.
- Does not speed up the authentication process to the new parent.

A child MAP maintains its uplink with its parent by using the AWPP adjacency request/response messages, which act as keepalive signals. If consecutive response messages are lost, the parent is considered lost, and the child MAP searches for a new parent. A MAP maintains a list of neighbors ON the current ON channel. If the AP loses its current parent, it roams to the next best potential neighbor. If no other neighbors are found, the AP scans or seeks across all the channels or subset channels to find a parent. This process is time-consuming.

The MAP Fast Ancestor Finding feature enables a method to reduce the need for sending or receiving beacons during network formation, while starting or deploying a new mesh network.

Configure background scanning (GUI)

Configure background scanning using GUI.

Enable background scanning on a wireless mesh profile to improve network performance and reliability.

Procedure

1. Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**.
2. Select your desired profile.
3. In **General** tab, check the **Background Scanning** check box.
4. Click **Update & Apply to Device**.

You have activated background scanning for the selected wireless mesh profile.

Configure background scanning

Use this procedure to configure background scanning in mesh deployments so that mesh access points (MAPs) can scan neighboring channels to find better parent nodes and maintain optimal uplink connections.

To enable background scanning in mesh deployments so that mesh access points (MAPs) can detect neighboring channels and identify better parent nodes, maintaining optimal uplink connections.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a mesh profile and enter the mesh profile configuration mode.

   Example

   ```
   Device(config)# wireless profile mesh mesh-profile-name
   ```

3. Configure background scanning to enable mesh access points to find better parent nodes and maintain optimal connections.

   Example

   ```
   Device(config-wireless-mesh-profile)# background-scanning
   ```

Background scanning is enabled in the mesh profile. Mesh APs scan neighboring channels to find better parent nodes. This helps maintain optimal uplink performance in the mesh network.

```
Device# configure terminal
Device(config)# wireless profile mesh mesh-profile-name
Device(config-wireless-mesh-profile)# background-scanning
```

Configure AP fast ancestor find mode (GUI)

Configure AP fast ancestor find mode using GUI.

Enable a child MAP to synchronize with any neighbor parent MAP across all channels.

Use the GUI to configure the MAP Fast Ancestor Find feature within a mesh profile.

Follow these steps to configure AP fast ancestor find mode through the GUI:

Procedure

1. Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**.
2. Click **Add**.
   The **Add Mesh Profile** page is displayed.
3. In the **Add Mesh Profile** page, click the **General** tab.

4. In the **Name** field, enter the mesh profile name.
5. In the **Description** field, enter a description for the mesh profile.
6. Check the **MAP Fast Ancestor Find** check box to enable a MAP (child) to synchronize with any neighbor MAP (parent) across all channels.
7. Click **Apply to Device** to save the configuration.

The MAP Fast Ancestor Find feature is enabled for the specified mesh profile.

Configure background scanning and MAP fast ancestor find mode (Task)

Configure background scanning and MAP fast ancestor find mode using commands.

Configure background scanning and MAP fast ancestor find mode using the CLI within a mesh profile for detailed configuration options.

Follow these steps to configure background scanning and MAP fast ancestor find mode through the CLI:

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a mesh profile and enter mesh profile configuration mode.

   Example

   ```
   Device# wireless profile mesh default-mesh-profile
   ```

3. Enable background scanning in mesh deployments.

   Example

   ```
   Device(config-wireless-mesh-profile)# background-scanning
   ```

   > **Note**
   >
   > In Cisco Catalyst 9124 Series Access Points, a dedicated RF ASIC radio is used for background scanning.

4. Enable fast ancestor find mode.

   Example

   ```
   Device(config-wireless-mesh-profile)# map-fast-ancestor-find
   ```

Background scanning and MAP fast ancestor find mode are enabled for the specified mesh profile.

Example

```
Device# configure terminal
Device# wireless profile mesh default-mesh-profile
Device(config-wireless-mesh-profile)# background-scanning
Device(config-wireless-mesh-profile)# map-fast-ancestor-find
```

Verify background scanning and MAP fast ancestor find

Provides verification methods for background scanning and MAP Fast Ancestor Find features.

To verify whether the Background Scanning and MAP Fast Ancestor Find features are enabled, run the **show wireless profile mesh detailed** command.

To verify the background scan, use this command:

```
Device# show wireless profile mesh detailed Mesh_Profile | i Background Scan
Background Scan               : ENABLED
```

To verify MAP Fast Ancestor Find, use this command:

```
Device# show wireless profile mesh detailed Mesh_Profile | i MAP fast ancestor
 find
MAP fast ancestor find        : ENABLED
```

Mesh serial backhaul

Covers wireless and serial backhaul operation, client access, radio selection, enabling serial backhaul in radio profiles, and verification for robust mesh connectivity options.

A mesh serial backhaul is a wireless mesh networking feature that

- enables separate uplink and downlink channels on dedicated radios to improve backhaul bandwidth
- allows inbound and outbound traffic to use exclusive communication paths for better performance (one radio is used as the uplink radio and a different one is used as the downlink radio), and
- supports universal access by extending network reach and avoiding shared-channel interference.

The Mesh Serial Backhaul feature is supported in the controller from Cisco IOS XE Cupertino 17.7.1 onwards for Cisco Catalyst 9124AXE outdoor APs. To enable this feature, associate a radio profile with a radio frequency (RF) tag. When you enable this feature, all APs with the same mesh profile share the mesh configuration. All APs with the same radio profile share the radio configuration.

The 2.4-GHz and 5-GHz radios, which are not used in serial backhaul, provide basic client access functionality. The downlink radio provides universal access.

> **Note**
>
> Slot 1 and slot 0 are supported as mesh backhaul. Slot 2 is not supported as a mesh backhaul. You can utilize slot 2 for client serving or for serial backhaul downlink.

Channel assignment

For the Mesh Serial Backhaul feature, channels are assigned according to these rules:

- Uplink and downlink channels are different.
- All the 5-GHz radios maintain a frequency guard between their operating channels. For example, 100-MHz channel spacing between radios in Cisco Catalyst 9124AXE outdoor APs.
- Dynamic Frequency Selection (DFS) channels are supported.

In a root access point, the controller assigns channels because the uplink is wired. A mesh access point uses the last channel configured by the controller for this radio or uses the default channel. If the channel used by the MAP is not compatible with the uplink, the MAP picks a valid random channel and notifies the controller. In another scenario, when the MAP receives a channel change alert on the uplink radio, it randomly picks a new downlink channel. The MAP checks the validity of the downlink radio. If the current channel is not compatible, the MAP picks a random channel.

Prerequisites for channel assignment

Ensure that these prerequisites are met before channel assignment:

- Enable tri-radio globally by running the **Device# ap tri-radio** command.
- Enable the dual radio on the APs by running the **Device# ap name** *ap-name* **dot11 5ghz dual-radio mode enable**command.

Use cases

These are some of the use cases for the Mesh Serial Backhaul feature:

- **Maximize throughput** : Serial backhaul allows the 5-GHz backhaul to operate on different channels, thereby maximizing throughput over multiple mesh hops.
- **Network segregation**: APs that have serial backhaul enabled, segregate backhaul channel on mesh topographies. This avoids localized link interferences.

Enable serial backhaul in radio profile (GUI)

Enable serial backhaul in radio profile using GUI.

Enable mesh backhaul features on a specific radio profile to support serial backhaul in mesh deployments.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF/Radio** > **Radio**.
2. Click **Add** to add a radio profile.
   The **Add Radio Profile** page is displayed.
3. In the **Add Radio Profile** page, enter the name, and description.
4. In the **Mesh Backhaul** field, choose the **Enabled** radio button to enable the feature.

5. In the **Mesh Designated Downlink** field, choose the **Enabled** radio button to enable the feature.

> **Note**
>
> Mesh Designated Downlink is supported only on slot number 2 of Mesh APs. Radio profiles must be matched carefully to RF tag slots.

6. Click **Apply to Device**.

The radio profile is updated, enabling serial backhaul for selected mesh APs.

Enable mesh configurations in radio profile (CLI)

Enable mesh configurations in radio profile using commands.

Configure your Cisco wireless AP to enable mesh backhaul and designate downlink radio slots in a radio profile.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure wireless radio profile and enter the radio-profile configuration mode.

Example

```
Device(config)# wireless profile radio radio-profile-name
```

3. Enable mesh backhaul.

Example

```
Device(config-wireless-radio-profile)# mesh backhaul
```

By default, this command is enabled. Mesh backhaul can be disabled on a specific slot, to stop the specific slot from being the backhaul candidate.

4. Enable the radio slot as a designated downlink.

Example

```
Device(config-wireless-radio-profile)# mesh designated downlink
```

By default, this command is disabled. It is enabled only for slot 2 of mesh APs. If another slot is configured as the designated downlink, a warning message appears: `Designated downlink is supported only on slot 2 of mesh APs. Associate in the RF tag accordingly.`

By default, all the radio slots are mesh-enabled and not designated as downlink.

The mesh backhaul and designated downlink configurations are now applied to your selected radio profile.

Enable serial backhaul (CLI)

Learn how to configure a wireless profile to designate a radio as a mesh downlink backhaul.

Configure a wireless profile to designate a radio as a mesh downlink backhaul.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the wireless RF tag to enter wireless RF tag profile configuration mode.

   Example

   ```
   Device(config)# wireless profile radio radio-profile-name
   ```

3. Enable the specified radio as a designated mesh downlink backhaul.

   Example

   ```
   Device(config-wireless-radio-profile)# mesh designated downlink
   ```

   The designated downlink is enabled in the radio profile only for slot 2.

4. Exit the submode and return to global configuration mode.

   Example

   ```
   Device(config-wireless-radio-profile)# exit
   ```

5. Configure the wireless RF tag to enter wireless RF tag profile configuration mode.

   Example

   ```
   Device(config)# wireless tag rf rf-profile-name
   ```

   The designated downlink is enabled in the radio profile only for slot 2.

6. Configure the designated downlink radio for serial backhaul.

   Example

   ```
   Device(config-wireless-rf-tag)# dot11 5ghz {slot1 | slot2} radio-profile
   radio-profile-name
   ```

> **Note**
>
> In mesh APs, the uplink and downlink are in the same slot by default. When you configure a designated downlink, the mesh AP is forced to use a specific radio as downlink.

The specified radio is now configured as a designated mesh downlink, and the mesh AP is forced to use that radio for downlink backhaul.

Fallback Mode

> **Note**
>
> If you configure at least one radio as a designated downlink, that radio will not be available for uplink. To prevent a configuration mistake, such as configuring the uplink radio as the designated downlink, the mesh AP uses a fallback timer. If the mesh AP cannot join the controller within 10 minutes, the designated configurations are cleared, and all radios become uplink-capable.

```
Device# configure terminal
Device(config)# wireless profile radio radio-mesh-downlink
Device(config-wireless-radio-profile)# mesh designated downlink
Device(config-wireless-radio-profile)# exit
Device(config)# wireless tag rf radio profile rf-map-tag
Device(config-wireless-rf-tag)# dot11 5ghz slot2 radio-profile
radio-mesh-downlink
```

Mesh serial backhaul - configuration example

This section presents commands for configuring mesh APs to enable only slot 0 and slot 1 for backhaul use.

This example explains how to configure mesh APs to allow only slot 0 and slot 1 for backhaul connectivity:

```
Device# configure terminal
Device(config)# wireless profile radio radio-mesh-downlink
Device(config-wireless-radio-profile)# no mesh backhaul
Device(config-wireless-radio-profile)# exit
```

```
Device(config)# wireless tag rf rf-map-tag
Device(config-wireless-rf-tag)# dot11 5ghz slot2 radio-profile mesh-disabled
```

Verify mesh serial backhaul

Lists commands to verify mesh serial backhaul configuration and status.

This reference provides commands to verify mesh AP serial backhaul configuration and operational status on wireless devices.

To verify mesh AP serial backhaul, run this command:

```
Device# show ap name MAP-SB config slot 2 | inc Mesh
Mesh Radio Role : Downlink Access
Mesh Backhaul : Enabled
Mesh Designated Downlink : Enabled
```

To verify serial backhaul enabled on a specific AP, run this command:

```
Device# show ap name MAP-SB mesh backhaul
MAC Address : 4cxx.4dxx.f4xx
Current Backhaul Slot: 1
Radio Type: Main
Radio Subband: All
Mesh Radio Role: Uplink Access <<<<<<
Administrative State: Enabled
Operation State: Up
Current Tx Power Level: 6
Current Channel: (104) <<<<<<
Antenna Type:
Internal Antenna Gain (in .5 dBm units): 1
MAC Address : 4cxx.4dxx.f4xx
Current Backhaul Slot: 2
Radio Type: Slave
Radio Subband: All
Mesh Radio Role: Downlink Access <<<<<<
Administrative State: Enabled
Operation State: Up
Current Tx Power Level: 8
Current Channel: (149) <<<<<<
Antenna Type:
Internal Antenna Gain (in .5 dBm units): 1
```

To verify mesh serial backhaul, run this command:

```
Device# show wireless profile radio detailed radio-mesh-downlink
Radio Profile name                              : radio-mesh-downlink
Description                                     :
Beam-Selection                                  : Not configured
Number of antenna to be enabled                 : 0
Mesh Backhaul                                   : Enabled
Mesh Designated Downlink                        : Enabled
```

Mesh backhaul RRM

Explains how mesh backhaul RRMs optimize wireless mesh network performance by dynamically selecting the best backhaul channels and reducing interference.

A mesh backhaul RRM is a wireless network optimization mechanism that

- continuously evaluates channel conditions to identify the least congested backhaul channels
- reduces interference and enhances mesh network stability, and
- automates channel selection for root and mesh access points (APs).

Root access points (RAPs) choose backhaul channels to operate in mesh networks. Until Cisco IOS XE Cupertino 17.8.1, this operation occurred by an explicit configuration, a least congested scan during RAP boot time, during the initial radio resource management (RRM) run without mesh access points (MAPs) connected, or a backhaul channel that was chosen at random. As a result, a poor backhaul channel selection resulted in poor performance.

From Cisco IOS XE Cupertino 17.9.1 onwards, RRM DCA is run on mesh backhaul, in auto mode, in FlexConnect or centralized networks. For APs that do not have dedicated (RHL) radios, DCA is triggered by running commands in the privilege EXEC mode.

RRM continuously evaluates the channel conditions to ensure that the network utilizes the least congested channels. The network uses the transmission static power if it is configured, or falls back to the default level. This is supported on APs that have dedicated radios to scan channel conditions, without any user perceptible interruption to the mesh network traffic.

In the mesh backhaul RRM feature, the RRM DCA decides all the downlink channels in a steady network. However, if an AP detects a change in its uplink roam or radar detection response, the AP chooses the best downlink to converge faster.

APs choosing the best possible downlink is limited to serial backhaul enabled APs only.

To avoid a poor channel backhaul selection, from Cisco IOS XE Dublin 17.14.1 onwards, the RRM DCA optimizes the RAP backhaul radio channel of a mesh subtree by considering the noise, interference, load, and the RF parameter measurements only from the RAP. The RRM DCA on Mesh Backhaul feature enables DCA to make better channel assignment for a mesh subtree, by having continuous measurements and inputs from the whole mesh tree required to run DCA.

To enable RRM DCA on a full mesh tree run the **wireless mesh backhaul rrm auto-dca** command. To trigger DCA once, run the **ap dot11 [5ghz | 2.4 ghz] rrm channel-update mesh** command.

To understand what happened during a DCA run on the mesh backhaul, use the **show wireless mesh rrm dca status** and **show wireless mesh rrm dca changed** commands.

Note

- In a topology with mixed APs (RF ASIC capable and non-capable APs), only inputs from the RF ASIC capable APs apply for **auto-dca** .
- All mesh APs in a subtree should be configured to belong to the same site-tag for DCA to work properly.
- This feature is limited to RF ASIC capable APs, such as Cisco Catalyst 9124 Series APs and Cisco Catalyst 9130 Series APs.

Radio Resource Management on mesh

This concept explains how Radio Resource Management (RRM) acts as a built-in RF engineer to provide real-time RF management of a wireless network.

Radio resource management (RRM) features are wireless network management functions that

- enable real-time RF management of the wireless network
- monitor lightweight APs for traffic load, interference, noise, and coverage, and
- operate automatically through embedded controller software.

The RRM measurement in the mesh AP backhaul is enabled based on these conditions:

- Mesh AP has the Root AP role.
- Root AP has joined using Ethernet link.
- Root AP does not serve any child AP.

Configure RRM on mesh backhaul (CLI)

Use this task to enable Radio Resource Management (RRM) on the mesh backhaul interface of your wireless access points using CLI commands. This configuration allows mesh root access points with Ethernet uplink and no child APs to report RRM noise information for optimized network management.

RRM measurement in the mesh AP backhaul is enabled if your access point meets these conditions:

- Mesh AP has the Root AP role.
- The Root AP has joined the network using an Ethernet link.
- Root AP does not serve any child AP.

Complete the steps to enable RRM in the mesh backhaul:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure RRM on the mesh backhaul.

   Example

   ```
   Device(config)# wireless mesh backhaul rrm
   ```

RRM is enabled on the mesh backhaul for APs.

```
Device# configure terminal
Device(config)# wireless mesh backhaul rrm
```

Configure RRM channel assignment for root access points globally

Use these tasks to configure the channel assignment of the RAP in privileged EXEC mode and globally through RRM.

Configure Radio Resource Management (RRM) channel assignment policies for all root access points. This configuration optimizes wireless performance, minimizes interference, and maintains network stability across your deployment.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure RRM for mesh backhaul.

Example

```
Device(config)# wireless mesh backhaul rrm
```

3. Optional: Configure auto DCA for RF Application Specific Integrated Circuit (ASIC) integrated RAPs.

Example

```
Device(config)# wireless mesh backhaul rrm auto-dca
```

All RAPs adopt the configured RRM channel assignment settings globally. This change results in improved channel distribution, reduced interference, and more reliable wireless connectivity across the network.

```
Device# configure terminal
Device(config)# wireless mesh backhaul rrm
Device(config)# wireless mesh backhaul rrm auto-dca
```

Complete these steps to configure the initial channel assignment of the RAP in privileged EXEC mode through RRM and to initiate channel selection for each bridge group.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device> enable
```

2. Initiate the update of the 802.11, 802.11a, or 802.11b channel selection for every mesh Cisco AP.

Example

```
Device# ap dot11 {24ghz | 5ghz | 6ghz} rrm channel-update mesh
```

3. Initiate the update of the 802.11, 802.11a, or 802.11b channel selection for mesh AP in the bridge group.

Example

```
Device# ap dot11 {24ghz | 5ghz | 6ghz} rrm channel-update mesh bridge-group
bridge-group-name
```

RRM completes the channel assignment for the RAP in privileged EXEC mode.

```
Device> enable
Device# ap dot11 5ghz rrm channel-update mesh
Device# ap dot11 5ghz rrm channel-update mesh bridge-group cisco-bridge-group
```

Configure RRM channel assignment for an AP

Use this task to manually trigger Radio Resource Management (RRM) channel assignment for a specific access point (AP) to optimize wireless performance.

Assign RRM channels to an AP. This action helps you optimize wireless channel usage and reduce interference.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

2. Trigger the RRM DCA process for your AP.

   Example

   ```
   Device# ap name Cisco-ap-name dot11 {24ghz | 5ghz | 6ghz} rrm channel update
    mesh
   ```

After you trigger the process, the system starts the RRM DCA process and updates the channel assignment for your selected radio band.

```
Device> enable
Device# ap name Cisco-ap-name dot11 5ghz rrm channel update mesh
```

RRM DCA status verification

Lists commands to verify the RRM DCA status.

Verify the RRM DCA status for mesh access points using specific commands and understand the output differences between manual and automatic DCA triggers.

To view the status of the DCA that is run for mesh APs, run this command:

```
Device# show ap name Cisco-AP config general | inc Mesh
Mesh profile name                               : default-mesh-profile
Mesh DCA Run Status:                            : Not Running
Last Mesh DCA Run                               : 02/07/2022 01:21:56
```

To verify the status of the last DCA run per radio, run this command:

```
Device# show wireless mesh rrm dca status
```

> **Note**
>
> The output for the **show AP config general | i Mesh** and the **show AP name <AP name> config general | i Mesh** command displays only the status for manual RRM DCA triggers performed with the **AP name <AP name> dot11 RRM channel update mesh** command.
>
> The output for the **show AP config general | i Mesh** and the **show AP name <AP name> config general | i Mesh** commands, does not update if only the global mesh RRM DCA is enabled (auto-DCA).

Dynamic frequency selection

This concept explains Dynamic Frequency Selection (DFS) and how it protects radar services by requiring compliant radio devices to detect, avoid, and switch away from radar-occupied frequencies.

A dynamic frequency selection (DFS) is a wireless communication protocol that

- enables radio devices to detect radar signals
- requires devices to cease transmission when radar is detected, and
- requires selecting and monitoring a new channel before resuming transmission.

Regulatory bodies enforce DFS to prevent interference with radar services in shared frequency bands by unlicensed wireless devices.

To protect radar services, regulatory bodies require devices on newly opened frequency sub-bands to operate using DFS. Your radio device must detect radar signals as required. If a radar event is detected in any AP within a sector, mesh access points immediately switch channels to maintain compliance.

For instance, When a radio detects a radar signal, the radio should stop transmitting for at least 30 minutes to protect that service. The radio should then select a different channel to transmit on, but only after monitoring it. If no radar is detected on the projected channel for at least one minute, the new radio service device can begin transmissions on that channel.

Configure dynamic frequency selection (CLI)

Use this procedure to configure dynamic frequency selection (DFS) for mesh profiles using command-line interface commands.

Configure dynamic frequency selection (DFS) on your device to enable radar detection and meet DFS requirements.

DFS specifies the types of radar waveforms that should be detected and the timers that must be used for operation in the DFS channel.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a mesh profile and enter mesh profile configuration mode.

Example

```
Device(config)# wireless profile mesh profile-name
```

3. Enable DFS.

Example

```
Device(config-wireless-mesh-profile)# full-sector-dfs
```

> **Note**
>
> DFS functionality allows a MAP that detects a radar signal to transmit this information to the RAP. The RAP treats this as a radar detection and moves the sector. This process is called the coordinated channel change. The coordinated channel change feature is always enabled on Cisco Wave 2 and newer APs. You can disable this feature only on Cisco Wave 1 APs.

Dynamic Frequency Selection is enabled for your mesh profile. Your mesh profile is now ready to operate and meets radar detection requirements.

```
Device# configure terminal
Device(config)# wireless profile mesh dfs-mesh-profile
Device(config-wireless-mesh-profile)# full-sector-dfs
```

Spectrum Intelligence for mesh

This concept explains Spectrum Intelligence features for mesh networks, including how they detect non-Wi-Fi interference, leverage CleanAir technology, and support radio management and mitigation in client and backhaul radio bands

A Spectrum Intelligence feature is a wireless network monitoring capability that

- scans 2.4 GHz and 5 GHz bands for non-Wi-Fi radio interference
- provides device and air quality reports using Cisco CleanAir technology, and
- includes mitigation features like Event-Driven Radio Resource Management (EDRRM) and Persistent Device Avoidance (PDA) that require CleanAir data.

The Spectrum Intelligence feature supports both client serving and monitor modes. CleanAir technology in mesh backhaul and access radios generates Interference Device Reports (IDR) and Air Quality Index (AQI). In mesh networks, you use mitigation features with client-access radios in the same way as with non-mesh networks. However, in backhaul radios, reports are controller-only and do not trigger ED-RRM actions. You cannot enable or disable CleanAir on mesh APs (MAPs) because configuration options are not available.

For more information about Spectrum Intelligence, see Configure spectrum intelligence (CLI) on page 2344 section.

Mesh backhaul at 2.4 GHz and 5 GHz

Explains how mesh backhauls connect wireless devices and affect throughput in network environments.

A backhaul is a wireless network interface that

- creates the connection between Mesh Access Points (MAPs)
- operates over 802.11a/n/ac/g, depending on the AP, and
- typically defaults to the 5-GHz frequency band.

Selecting the appropriate backhaul rate is important for efficient spectrum use. It can directly affect client device throughput, a critical metric in evaluating wireless performance.

Mesh backhaul is supported at 2.4 GHz and 5 GHz. By default, the backhaul interface for mesh APs uses 802.11a/ac/ax. In some countries, mesh networks cannot use a 5 GHz backhaul. In countries where 5 GHz is allowed, using 2.4 GHz radio frequencies achieves more extensive mesh or bridge distances.

When a RAP receives a slot-change configuration, the RAP propagates it to all child MAPs. All MAPs disconnect and join the newly configured backhaul slot.

For information about APs that support mesh backhaul, see https://www.cisco.com/c/en/us/td/docs/wireless/access_point/feature-matrix/ap-feature-matrix.html .

Note

In Israel, you must ensure that you run the **ap country IO** command to enable the outdoor country code for the selected radio. After you configure using the **ap country IO** command, the 2.4-GHz radio is enabled and 5-GHz radio is disabled.

Configure mesh backhaul (CLI)

Configure mesh backhaul using commands.

Set the mesh backhaul frequency for an access AP using CLI commands.

Procedure

Change the mesh backhaul to 2.4 GHz.

Example

```
Device # ap name test-ap mesh backhaul radio dot11 24ghz
```

The specified access point now uses 2.4 GHz as its mesh backhaul radio frequency.

Specify the backhaul slot for the root AP (GUI)

Use this task to configure the backhaul slot for a root access point using the web interface.

Configure the backhaul slot for a root access point to optimize wireless mesh connectivity and ensure reliable network communication.

Procedure

1. Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**.

2. Click **Add**.
3. In the **General** tab, enter the **Name** of the mesh profile.
4. In the **Advanced** tab, select the rate types for **5 GHz Band Backhaul** and **2.4 GHz Band Backhaul** from the **Rate Types** drop-down list.
5. Click **Apply to Device**.

The system updates the root access point with the specified backhaul slot settings. It uses the selected rate types for 5 GHz and 2.4 GHz band backhaul connections.

Specify the backhaul slot for the root AP (CLI)

Learn how to specify the mesh backhaul slot for a root AP using Command-Line Interface (CLI) commands to optimize wireless mesh connectivity.

Assign a specific radio slot (2.4 GHz or 5 GHz) to the mesh backhaul of your root access point to optimize wireless connectivity and network performance.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

2. Specify the mesh backhaul radio slot for your root AP.

   Example

   ```
   Device# ap name ap-name mesh backhaul radio dot11 {24ghz | 5ghz} [slot slot-id]
   ```

Assign a radio slot (2.4 GHz or 5 GHz) to the mesh backhaul of your root access point to improve wireless connectivity and network performance.

```
Device> enable
Device# ap name rap1 mesh backhaul radio dot11 24ghz slot 2
```

Mesh backhaul verification

Provides commands to verify mesh backhaul configuration and status.

Use the **show AP name mesh backhaul** command to view details of the mesh backhaul at 2.4 GHz.

```
Device# show ap name test-ap mesh backhaul

MAC Address : xxxx.xxxx.xxxx
Current Backhaul Slot: 0
Radio Type: 0
Radio Subband: All
Mesh Radio Role: DOWNLINK
Administrative State: Enabled
Operation State: Up
```

```
Current Tx Power Level:
Current Channel: (11)
Antenna Type: N/A
Internal Antenna Gain (in .5 dBm units): 0
```

Use the **show wireless mesh AP backhaul** command to view the mesh backhaul details.

```
Device# show wireless mesh ap backhaul

MAC Address : xxxx.xxxx.0x11
Current Backhaul Slot: 1
Radio Type: Main
Radio Subband: All
Mesh Radio Role: Downlink
Administrative State: Enabled
Operation State: Up
Current Tx Power Level: 6
Current Channel: (100)*
Antenna Type: N/A
Internal Antenna Gain (in .5 dBm units): 10
```

Use the **show AP summary** command to view the radio MAC address and the corresponding AP name.

```
Device# show ap summary
Number of APs: 1
AP Name    Slots  AP Model         Ethernet       MAC Radio MAC  Location
     Country IP Address   State
-------------------------------------------------------------------------------
AP-Cisco-1  2     AIR-APXXXXX-E-K9 xxxx.xxxx.xxd4 xxxx.xxxx.0x11 default
location  DE     10.11.70.170 Registered
```

Backhaul client access

This concept explains how enabling Backhaul Client Access allows wireless client association over the backhaul radio.

Backhaul client access is a wireless networking feature that

- allows wireless clients to associate with mesh APs using the backhaul radio
- supports both 2.4 GHz and 5 GHz backhaul radios, and
- permits the backhaul radio to carry both client traffic and backhaul traffic simultaneously.

When backhaul client access is disabled, only backhaul traffic is sent over the backhaul radio, and client association is performed only over the access radio.

> **Note**
>
> Backhaul client access is disabled by default. After you enable backhaul client access, all MAPs except the subordinate AP and its child APs in a daisy-chained deployment reboot.

Configure backhaul client access (GUI)

Configure backhaul client access using GUI.

Enable backhaul client access for mesh device profiles.

Procedure

1. Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**.
2. Choose a profile.
3. In **General** tab, check the **Backhaul Client Access** check box.
4. Click **Update & Apply to Device**.

The system updates the profile and enables backhaul client access for devices that use the selected mesh profile.

Configure backhaul client access (CLI)

Configure backhaul client access using commands.

Enable backhaul client access on a mesh profile. This allows client devices to connect to the APs through the mesh backhaul. The result is increased network flexibility and coverage.

> **Note**
>
> Backhaul client access is disabled by default. After you enable it, all MAPs reboot, except for the subordinate AP and its child APs in a daisy-chained deployment.

Complete these steps to enable backhaul client access on a mesh profile:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a mesh profile and enter mesh profile configuration mode.

   Example

   ```
   Device(config)# wireless profile mesh profile-name
   ```

3. Configure backhaul with client access AP.

   Example

   ```
   Device(config-wireless-mesh-profile)# client-access
   ```

You have enabled backhaul client access on the selected mesh profile.

```
Device# configure terminal
Device(config)# wireless profile mesh profile-name
Device(config-wireless-mesh-profile)# client-access
```

Configure dot11ax rates on mesh backhaul per AP (GUI)

Configure Dot11ax rates on mesh backhaul per access point using GUI.

This task enables you to configure specific dot11ax (Wi-Fi 6) rates for mesh backhaul connections on individual APs.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
   The **All Access Points** section, which lists all the configured APs in the network, is displayed with their corresponding details.
2. Select the mesh AP that has been configured.
   The **Edit AP** page is displayed.
3. Choose the **Mesh** tab.
4. In the **General** section, under the **Backhaul** section, the default **Backhaul Radio Type** , **Backhaul Slot ID** , and **Rate Types** field details are displayed. Note that the values for **Backhaul Radio Type** and **Backhaul Slot ID** can be changed only for a root AP.
5. From the **Rate Types** drop-down list, choose the backhaul rate type.

   Based on the choice, enter the details for the corresponding fields that are displayed.The backhaul interface varies between auto and the 802.11a/b/g/n/ac/ax rates, depending on the AP. The Cisco Catalyst 9124AX Outdoor AP is the only AP that supports 11ax backhaul rates on the mesh backhaul.
6. In the **Backhaul MCS Index** field, enter the Modulation Coding Scheme (MCS) rate, that can be transmitted between the APs. The valid range is from 0 to 11, on both bands.
7. In the **Spatial Stream** field, enter the number of spatial streams that are supported. The maximum number of spatial streams supported on a single radio in a 5 GHz radio band is 8, while 2.4 GHz radio band supports 4 spatial streams.
8. Click **Update and Apply to Device**.

After you complete the steps, the selected mesh access point operates using the configured dot11ax rates for its backhaul link.

Configure Dot11ax rates on mesh backhaul in mesh profile (GUI)

Configure Dot11ax rates on mesh backhaul in mesh profile using the GUI to optimize data transmission between access points.

This task allows you to configure Dot11ax rates on mesh backhaul in mesh profile to optimize data transmission between access points using advanced wireless standards.

Use this procedure when you need to configure mesh backhaul rates using the Dot11ax standard. This configuration is specifically supported on Cisco Catalyst 9124AXI/D Series outdoor Access Points and allows for enhanced wireless mesh networking performance.

Procedure

1. Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**.
2. Click **Add**.

The **Add Mesh Profile** window is displayed.

3. In the **Add Mesh Profile** window, click the **General** tab.
4. In the **Name** field, enter the mesh profile name.
5. Click the **Advanced** tab.
6. In the **5 GHz Band Backhaul** section and the **2.4 GHz Band Backhaul** section, choose the **dot11ax** backhaul rate type from **Rate Types** the drop-down list.

> **Note**
>
> Cisco Catalyst 9124AXI/D Series outdoor Access Point is the only AP to support 11ax backhaul rates on the mesh backhaul.

7. In the **Dot11ax MCS index** field, specify the MCS rate at which data can be transmitted between the APs. The value range is between 0 to 11, on both the radio bands.
8. In the **Spatial Stream** field, enter a value. The maximum number of spatial streams supported on a single radio in a 5-GHz radio band is 8, while 2.4- GHz radio band supports 4 spatial streams.
9. Click **Update and Apply to Device**.

The Dot11ax rates are now configured on the mesh backhaul in the mesh profile. The mesh network will use the specified MCS rates and spatial stream configuration for data transmission between access points.

Configure wireless backhaul data rate (CLI)

Use this procedure to configure the wireless backhaul data rate for access points using CLI commands, which helps optimize network coverage and performance based on your deployment requirements.

Configure the data transmission rate for wireless backhaul connections between APs using CLI commands to optimize wireless network performance, coverage, and spectrum use.

Use backhaul to create a wireless connection between APs. Depending on the AP, the backhaul interface can be 802.11bg, 802.11a, 802.11n, or 802.11ac. Selecting a rate lets you use the available RF spectrum effectively.

Data rates can also affect the RF coverage and network performance. Lower data rates, for example, 6 Mbps, can extend farther from the AP than higher data rates, for example, 1300 Mbps. As a result, the data rate affects cell coverage, and consequently, the number of APs required.

> **Note**
>
> You can configure backhaul data rate, preferably, through the mesh profile. In certain cases, where a specific data rate is needed, use the command to configure the data rate per AP.

Follow this procedure to configure wireless backhaul data rate in privileged EXEC mode or in mesh profile configuration mode.

Procedure

1. Enter privileged EXEC mode.

Example

```
Device> enable
```

2. Configure backhaul transmission rate.

Example

```
Device# ap name ap-name mesh backhaul rate {auto | dot11abg | dot11ac | dot11n}
```

3. Configure a mesh profile and enter mesh profile configuration mode.

Example

```
Device(config)# wireless profile mesh profile-name
```

4. Configure backhaul transmission rate.

Example

```
Device(config-wireless-mesh-profile)# backhaul rate dot11  { 24ghz | 5ghz |
6ghz} dot11n RATE_6M
```

> **Note**
>
> Ensure the rate you configure on the AP (step 2) matches the rate you configure on the mesh profile (step 4).

After you configure the wireless backhaul data rate, your APs will balance coverage area and performance for your deployment.

```
Device> enable
Device# ap name ap1 mesh backhaul rate auto
Device(config)# wireless profile mesh mesh1
Device(config-wireless-mesh-profile)# backhaul rate dot11 5ghz dot11n mcs 31
```

Configure data rate per AP (CLI)

Learn how to configure data rates for each access point using CLI commands to optimize wireless network performance

Configure data transmission rates for each AP on mesh backhaul for the 2.4 GHz and 5 GHz bands using the command-line interface.

Procedure

1. Enter privileged EXEC mode.

Example

```
Device> enable
```

2. Configure the 802.11ax mesh backhaul rates for the 2.4 GHz and 5 GHz bands.

Example

```
Device# ap name ap-name mesh backhaul rate dot11ax mcs <0-11> ss <1-8>
```

The AP uses the configured backhaul data rates for 2.4 GHz and 5 GHz bands.

```
Device> enable
Device# ap name ap-name mesh backhaul rate dot11ax mcs 5 ss 4
```

Configure data rate using mesh profile (CLI)

Use this procedure to configure the backhaul data rate for mesh profiles using command-line interface commands on your device.

Set specific data rates for 2.4 GHz and 5 GHz backhaul communication in mesh profiles to optimize wireless mesh network performance.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a mesh profile and enter mesh profile configuration mode.

Example

```
Device(config)# wireless profile mesh profile-name
```

3. Configure the backhaul transmission rate for the 2.4 GHz band and the 5 GHz band. For the 2.4 GHz band, set the 802.11ax spatial stream value from 1 to 4. For the 5 GHz band, set the spatial stream value from 1 to 8.

Example

```
Device(config-wireless-mesh-profile)# backhaul rate dot11 {24ghz | 5ghz}
dot11ax mcs <0-11> spatial-stream <1-8>
```

The mesh profile is updated with the configured data rate for the specified frequency band and spatial streams.

```
Device# configure terminal
Device(config)# wireless profile mesh profile-name
```

```
Device(config-wireless-mesh-profile)# backhaul rate dot11 5ghz dot11ax mcs 5
spatial-stream 6
```

Verify dot11ax rates on mesh backhaul

Lists commands to verify 802.11ax rates on mesh backhaul.

To verify the 802.11ax rates on mesh backhaul in the mesh profile, use this command:

```
Device# show wireless profile mesh detailed default-mesh-profile
Mesh Profile Name              : default-mesh-profile
-----------------------------------------------
Description                    : default mesh profile
.
.
Backhaul tx rate(802.11bg)     : 802.11ax mcs7 ss1
Backhaul tx rate(802.11a)      : 802.11ax mcs9 ss2
```

To verify the 802.11ax rates on mesh backhaul in the general configuration of an AP, use this command:

```
Device# show ap config general
Cisco AP Identifier             : 5c71.0d17.49e0
.
.
Backhaul slot id                : 1
Backhaul tx rate                : 802.11ax mcs7 ss1
```

Ethernet bridging

Explains Ethernet bridging across a mesh network, covering configuration tasks for bridging, multicast forwarding, and workgroup-bridge support to extend network connectivity.

An Ethernet bridge is a network device that

- enables you to secure activation of Ethernet ports on MAPs
- supports both tagged and untagged packets for flexible deployment, and
- allows segmenting application traffic with VLAN tagging between wireless and wired LANs.

For security, the Ethernet port on all MAPs is disabled by default. They can be enabled only through Ethernet bridging configuration on both the root and respective MAP.

Secondary Ethernet interfaces support both tagged and untagged packets.

In a point-to-point bridging, a Cisco Aironet 1500 Series MAP can extend remote networks by using the backhaul radio to bridge multiple segments of a switched network. This is fundamentally a wireless mesh network with one MAP and no WLAN clients.

In point-to-multipoint networks, client access can still be provided with Ethernet bridging enabled, although if bridging between buildings, MAP coverage from a high rooftop might not be suitable for client access. To use an Ethernet-bridged application, enable the bridging feature on the RAP and on all the MAPs in that sector.

In a mesh environment, with VLAN support for Ethernet bridging, the secondary Ethernet interfaces on MAPs are assigned a VLAN individually from the controller. Wired and wireless backhaul links operate as trunks with all VLANs allowed. Non-Ethernet bridged traffic, as well as untagged Ethernet bridged traffic travels along the mesh using the native VLAN of the APs in the mesh. It is similar for all the traffic to and from the wireless clients that the APs are servicing. The VLAN-tagged packets are tunneled through AWPP over wireless backhaul links.

> **Note**
>
> Ensure Ethernet bridging is enabled for every parent mesh AP along the data path to the controller.

Ethernet bridging should be enabled for these scenarios:

- Use mesh nodes as bridges.
- Connect Ethernet devices, such as a video camera on a MAP using its Ethernet port.

VLAN tagging for MAP Ethernet clients

Primary interfaces refer to mesh AP backhauls, and secondary interfaces refer to other AP interfaces.

Ethernet VLAN tagging segments application traffic within a mesh and forwards it to a wired LAN (access mode) or another wireless mesh network (trunk mode).

Configure Ethernet bridging (GUI)

Enable Ethernet bridging on a mesh profile via the configuration interface.

Enable Ethernet bridging on a mesh profile so that network traffic can be transparently passed between Ethernet segments.

Procedure

1. Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**.
2. Click **Add**.
3. In **General** tab, enter the **Name** of the mesh profile.
4. In the **Advanced** tab, check the **VLAN Transparent** check box to enable VLAN transparency.
5. In **Advanced** tab, check the **Ethernet Bridging** check box.
6. Click **Apply to Device**.

Ethernet bridging is now enabled for the selected mesh profile.

Configure Ethernet bridging (CLI)

Learn how to configure Ethernet bridging on access points using CLI commands to enable device connectivity.

Configure Ethernet bridging on mesh access points (APs) to allow Ethernet devices to connect through AP ports.

The Ethernet ports on MAPs are disabled by default. To enable them, configure Ethernet bridging on the Root AP and the relevant MAPs.

You can enable Ethernet bridging to:

- Use the mesh nodes as bridges.
- Connect Ethernet devices, such as a video camera, on a MAP using the MAP's Ethernet port.

Before you begin

- Ensure that you configure these commands under the mesh profile configuration for Ethernet bridging to be enabled:
  - **ethernet-bridging**: Enables the Ethernet Bridging feature on an AP.

- **no ethernet-vlan-transparent**: Makes the wireless mesh bridge VLAN aware. VLAN filtering is allowed with this AP command: **[no] mesh ethernet { 0 | 1 | 2 | 3 } mode trunk vlan allowed .**

  Note

  If you want all VLANs bridged (the bridge acts like a piece of wire), enable VLAN transparency to allow all VLANs to pass. To avoid unnecessary traffic flooding the network, filter VLANs on the wired side when using VLAN transparent mode.

- Configure the switch port that connects to the root AP as a trunk port so Ethernet bridging works.
- For Bridge mode APs, use the **ap name** *name-of-rap* **mesh vlan-trunking native** *vlan-id* command to configure a trunk VLAN on the corresponding RAP. You must configure this command to enable the Ethernet Bridging feature on the AP.
- For FlexConnect+Bridge APs, configure the native VLAN ID under the corresponding flex profile.

e

Note

To ensure that the MAPs apply the Ethernet VLAN configuration on the controller, configure the native VLAN on the RAP by running this command:

```
Device# ap name ap-name no mesh vlan-trunking
Device# ap name ap-name mesh vlan-trunking native 247
```

Alternatively, you can configure native VLAN on the RAP and then the MAP in this order:

```
Device# ap name ap-name no mesh vlan-trunking
Device# ap name ap-name mesh vlan-trunking native vlan_id
Device# ap name ap-name mesh ethernet 1 mode trunk vlan native native
Device# ap name ap-name mesh ethernet 0 mode trunk vlan allowed allowed
```

To verify the status of RAP and MAP, run this command:

```
Device# show mesh forwarding all
```

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

   Enter your password, if prompted.

2. Configure the Ethernet port of the AP and set the mode.

Example

```
Device# ap name ap-name mesh ethernet {0 | 1 | 2 | 3} mode access vlan-id
```

3. Set the native VLAN for the trunk port.

Example

```
Device# ap name ap-name mesh ethernet {0 | 1 | 2 | 3} mode trunk vlan native
 21
```

4. Configure the allowed VLANs for the trunk port.

Example

```
Device# ap name ap1 mesh ethernet  { 0 | 1 | 2 | 3 } mode trunk vlan allowed
 vlan-id
```

This command permits VLAN filtering on an Ethernet port of any Mesh or Root Access Point. It is active only when VLAN transparency is disabled in the mesh profile.

Ethernet bridging is enabled on the AP, allowing devices to communicate through configured VLANs.

```
Device> enable
Device# ap name ap1 mesh ethernet 1 mode access 21
Device# ap name ap1 mesh ethernet 1 mode trunk vlan native 21
Device# ap name ap1 mesh ethernet 1 mode trunk vlan allowed 21
```

Multicast over mesh Ethernet bridging network

This concept explains the mesh multicast modes available for bridging-enabled APs in mesh Ethernet networks and how each mode manages multicast and broadcast packet transmission.

A mesh multicast mode is a traffic management setting for bridging-enabled APs that

- determine how multicast and broadcast packets are forwarded across the mesh Ethernet network
- manage only non-CAPWAP multicast traffic, and
- help optimize bandwidth by reducing unnecessary multicast transmissions.

Mesh multicast modes

Mesh multicast modes determine how bridging-enabled APs such as MAP and RAP send multicast packets among Ethernet LANs within a mesh network. Mesh multicast modes manage only non-CAPWAP multicast traffic. CAPWAP multicast traffic is governed by a different mechanism.

Three different mesh multicast modes are available to manage multicast and broadcast packets on all MAPs. Enabling these modes ensures that unnecessary multicast transmissions within the mesh network are reduced and backhaul bandwidth is conserved.

The three mesh multicast modes are:

- Regular mode: In regular mode, bridging-enabled RAP and MAP multicast data across the entire mesh network and all its segments.

- In-only mode: When a MAP receives multicast packets from the Ethernet, it forwards them to the corresponding RAP's Ethernet network. No additional forwarding occurs. This setting ensures that the RAP does not send non-CAPWAP multicasts back to the MAP Ethernet networks within the mesh network (their point of origin). The system filters out MAP to MAP multicasts so these do not occur.
- In-out mode: The RAP and MAP multicast in different ways.
  - If a MAP receives multicast packets over Ethernet, it sends them to the RAP. The MAP does not send these packets to other MAPs over Ethernet; the system filters MAP-to-MAP packets from the multicast stream.
  - If a RAP receives multicast packets over Ethernet, it sends them to all the MAPs and their respective Ethernet networks. When in-out mode operates, partition your network to ensure a multicast sent by one RAP is not received by another RAP on the same Ethernet segment and then sent back into the network.

Configure multicast modes over mesh

Use this procedure to configure multicast operation modes on a mesh network.

Enable multicast forwarding behavior so that your wireless mesh network operates efficiently.

- If multicast packets are received at a MAP over Ethernet, the MAP sends them to the RAP. The MAP does not send them to other MAPs; it filters out MAP-to-MAP packets from the multicast.
- If multicast packets are received at a RAP over Ethernet, they are sent to all the MAPs and their respective Ethernet networks.
- The *in-out* mode is the default mode. To prevent a multicast sent by one RAP from being received by another RAP on the same Ethernet segment and then sent back into the network, properly partition your network when the *in-out* mode is in operation.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a mesh profile and enter mesh profile configuration mode.

   Example

   ```
   Device(config)# wireless profile mesh profile-name
   ```

3. Configure mesh multicast mode.

   Example

   ```
   Device(config-wireless-mesh-profile)# multicast {in-only | in-out | regular}
   ```

You have set the mesh profile to the multicast mode you want. This setting controls how multicast packets are routed in the network.

```
Device# configure terminal
Device(config)# wireless profile mesh mesh-multicast
Device(config-wireless-mesh-profile)# multicast regular
```

Workgroup bridge

This concept explains how workgroup bridges (WGBs) connect wired networks over a single wireless segment and their key operational attributes.

A workgroup bridge (WGB) is a wireless networking device that

- connects wired networks over a single wireless segment
- informs the corresponding mesh access point (MAP) of all wired clients on its segment through IAPP messages, and
- uses an additional MAC address header to route packets to and from wired clients.

Data packets for WGB clients contain an extra MAC address in the 802.11 header (four MACs, instead of the usual three), with the extra address being that of the WGB itself. The extra MAC in the header is the address of the workgroup bridge itself. This extra MAC address is used to route a packet to and from the corresponding clients.

You can configure APs as workgroup bridges. You need only one radio interface for controller connectivity. Use the Ethernet interface for wired client connectivity and another radio interface for wireless clients.

In Cisco Catalyst 9800 Series Wireless Controllers, WGB acts as a client association that allows wired clients behind the WGB to pass data over the mesh network. Wired clients with different VLANs behind a WGB are also supported.

DHCP and NAT functionality on root AP (RAP)

Describes integrated DHCP server and NAT functions provided by root APs, including operational processes, configuration steps, and verification commands for network service delivery.

DHCP and NAT functionality is a wireless network feature that

- assigns private IPv4 addresses to client devices using DHCP
- translates these private addresses to public addresses for network access with NAT, and
- enables root APs (RAPs) to provide seamless connectivity for clients in a mesh network.

The APs associated with a mesh network can play one of the two roles:

- Root Access Point (RAP) – An AP can be a root AP for multiple mesh networks.
- Mesh Access Point (MAP) – An AP can be a mesh AP for only one mesh network at a time.

> **Note**
>
> This feature is applicable for Cisco Aironet 1542 series outdoor APs.

How DHCP and NAT functionalities work on root AP - IPv4 scenario

Learn how root access points (RAPs) in mesh networks assign private IP addresses to your devices and help you securely access external networks using DHCP and NAT.

In wireless mesh networking, root APs (RAPs) can be enabled to provide both DHCP (Dynamic Host Configuration Protocol) and NAT (Network Address Translation) functionalities. This feature enables the controller to send a TLV to RAP when a new RAP joins the controller.

Summary

RAPs use integrated DHCP and NAT functionalities to assign private IPv4 addresses to wireless clients and provide secure, translated access to external networks. The process orchestrates communication between the network controller, RAP, and client device.

Workflow

The workflow consists of these steps:

1. The controller sends a TLV to configure the RAP with DHCP and NAT functionalities when a RAP joins the network.
2. A client device associates to the RAP's SSID.
3. The RAP assigns a private IPv4 address to the client using its DHCP functionality.
4. The RAP applies NAT to translate the client's private address, enabling secure access to external networks.

Result

Wireless clients that connect to RAPs receive private IPv4 addresses and can securely access external networks using DHCP and NAT services managed by the RAP.

Configure DHCP server on root AP (CLI)

Configure DHCP server on RAP.

Configure a DHCP server on the root AP.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile.

   Example

   ```
   Device(config)# ap profile ap-profile-name
   ```

3. Configure the DHCP server on the root AP.

Example

```
Device(config-ap-profile)# dhcp-server
```

4. Exit the configuration mode.

Example

```
Device(config-ap-profile)# end
```

The system saves the configuration, exits configuration mode, and returns to privileged **EXEC** mode.

The root AP is now configured as a DHCP server and assigns IP addresses to associated clients.

DHCP server for root AP configuration verification

Provides verification of DHCP server for RAP configuration.

To verify the DHCP server for root AP configuration, use this command:

```
Device# show ap config general
Cisco AP Name   : AP4C77.6DF2.D588
================================================
<SNIP>
Dhcp Server                                     : Enabled
```

Mesh convergence

Explains mesh convergence, including failure detection, alternate path selection, fast network recovery mechanisms, and configuration and verification of parent loss detection and gateway reachability.

A mesh convergence event is a network recovery mechanism that

- allows MAPs to re-establish connection with a controller after losing the backhaul link to the current parent
- enables the use of a maintained subset of channels for future scanning and parent identification, and
- supports multiple convergence methods to optimize reconnection time.

The table presents the supported convergence methods.

Table 169: Mesh convergence

Mesh convergence	Parent Loss Detection / Keepalive Timers
Standard	21 seconds / 3 seconds
Fast	7 seconds / 3 seconds
Very Fast	4 seconds / 2 seconds
Noise-tolerant-fast	21 seconds / 3 seconds

Noise-tolerant fast detection

Learn how noise-tolerant fast detections are used for parent loss detection in network environments, focusing on their mechanisms for evaluating connectivity and triggering recovery actions.

A noise-tolerant fast detection is a detection method that

- monitors the response to Adaptive Wireless Path Protocol (AWPP) neighbor requests at specified intervals
- identifies parent connectivity loss through missed responses, and
- initiates network recovery actions such as roaming or full scans when loss is detected.

Noise-tolerant fast detection occurs when there is no response to an AWPP neighbor request. In the standard method, the system evaluates the current parent every 2 1 seconds. Each neighbor receives a unicast request every three seconds, and the parent also receives a request. If the parent does not respond, the device either roams to an available neighbor on the same channel or performs a full scan to find a new parent.

Configure mesh convergence (CLI)

Configure mesh convergence using commands.

This section provides information about how to configure mesh convergence.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a mesh profile.

   Example

   ```
   Device(config)# wireless profile mesh profile-name
   ```

3. Configure mesh convergence method in a mesh profile.

   Example

   ```
   Device(config-wireless-mesh-profile)# convergence {fast | noise-tolerant-fast
    | standard | very-fast}
   ```

You have successfully configured mesh convergence for the specified mesh profile on the device.

```
Device# configure terminal
Device(config)# wireless profile mesh mesh1
Device(config-wireless-mesh-profile)# convergence fast
```

Mesh convergence verification

Provides commands to verify mesh convergence configuration and status.

Verify mesh convergence configuration and view subset channels for bridge groups using command-line interface commands.

This example shows the output from the **show wireless profile mesh detailed** command, which displays the mesh convergence method.

```
Device# show wireless profile mesh detailed default-mesh-profile

Mesh Profile Name             : default-mesh-profile
-------------------------------------------------
Description                 : default mesh profile
Convergence Method          : Fast
```

This example shows the output from **show wireless mesh convergence subset-channels** command, which displays the subset channels for the selected bridge group name.

```
Device# show wireless mesh convergence subset-channels

Bridge group name                Channel
-----------------------------------------
Default                          132
```

Mesh network recovery mechanisms

Learn the mechanisms that improve mesh network recovery by rapidly detecting uplink gateway reachability failures in mesh APs.

A mesh network recovery mechanism is a network self-healing feature that

- rapidly detects uplink gateway reachability failures in mesh APs
- automatically triggers alternate uplink selection to maintain connectivity, and
- uses ICMP pings to default gateways to verify uplink status after failover events or controller disconnections.

In all 802.11ac Wave 2 and 802.11ax mesh APs, detecting uplink gateway failures quickly speeds up mesh network recovery. Mesh APs use ICMP ping to the default gateway, either IPv4 or IPv6, to check uplink gateway reachability.

Reachability checks for mesh APs work

A mesh AP triggers the reachability check in two scenarios:

- After a new uplink is selected and before the mesh AP joins the controller

  After a new uplink is selected, the mesh AP has a 45-second window to reach the gateway (through static IP or DHCP) through the selected uplink. If the mesh AP does not reach the gateway within 45 seconds, the system blocks the current uplink and starts selecting a new uplink. If the AP joins the controller during this window, it stops the reachability check. The system does not perform gateway reachability checks during normal operations.

- As soon as the mesh AP times out its connection with the controller

  When the mesh AP times out its connection with the controller and fails to reach the gateway within 5 seconds, the system marks the current uplink as blocked and starts the uplink selection process.

Mesh daisy chaining

Describes Ethernet daisy chaining among mesh APs, presenting prerequisites, restrictions, configuration procedures, enabling on specific AP models, and verification for chained connectivity.

A mesh daisy chain is a wireless networking topology that

- connects mesh APs in series to relay data
- enables both serial backhaul communication and network extension for client access, and
- supports flexible deployment by allowing MAPs to connect in different modes and power configurations.

Mesh APs can daisy chain when operating as Mesh APs. In a daisy chain configuration, MAPs function as either serial backhaul links—using different channels for uplink and downlink to increase backhaul bandwidth—or as extensions of universal network access. Extending universal access allows a local mode or FlexConnect mode Mesh AP to connect to the Ethernet port of a MAP, expanding network reach for client devices.

Wired daisy-chained APs require specific cabling based on power source:

- If an AP uses DC power, connect an Ethernet cable from the LAN port of the Primary AP directly to the PoE-in port of the Subordinate AP.

Prerequisites for mesh Ethernet daisy chaining

This topic outlines the essential requirements that must be met before implementing mesh Ethernet daisy chaining.

Before you deploy mesh Ethernet daisy chaining, complete several configuration steps and verify specific settings on your APs. These prerequisites include designating AP roles, enabling relevant features, configuring VLAN support, and using the proper cabling to ensure optimal system performance and compatibility.

- Ensure that you have configured the AP role as root AP.
- Ensure that you have enabled Ethernet bridging and Strict Wired Uplink on the corresponding AP.
- Ensure that you have disabled VLAN transparency.
- To enable VLAN support on each root AP operating in bridge mode, use the **ap name** *name-of-rap* **mesh vlan-trunking [native]** *vlan-id* command. This command configures a trunk VLAN on the corresponding RAP.
- To enable VLAN support on each root AP, for Flex+Bridge APs, you must configure the native VLAN ID under the corresponding Flex profile.
- Ensure that you use 4-pair cables that support 1000 Mbps. This feature does not work properly with 2-pair cables that support 100 Mbps.

Restrictions for mesh Ethernet daisy chaining

This topic outlines the essential requirements you must meet before implementing mesh Ethernet daisy chaining on Cisco Industrial Wireless APs.

Review and meet these requirements before you configure mesh Ethernet daisy chaining:

- This feature is applicable to the Cisco Industrial Wireless 3702 AP and Cisco Catalyst 9124 Series APs.
- This feature is applicable to APs operating in Bridge mode and Flex+Bridge mode only.
- In Flex+Bridge mode, if local switching WLAN is enabled, the work group bridge (WGB) multiple VLAN is not supported.
- For Ethernet daisy chain topology, connect a power injector as the power supply for the AP. Do not connect the Cisco Industrial Wireless 3702 PoE out port to the PoE in port of another Cisco Industrial Wireless 3702 AP.

- The network convergence time increases as the number of APs in the chain increases.
- Any EWC-capable AP that is part of daisy chaining and has been assigned the RAP role must operate in CAPWAP mode (ap-type capwap).

Configure mesh Ethernet daisy chaining (CLI)

Configure mesh ethernet daisy chaining using commands.

This section provides information about how to configure the Mesh Ethernet Daisy Chaining feature on a mesh AP.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the AP profile.

   Example

   ```
   Device(config)# ap profile default-ap-profile
   ```

3. Configure persistent SSID broadcast and ensure strict wired uplink. RAP will not switch to wireless backhaul when you configure this command.

   Example

   ```
   Device(config-ap-profile)# ssid broadcast persistent
   ```

Mesh Ethernet daisy chaining is successfully enabled on the mesh AP. The SSID broadcast is persistent and the RAP uses a strict wired uplink.

Enable mesh Ethernet daisy chaining

Enable mesh ethernet daisy chaining.

This section describes how to enable the Mesh Ethernet Daisy Chaining feature on a Cisco IW 3702 AP.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a mesh profile.

Example

```
Device(config)# wireless profile mesh default-mesh-profile
```

3. Connect remote wired networks to each other.

Example

```
Device(config)# ethernet-bridging
```

4. Disable VLAN transparency to ensure that the bridge is VLAN aware.

Example

```
Device(config)# no ethernet-vlan-transparent
```

The Ethernet daisy chaining feature is enabled.

Mesh Ethernet daisy chaining verification

Provides commands to verify mesh ethernet daisy chaining configuration and status.

Use these commands to verify mesh ethernet daisy chaining configuration and check the Persistent SSID broadcast status for access points.

Use the **show AP config general** command to find out if a Persistent SSID is configured for an AP.

```
Device# show ap 3702-RAP config general

Persistent SSID Broadcast                  Enabled/Disabled
```

Use the **show wireless mesh persistent-SSID-broadcast summary** command to view the Persistent SSID broadcast status for all bridge root APs

```
Device# show wireless mesh persistent-ssid-broadcast summary

 AP Name     AP Model BVI MAC        BGN        AP Role              Persistent
 SSID state
-------     -------- -------        ---        -------
---------------------
3702-RAP     3702     5c71.0d07.db50 ap_name    Root AP               Enabled
1560-RAP    1562E    380e.4dbf.c6b0 ap_name    Root AP              Disabled
```

FlexConnect + bridge mode

Covers resilient mesh deployments using FlexConnect and bridge-mode APs, including configuration of flex profiles, site tags, mesh profiles, AP associations, and verification for enhanced redundancy.

A FlexConnect bridge mode is a wireless network configuration that

- enables FlexConnect capabilities on mesh (bridge mode) APs,
- allows mesh APs to inherit VLANs from the root AP, and

- supports VLAN trunking and native VLAN ID configuration per AP.

FlexConnect + bridge mode enables FlexConnect capabilities on mesh (bridge mode) APs. Mesh APs inherit VLANs from the root AP to which they are connected.

Any EWC-capable AP in FlexConnect mode connected to a MAP should be in CAPWAP mode (AP-type CAPWAP).

You can enable or disable VLAN trunking and configure a native VLAN ID on each AP for any of these modes:

- FlexConnect
- FlexConnect + bridge (FlexConnect + mesh)

Flex resilient with Flex and bridge modes

Explains Flex Resilient deployment with Flex and bridge mode APs, detailing configuration steps for Flex and mesh profiles, site tag and mesh associations, switch interface setup, AP attachment, and verification of resilient configurations.

A flex resilient system in flex and bridge modes is a network feature that

- operates within a configuration involving both flex and bridge mode APs
- allows the primary functionality of APs as mesh links between a root access point (RAP) and mesh access point (MAP), and
- ensures that the feature's operations are limited to flex and bridge mode APs.

Additional reference information

- Flex+Bridge Mode: A configuration mode combining the flexibility of standalone (local) management for APs with stable mesh network connectivity through root and mesh APs.
- Mesh Link Operation: Exists between RAP and MAP, allowing communication and bridging.
- Client Connectivity:
  - No new or disconnected clients can associate with a Mesh AP in flex+bridge mode.
  - Ongoing connections are maintained until a loss of the parent link for a child MAP.
- In a locally switching WLAN, the client traffic of a Flex+Bridge MAP is sent to the RAP's switchport.
- A child MAP that loses its parent connection cannot connect to a new parent without reacquiring a connection to the CAPWAP controller.
- The feature does not operate on non-Flex+Bridge mode APs.
- A new or disconnected wireless client cannot associate with a Mesh AP when in this mode.

Configure a flex profile (GUI)

Configure a flex profile using GUI.

Set up a flex profile to enhance network resilience and enable local authentication.

Use these steps to configure a flex profile using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Flex**.

2. Click a **Flex Profile Name**. The **Edit Flex Profile** dialog box appears.
3. Under the **General** tab, choose the **Flex Resilient** check box to enable the Flex Resilient feature.
4. Under the **VLAN** tab, choose the required VLANs.
5. (Optionally) Under the **Local Authentication** tab, choose the desired server group from the **Local Accounting RADIUS Server Group** drop-down list. Also, choose the **RADIUS** check box.
6. Click **Update & Apply to Device**.

The flex profile is successfully configured and updated in the system.

Configure a Flex profile (CLI)

Configure a Flex Profile using commands.

Configure a Flex Profile using commands on a network device to enable specific features and settings.

Use these steps to configure a flex profile using the CLI.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a flex profile and enters flex profile configuration mode using the **wireless profile flex** *flex-profile* command.

   Example

   ```
   Device(config)# wireless profile flex new-flex-profile
   ```

3. Enable ARP caching.

   Example

   ```
   Device(config-wireless-flex-profile)# arp-caching
   ```

4. Enable default parameters for the Flex profile using the **description** *description* command.

   Example

   ```
   Device(config-wireless-flex-profile)# description "new flex profile"
   ```

5. Configure native vlan-id information.

   Example

   ```
   Device(config-wireless-flex-profile)# native-vlan-id 2660
   ```

6. Enable the resilient feature.

Example

```
Device(config-wireless-flex-profile)# resilient
```

7. Configure VLAN name using the **vlan-name** *vlan_name* command.

   Example

   ```
   Device(config-wireless-flex-profile)# vlan-name VLAN2659
   ```

8. Configure the VLAN ID.

   Example

   ```
   Device(config-wireless-flex-profile)# vlan-id 2659
   ```

   The valid VLAN ID ranges from 1 to 4096.

9. Exit the configuration mode and return to the privileged EXEC mode.

   Example

   ```
   Device(config-wireless-flex-profile)# end
   ```

The Flex profile is configured on the network device with the specified settings and features.

Configure a site tag (CLI)

Configure a Site Tag using commands.

Configure a site tag on a network device to manage wireless networks efficiently and map APs to specific site tags. Use these steps to configure a site tag using CLI.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a site tag and enter site tag configuration mode using the **wireless tag site** *site-name* command.

   Example

   ```
   Device(config)# wireless tag site new-flex-site
   ```

3. Configure a flex profile using the **flex-profile** *flex-profile-name* command.

Example

```
Device(config-site-tag)# flex-profile new-flex-profile
```

4. Remove **Local site** configured from the site tag.

Example

```
Device(config-site-tag)# no local-site
```

5. Map a site tag to an AP using the **site-tag** *site-tag-name* command.

Example

```
Device(config-site-tag)# site-tag new-flex-site
```

6. Exit configuration mode and return to privileged EXEC mode.

Example

```
Device(config-site-tag)# end
```

The site tag is configured and mapped to the specified AP, facilitating effective management of wireless networks within the defined site tag.

Configure a mesh profile (CLI)

Configure a mesh profile using commands.

Configure a mesh profile on a network device to enable communication between mesh nodes, ensuring VLAN awareness and connectivity management.

Use these steps to configure a mesh profile using CLI commands.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a mesh profile and enter the mesh profile configuration mode using the **wireless profile mesh** *profile-name* command.

Example

```
Device(config)# wireless profile mesh Mesh_Profile
```

3. Disable VLAN transparency to ensure that the bridge is VLAN aware.

Example

```
Device(config-wireless-profile-mesh)# no ethernet-vlan-transparent
```

4. Exit the configuration mode and return to the privileged EXEC mode.

Example

```
Device(config-wireless-profile-mesh)# end
```

The mesh profile is configured, VLAN transparency is disabled to ensure VLAN awareness, and the device returns to privileged EXEC mode.

Associate the wireless mesh to an AP profile (CLI)

Associate Wireless Mesh to an AP Profile using commands.

Associate a wireless mesh to an AP profile to ensure proper configuration and management of AP.

Use these steps to associate the wireless mesh to an AP profile.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the AP profile and enter AP profile configuration mode using the **ap profile** *ap-profile-name* command.

Example

```
Device(config)# ap profile new-ap-join-profile
```

3. Configure the mesh profile in AP profile configuration mode using the **mesh-profile** *mesh-profile-name* command.

Example

```
Device(config-ap-profile)# mesh-profile Mesh_Profile
```

4. Configure the Secure Shell (SSH).

Example

```
Device(config-ap-profile)# ssh
```

5. Specify the AP management username and password for managing APs using the **mgmtuser username** *username* **password** {**0** | **8**} *password* command.

Example

```
Device(config-ap-profile)# mgmtuser username Cisco password 0 Cisco secret 0
 Cisco
```

- **0**: Specifies an UNENCRYPTED password.
- **8**: Specifies an AES encrypted password.

> **Note**
>
> While configuring a username, ensure that special characters are not used as they can lead to configuration errors.

6. Exit configuration mode and return to privileged EXEC mode.

Example

```
Device(config-ap-profile)# end
```

The AP profile is configured with the wireless mesh, allowing for proper management and operation of the AP.

Attach site tag to an access point (CLI)

Attach site tag to an AP using commands.

Attach a site tag for AP identification and configuration management.

Use these steps to attach a site tag to an AP:

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure APs and enters ap-tag configuration mode using the **ap** *mac-address* command.

Example

```
Device(config)# ap F866.F267.7DFB
```

3. Map a site tag to the AP using the**site-tag** *site-tag-name* command.

Example

```
Device(config-ap-tag)# site-tag new-flex-site
```

> **Note**
>
> Associating Site Tag causes the associated AP to reconnect.

4. Exit configuration mode and return to privileged EXEC mode.

Example

```
Device(config-ap-tag)# end
```

The AP is now associated with the specified site tag and will reconnect.

Configure switch interface for APs (CLI)

Configure Switch Interface for APs using commands.

Configure a switch interface for APs using CLI to ensure proper VLAN assignment and trunk settings.

Use these steps to configure switch interface for APs:

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enter the interface to be added to the VLAN.

Example

```
Device(config)# interface <int-id>
```

3. Assign the allowed VLAN ID to the port using the **switchport trunk native vlan** *vlan-id* command.

Example

```
Device(config-if)# switchport trunk native vlan 2660
```

4. Assign the allowed VLAN ID to the port using the **switchport trunk allowed vlan** *vlan-id* command.

Example

```
Device(config-if)# switchport trunk allowed vlan 2659,2660
```

5. Sets the trunking mode to trunk unconditionally.

Example

```
Device(config-if)# switchport mode trunk
```

Note

When the controller works as a host for spanning tree, ensure that you configure portfast trunk, using **spanning-tree portfast trunk** command, in the uplink switch to ensure faster convergence.

6. Exit configuration mode and return to privileged EXEC mode.

Example

```
Device(config-if)# end
```

The switch interface is successfully configured for AP connectivity, enabling effective VLAN management and ensuring faster network convergence for connected APs.

Flex resilient with flex and bridge mode AP configuration verification

Provides commands to verify Flex Resilient with Flex and Bridge Mode AP Configuration.

Use the verification commands to view details about AP mode and model, MAP mode details, RAP mode details, and the status of the Flex Profile–Resilient feature.

Use this command to view details about the AP mode and model.

```
Device# show ap name <ap-name> config general | inc AP Mode
AP Mode                                          : Flex+Bridge
AP Model                                         : AIR-CAP3702I-A-K9
```

Use this command to view the MAP mode details.

```
Device# show ap name MAP config general | inc AP Mode
AP Mode                                          : Flex+Bridge
AP Model                                         : AIR-CAP3702I-A-K9
```

Use this command to view the RAP mode details.

```
Device# show ap name RAP config general | inc AP Mode
AP Mode                                          : Flex+Bridge
AP Model                                         : AIR-AP2702I-A-K9
```

Use this command to check the status of the Flex Profile–Resilient feature.

```
Device# show  wireless profile flex detailed FLEX_TAG | inc resilient
Flex resilient                 : ENABLED
```

Mesh AP Roles and Topology

Describes AP roles, preferred-parent selection, leaf nodes, channel synchronization, and battery-state settings.

Select a preferred parent (GUI)

Configure the preferred parent MAC address for an access point in mesh mode to optimize wireless connectivity.

Direct a mesh-capable AP to use a designated parent for uplink within the wireless topology.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click the AP.
3. In the **Mesh** tab, enter the **Preferred Parent MAC**.
4. Click **Update & Apply to Device**.

The AP uses the specified preferred parent MAC address for uplink, optimizing connectivity in the mesh network.

Select a preferred parent (CLI)

Learn how to configure a mesh access point (AP) to use a specific preferred parent by entering CLI commands.

To configure a preferred parent for a MAP, complete these steps.

You can override the AWPP-defined parent selection and assign a specific preferred parent to a mesh AP using this mechanism.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

2. Configure mesh parameters for the AP and set the mesh-preferred parent MAC address.

   Example

   ```
   Device# ap name ap-name mesh parent preferred mac-address
   ```

> **Note**
>
> Ensure that you use the radio MAC address of the preferred parent.
>
> For Cisco Wave 1 APs, specify the MAC address of the actual mesh neighbor as the preferred parent. Use the base radio MAC address that ends with the letter "f". For example, if the base radio MAC address is 00:24:13:0f:92:00, then you must specify 00:24:13:0f:92:0f as the preferred parent.
>
> ```
> Device# ap name ap1 mesh parent preferred 00:24:13:0f:92:0f
> ```
>
> For Cisco Wave 2 APs, specify the MAC address for the preferred parent by adding "0x11" to the last two characters of the base radio MAC address. For example, if the base radio MAC address is 00:24:13:0f:92:00, then you must specify 00:24:13:0f:92:11 as the preferred parent.
>
> ```
> Device# ap name ap1 mesh parent preferred 00:24:13:0f:92:11
> ```

The mesh AP is configured to use the specified MAC address as its preferred parent.

```
Device> enable
Device# ap name ap1 mesh parent preferred 00:0d:ed:dd:25:8F
```

Change the role of an AP (GUI)

Change the role of an AP using GUI.

Change the operational role of an AP to support different network topologies.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click the **Access Point**.
3. In the **Mesh** tab, choose **Root** or **Mesh** from the **Role** drop-down list.
4. Click **Update & Apply to Device**.

The AP automatically restarts to apply the new role configuration.

Change the role of an AP (CLI)

Learn how to change the role of a Cisco access point (AP) between mesh and root modes using CLI commands to adapt your wireless network configuration.

Change your Cisco access point (AP) role between mesh AP (MAP) and root AP (RAP) using command-line interface (CLI) commands.

To change the AP from MAP to RAP or the reverse, follow this procedure. By default, your AP joins the controller in the mesh AP role.

Procedure

1. Enter privileged EXEC mode.

Example

```
Device> enable
```

2. Change the role of the Cisco bridge mode AP. The AP reboots after the role change.

Example

```
Device# ap name ap-name role {mesh-ap | root-ap}
```

The specified AP reboots and operates in its new role. After rebooting, it connects and functions in your wireless mesh network.

```
Device> enable
Device# ap name ap1 role root-ap
```

Mesh leaf node

Learn how mesh leaf nodes operate in a wireless mesh network and describes why lower performance Mesh Access Points (MAPs) are configured as leaf nodes.

A mesh leaf node is a type of mesh access point (MAP) that

- operates only as a child node within a wireless mesh network
- is typically assigned to MAPs with lower performance, and
- cannot serve as a parent MAP, ensuring wireless backhaul performance is not degraded.

Leaf nodes maintain stable network throughput by preventing low-performance devices from acting as wireless backhaul links for other MAPs.

Configure mesh leaf node (CLI)

Use this procedure to configure a mesh leaf node using CLI commands.

Set an AP as a mesh leaf node. Other mesh APs cannot select this AP as a parent MAP.

Procedure

---

1. Enter privileged EXEC mode.

Example

```
Device> enable
```

2. Configure the AP to operate only as a leaf node. Other MAPs cannot select this AP as a parent MAP.

Example

```
Device# ap name ap-name mesh block-child
```

Use the **no** form of this command to change it to a regular AP.

The AP operates solely as a mesh leaf node and cannot be selected as a parent MAP.

```
Device> enable
Device# ap name ap1 mesh block-child
```

Configure the mesh leaf node (GUI)

Configure the mesh leaf node using GUI.

Configure a mesh leaf node so that it blocks child connections to enforce network segmentation.

Procedure

---

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click the AP.
3. In the **Mesh** tab, check the **Block Child** check box.
4. Click **Update & Apply to Device**.

Your mesh leaf node now blocks child connections.

Configure subset channel synchronization

Learn how to configure subset channel synchronization for a mobility group to ensure faster mesh AP convergence.

Enable synchronization of mesh subset channels to ensure efficient convergence of RAPs and MAPs.

The controller sends all channels used by RAPs to MAPs for future seeking and convergence. The controller keeps a list of subset channels for each Bridge Group Name (BGN) and shares this list across all controllers in a mobility group.

The subset channel list includes channels where RAPs within a Bridge Group Name (BGN) operate. This list is distributed to all mesh access points (MAPs) within controllers and across controllers. Maintaining a subset channel list helps Mesh APs converge faster. You can select the convergence method in the mesh profile. The system sends the subset channel list to MAPs if the convergence method is non-standard.

Follow these steps to configure subset channel synchronization for a mobility group.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure subset channel synchronization for a mobility group.

Example

```
Device(config)# wireless mesh subset-channel-sync mac
```

The mobility group synchronizes subset channels, which enables mesh APs to converge faster.

```
Device# configure terminal
Device(config)# wireless mesh subset-channel-sync mac
```

Configure battery state for mesh AP (GUI)

Use this procedure to configure the battery state for a mesh access point using the GUI.

Configure the battery state on a mesh AP using the web interface.

Procedure

1. Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**.
2. Select a profile.
3. On the **General** tab, check the **Battery State for an AP** check box.
4. Click **Update & Apply to Device**.

After you configure the profile, the profile displays the battery state for the selected mesh AP.

Configure battery state for mesh AP (CLI)

Use this task to configure the battery backup feature for outdoor mesh APs to provide temporary power during outages and support external devices, using CLI commands.

Some outdoor APs come with the option of battery backup. The AP also includes a POE-out port that powers a video surveillance camera. The integrated battery provides temporary backup power during external power interruptions.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a mesh profile, and enter mesh profile configuration mode.

   Example

   ```
   Device(config)# wireless profile mesh profile-name
   ```

3. Configure the battery state for an AP.

Example

```
Device(config-wireless-mesh-profile)# battery-state
```

You have configured the battery state for the mesh AP, so backup power is now available when required.

```
Device# configure terminal
Device(config)# wireless profile mesh mesh-profile-name
Device(config-wireless-mesh-profile)# battery-state
```

Link test

Explains procedures for testing and assessing mesh backhaul link quality and performance, including GUI and CLI-based link test configurations.

A link test is a radio communication diagnostic tool that

- determines the radio link quality between two devices
- supports both ping and CCX test types for different directions, and
- allows both APs and clients to act as initiators and responders.

Two types of link-test packets are transmitted during a link test: request and response. Any radio that receives a link-test request packet fills in the appropriate text boxes. It then echoes the packet back to the sender with the response type set. The radio link quality in the client-to-access point direction can differ from that in the access point-to-client direction. This difference occurs because transmit power and receive sensitivity are distributed asymmetrically on both sides.

Types of link tests

The two types of link tests that you can perform are:

- **Ping link test**: With the ping link test, the controller tests link quality in the client-to-access point direction. The controller polls the RF parameters of the ping reply packets received by the AP to determine the client-to-access point link quality.
- **CCX link test**: With the CCX link test, the controller can also test the link quality in the access point-to-client direction. The controller sends link-test requests to the client. The client records RF parameters, such as received signal strength indicator (RSSI) and signal-to-noise ratio (SNR), in the response packet. Both requestor and responder roles exist on the AP and controller. You can initiate a link test from the AP, controller, or a CCX v4 or v5 client.

Use a link test on mesh backhaul (GUI)

Use this procedure in the graphical user interface to run a link test on a mesh backhaul connection. The task enables measurement of wireless link quality, packet loss, and connectivity between mesh access points to help ensure optimal network performance.

Network administrators can validate wireless mesh backhaul connectivity and signal strength with this task. Running a link test enables proactive identification of performance issues, ensures reliable mesh network operation, and helps troubleshoot weak connections.

Procedure

1. Choose **Monitoring** > **Wireless** > **AP Statistics** > **General**.

2. Select the Access Point.
3. Choose **Mesh** > **Neighbor** > **Linktest**.
4. Select the desired values from the **Date Rates**, **Packets to be sent (per second)**, **Packet Size (bytes)**, and **Test Duration (seconds)** drop-down lists.
5. Click **Start**.

You can see the detailed link test results for the selected mesh backhaul connection.

Use a link test on mesh backhaul

Use a link test on mesh backhaul using commands.

To initiate a link test between neighbor mesh APs, complete these steps.

> **Note**
>
> Use the **test mesh linktest mac-address** *neighbor-ap-mac* **rate** *data-rate* **fps** *frames-per-second* **frame-size** *frame-size* command to perform link test from an AP.

Procedure

1. Enters privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

2. Set the link test parameters.

   Example

   ```
   Device# ap name ap-name mesh linktest H.H.H data-rate packet-per-sec
   packet-size test-duration
   ```

   - *H.H.H*: Specify the destination AP MAC address.
   - *data-rate*: Specify the data rates in Mbps. The values are 1, 2, 5, 6, 9, 11, 12, 18, 24, 36, 48, 54, 108, or m0 through m15.
   - *packet-per-sec*: Specify the packets to be sent per second. The range is from 1 to 25,000.
   - *packet-size*: Specify the packet size in bytes. The range is from 1 to 1,500.
   - *test-duration*: Specify the test duration in seconds. The range is from 10 to 300.

The system displays test results, including throughput, loss, and latency metrics, to verify link quality between access points.

```
Device> enable
Device# ap name ap1 mesh linktest F866.F267.7DFB 24 234 1200 200
```

Air time fairness on mesh

Explains how ATF on mesh ensures fair airtime distribution and optimizes performance wirelessly in mesh networks.

Air time fairness (ATF) on mesh is a wireless quality of service feature that

- regulates downlink airtime distribution in mesh networks using token bucket mechanisms
- ensures fair access to radio airtime across mesh access points and their clients, and
- operates on both client access radios and backhaul radios to provide network-wide fairness.

ATF on mesh operation details

The Air Time Fairness (ATF) on Mesh feature is conceptually similar to the ATF feature for local access points (APs). ATF is a form of wireless quality of service (QoS) that regulates downlink airtime (as opposed to egress bandwidth). Before a frame is transmitted, the ATF budget for that SSID is checked to ensure that there is sufficient airtime budget to transmit the frame. Each SSID can be thought of as having a token bucket (1 token = 1 microsecond of airtime). If the token bucket contains enough airtime to transmit the frame, it is transmitted over air. Otherwise, the frame can either be dropped or deferred. Deferring a frame means that the frame is not admitted into the Access Category Queue (ACQ). Instead, it remains in the Client Priority Queue (CPQ) and transmitted at a later time when the corresponding token bucket contains a sufficient number of tokens (unless the CPQ reaches full capacity, at which point, the frame is dropped). The majority of the work involved in the context of ATF takes place on the APs. The wireless controller is used to configure the ATF on Mesh and display the results.

In a mesh architecture, the mesh APs (parent and child MAPs) in a mesh tree access the same channel on the backhaul radio for mesh connectivity between parent and child MAPs. The root AP is connected by wire to the controller, and MAPs are connected wirelessly to the controller. Hence, all the CAPWAP and Wi-Fi traffic are bridged to the controller through the wireless backhaul radio and through RAP. In terms of physical locations, normally, RAPs are placed at the roof top and MAPs in multiple hops are placed some distance apart from each other based on the mesh network segmentation guidelines. Hence, each MAP in a mesh tree can provide 100 percent of its own radio airtime downstream to its users though each MAP accessing the same medium. Compare this to a non-mesh scenario, where neighboring local-mode unified APs in the arena next to each other in different rooms, serving their respective clients on the same channel, and each AP providing 100% radio airtime downstream. ATF has no control over clients from two different neighboring APs accessing the same medium. Similarly, it is applicable for MAPs in a mesh tree.

For outdoor or indoor mesh APs, ATF must be supported on client access radios that serve regular clients similarly to how it is supported on ATF on non-mesh unified local mode APs to serve the clients. Additionally, it must also be supported on backhaul radios which bridge the traffic to/from the clients on client access radios to RAPs (one hop) or through MAPs to RAPs (multiple hops). It is a bit tricky to support ATF on the backhaul radios using the same SSID/Policy/Weight/Client fair-sharing model. Backhaul radios do not have SSIDs and it always bridge traffic through their hidden backhaul nodes. Therefore, on the backhaul radios in a RAP or a MAP, the radio airtime downstream is shared equally, based on the number of backhaul nodes. This approach provides fairness to users across a wireless mesh network, where clients associated to second-hop MAP can stall the clients associated to first-hop MAP where second-hop MAP is connected wireless to first-hop MAP through backhaul radio even though the Wi-Fi users in the MAPs are separated by a physical location. In a scenario where a backhaul radio has an option to serve normal clients through universal client access feature, ATF places the regular clients into a single node and groups them. It also enforces the airtime by equally sharing the radio airtime downstream, based on the number of nodes (backhaul nodes plus a single node for regular clients).

Configure ATF on mesh (GUI)

Configure ATF on mesh using GUI.

Configure ATF to optimize bandwidth allocation among clients on a mesh network.

Procedure

1. Choose **Configuration** > **Wireless** > **Airtime Fairness** > **Global Config**.
2. For **5 GHz Band** and **2.4 GHz Band**, enable the **Status** and the **Bridge Client Access** toggle button.
3. To select the **Mode**, click the **Monitor** or **Enforced** radio button.
4. Enable or disable the **Optimization** toggle button.
5. Enter **Airtime Allocation**.
6. Click **Apply to Device**.

The ATF settings are applied and mesh APs operate with the new airtime fairness parameters.

Configure ATF on mesh

Configure ATF on mesh using commands.

Ensure fair airtime allocation and optimize wireless performance for mesh environments.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an RF profile to enter RF profile configuration mode.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6gh} rf-profile rf-profile1
   ```

3. Configure airtime allocation weight percentage for mesh APs.

   Example

   ```
   Device(config-rf-profile)# airtime-fairness bridge-client-access
   airtime-allocation allocation-weight-percentage
   ```

You have now configured mesh APs for ATF. Balanced airtime allocation improves wireless network efficiency.

```
Device# configure terminal
Device(config)# ap dot11 24ghz rf-profile rfprof24_1
Device(config-rf-profile)# airtime-fairness bridge-client-access
airtime-allocation 10
```

Create an ATF policy for a MAP

Learn how to create an ATF (Airtime Fairness) policy for a MAP using command line interface commands.

Define and enable an Airtime Fairness (ATF) policy for a Mesh Access Point (MAP). This policy optimizes wireless network performance and ensures equitable airtime distribution among connected clients.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile to enter wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy  rr-xyz-policy-1
   ```

3. Enable ATF in the existing RF profile.

   Example

   ```
   Device(config-wireless-policy)# dot11 24ghz airtime-fairness atf-policy
   ```

Airtime Fairness is applied to the specified MAP when the policy is enabled.

```
Device# configure terminal
Device(config)# wireless profile policy rr-xyz-policy-1
Device(config-wireless-policy)# dot11 24ghz airtime-fairness atf-policy1
```

Create an ATF policy (GUI)

Create an ATF policy for a MAP using GUI.

Create a new Air Time Fairness (ATF) policy and apply it to a MAP (Managed Access Point) for optimal client traffic sharing.

Procedure

1. Choose **Configuration** > **Air Time Fairness** > **Profiles**.
2. On the **Profiles** window, click **Add**.
3. In the **Add ATF Policy** window, specify a name, ID, and weight for the ATF policy.

> **Note**
>
> A weighted ratio is used instead of percentages, allowing the total to exceed 100. The minimum weight that you can set is 5.

4. Use the slider to enable or disable the **Client Sharing** feature.
5. Click **Save & Apply to Device** to save your ATF configuration.
6. Optional: To delete a policy, check the check box next to the appropriate policy and click **Delete**.
7. Optional: To edit an existing ATF policy, select the check box next to the policy you want to edit.

   In the **Edit ATF Policy** window that is displayed, you can modify the weight and client sharing details for the policy.

You have created and applied the new ATF policy to the selected MAP.

Add an ATF to a policy profile (GUI)

Add an ATF to a policy profile using GUI.

Enable ATF on your selected policy profiles to assign fair radio resource allocation policies to wireless clients.

Procedure

---

1. Choose **Configuration > Tags & Profiles > Policy**.
2. Click the name of your policy profile.
3. Click the **Advanced** tab.
4. In the **Air Time Fairness Policies** section, choose the appropriate status for the 2.4 GHz Policy and 5 GHz Policy.
5. Click **Update & Apply to Device**.

Your selected policy profile is updated and now includes the ATF settings you chose.

Enable ATF mode in an RF profile (GUI)

Enable ATF mode in an RF profile using GUI.

Enable Airtime Fairness mode to distribute wireless bandwidth more equitably among clients in an RF profile.

Procedure

---

1. Choose **Configuration > Tags & Profiles > RF**.
2. Click the name of the corresponding RF profile.
3. In the **RF Profile** window, click the **Advanced** tab.
4. In the **ATF Configuration** section, choose the appropriate status:
   - **Status**: If you choose **Enabled** as the status, select the **Mode** as either **Monitor** or **Enforced**. You can also enable or disable optimization for this mode.
   - **Bridge Client Access**

- **Airtime Allocation**: Enter the allocation value. You can set the value only after you enable the **Bridge Client Access**.

5. Click **Update & Apply to Device**.

ATF mode is enabled for the selected RF profile, and the corresponding settings are applied to the device.

Verify ATF configuration on mesh

Verify ATF configuration on mesh.

You can verify Cisco ATF configurations on mesh APs using these commands.

Use this **show** command to display the ATF configuration summary of all the radios:

```
Device# show ap airtime-fairness summary

AP Name                           MAC Address       Slot Admin    Oper
Mode           Optimization
-------------------------------- ----------------- ---- -------- -----------
-------------- ------------
ap1/2                             6c:99:89:0c:73:a0 0    ENABLED  DOWN
Enforce-Policy Enabled
ap1/2                             6c:99:89:0c:73:a0 1    ENABLED  UP
Enforce-Policy Enabled
ap1/3                             6c:99:89:0c:73:a1 0    ENABLED  DOWN
Enforce-Policy Enabled
ap1/3                             6c:99:89:0c:73:a1 1    ENABLED  UP
Enforce-Policy Enabled
```

Use this **show** command to display the ATF configuration for a 2.4-GHz radio:

```
Device# show ap dot11 24ghz airtime-fairness

AP Name                           MAC Address       Slot Admin    Oper
Mode           Optimization
-------------------------------- -----------------   ---- -------- -----------
-------------- ------------
ap1/2                             6c:99:89:0c:73:a0 1    ENABLED  UP
Enforce-Policy Enabled
```

Use this **show** command to display the ATF WLAN statistics:

```
Device# show ap name ap1 dot11 24ghz airtime-fairness wlan 12 statistics

AP Name                           MAC Address       Slot Admin    Oper
Mode           Optimization
-------------------------------- ----------------- ---- -------- -----------
-------------- ------------
ap1/2                             6c:99:89:0c:73:a0 0    ENABLED  DOWN
Enforce-Policy Enabled
ap1/2                             6c:99:89:0c:73:a0 1    ENABLED  UP
Enforce-Policy Enabled      Network level
```

Use this **show** command to display the wireless mesh summary:

```
Device# show wireless profile mesh summary

Number of Profiles: 2
```

```
Profile-Name                        BGN         Security  Bh-access  Description
------------------------------------------------------------------------------------------
mesh1                                             EAP       DISABLED

default-mesh-profile                              EAP       DISABLED   default
mesh profile
```

Use this **show** command to display the mesh ATF client access information:

```
Device# show mesh atf client-access

AP Name              Client Access Allocation   Override   Current nodes
                      Default %    Current %
------------------   ---------    ---------    --------   -------------
RAP                      25           40        Enabled          4
RAP                      33           40        Enabled          3
```

Mesh call admission control

Describes call admission control (CAC) for mesh, outlining how real-time session admission is controlled to prevent bandwidth oversubscription and includes configuration steps.

A mesh call admission control is a quality of service mechanism that

- continuously monitors bandwidth available to mesh access points
- regulates voice call admissions to maintain acceptable call quality, and
- rejects calls when bandwidth or resource limits are reached.

Call Admission Control (CAC) enables a mesh AP to maintain controlled quality of service (QoS) on the controller. This management helps maintain voice quality on the mesh network. Bandwidth-based, or static, CAC enables the client to specify how much bandwidth or shared medium time is required to accept a new call. Each AP determines whether it can accommodate a call by checking the available bandwidth and comparing it to the bandwidth required for the call. If there is not enough bandwidth available to maintain the maximum allowed number of calls with acceptable quality, the mesh AP rejects the call.

Mesh CAC is not supported in these scenarios.

- APs in a Mesh tree assigned with different site tags.
- APs in a Mesh tree assigned with the default site tag.

Configure mesh CAC (CLI)

Configure mesh CAC using commands.

Enable mesh CAC to manage call admission and ensure quality over wireless mesh links.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enable mesh CAC mode.

Example

```
Device(config)# wireless mesh cac
```

Mesh CAC is activated. Only the supported number of concurrent calls can occur over mesh links, which preserves call quality.

Fast teardown for a mesh deployment

Explains the fast teardown procedure for mesh deployments, detailing how failed mesh relationships are swiftly removed, with steps for configuration and verification to accelerate recovery and topology rebuilding.

A fast teardown is a mesh deployment feature that

- enables rapid detection of root AP uplink failures
- helps restore or reconfigure network service when the uplink is lost, and
- applies to mesh deployments with unreliable uplinks, such as wireless microwave links.

Fast teardown for mesh APs is not supported on Cisco Industrial Wireless (IW) 3702 Access Points.

Configure fast teardown for a mesh AP profile (GUI)

Learn how to enable fast teardown for a mesh AP profile to accelerate recovery from uplink failures in mesh networks.

Use this procedure to configure mesh AP profiles for rapid detection of uplink failures and improve mesh network resiliency.

Procedure

1. Choose **Configuration** > **Wireless** > **Mesh** > **Profiles**.
2. Click **Add**.
3. In the **Add Mesh Profile** window, click **Advanced**.
4. Select a security mode, an authentication method, and an authorization method.
5. Enable **Ethernet bridging** if it is required.
6. Enter the bridge group name. Then enable Strict Match BGN.
7. Select a band backhaul transmission rate for your radio.
8. Perform these actions in the **Fast Roaming** section:
   - Check the **Fast Teardown** check box to detect the root AP uplink failure faster in a mesh deployment and to address fast teardown of the mesh network when an uplink failure occurs.
   - In the **Number of Retries** field, enter the number of retries allowed until gateway is considered unreachable. The valid range is from 1 to 10.
   - In the **Interval value** field, enter the retry value. The valid range is from 1 to 10 seconds.

- In the **Latency Threshold** field, enter the threshold for a round-trip latency between the AP and the controller. The valid range is from 1 and 500 milliseconds.
- In the **Latency Exceeded Threshold** field, enter the latency interval in which at least one ping must succeed in less than the specified time. The valid range is from 1 to 30 seconds.
- In the **Uplink Recovery Interval** field, enter the time during which root AP uplink must be stable in order to accept the child connections. The valid range is from 1 and 3600 seconds.

9. Click **Apply to Device**.

Your mesh AP profile now supports fast teardown. This configuration enables the system to detect uplink failures more quickly, helping your devices recover rapidly.

Configure fast teardown for a mesh AP profile (CLI)

Use this task to configure the fast teardown feature for a mesh AP profile using CLI commands to improve recovery times in mesh networks.

Enable and configure fast teardown for a mesh AP profile to reduce network restoration time and increase network reliability.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a mesh profile and enter the mesh profile configuration mode.

   Example

   ```
   Device(config)# wireless profile mesh profile-name
   ```

3. Enable the fast teardown of mesh network and configure the feature's parameter.

   Example

   ```
   Device(config-wireless-mesh-profile)# fast-teardown
   ```

4. Enable the fast teardown feature.

   Example

   ```
   Device(config-wireless-mesh-profile-fast-teardown)# enabled
   ```

5. Optional: Configure the retry interval.

   Example

   ```
   Device(config-wireless-mesh-profile-fast-teardown)# interval duration
   ```

The valid range is from 1 to 10 seconds.

6. Optional: Set the latency interval so that at least one ping succeeds within your chosen threshold time.

   Example

   ```
   Device(config-wireless-mesh-profile-fast-teardown)# latency-exceeded-threshold
    duration
   ```

   The valid range is from 1 to 30 seconds.

7. Optional: Specify the latency threshold.

   Example

   ```
   Device(config-wireless-mesh-profile-fast-teardown)# latency-threshold
   threshold-range
   ```

   The valid range is from 1 to 500 milliseconds.

8. Optional: Specify the number of retries until the gateway is considered unreachable.

   Example

   ```
   Device(config-wireless-mesh-profile-fast-teardown)# retries retry-limit
   ```

   The valid range is from 1 to 10.

9. Optional: Specify the time during which root access point uplink has to be stable to accept child connections.

   Example

   ```
   Device(config-wireless-mesh-profile-fast-teardown)# uplink-recovery-intervals
    recovery interval
   ```

   The valid range is from 1 to 3600 seconds.

You have configured the mesh AP profile with fast teardown enabled and applied custom parameters. This configuration improves your network's responsiveness to disruptions.

```
Device# configure terminal
Device(config)# wireless profile mesh mesh1
Device(config-wireless-mesh-profile)# fast-teardown
Device(config-wireless-mesh-profile-fast-teardown)# enabled
Device(config-wireless-mesh-profile-fast-teardown)# interval 5
Device(config-wireless-mesh-profile-fast-teardown)# latency-exceeded-threshold
 20
Device(config-wireless-mesh-profile-fast-teardown)# latency-threshold 20
Device(config-wireless-mesh-profile-fast-teardown)# retries 1
Device(config-wireless-mesh-profile-fast-teardown)# uplink-recovery-intervals
 1
```

Fast teardown verification with default mesh profile

Provides verification of fast teardown with default mesh profile.

To verify the fast teardown with the default mesh profile, use this command:

```
Device# show wireless profile mesh detailed default-mesh-profile
Mesh Profile Name            default-mesh-profile
------------------------------------------------
Fast Teardown                          : ENABLED
Number of Retries                      : 4
Interval in sec                        : 1
Latency Threshold in msec              : 10
Latency Exceeded Threshold in sec      : 8
Uplink Recovery Interval in sec        : 60
```

194 Redundant Root Access Point (RAP) Ethernet Daisy Chaining

Topics:

- Redundant RAP Ethernet daisy chaining

Redundant RAP Ethernet daisy chaining

Explains how redundant RAP Ethernet daisy chains allow RAPs to be connected using wired Ethernet to avoid latency in backhaul link failure recovery.

A redundant RAP Ethernet daisy chain is a network topology feature that

- connects Root APs (RAPs) in a sequence using wired Ethernet links
- provides redundant paths with two switches acting as designated ports for failover, and
- minimizes latency in backhaul link failure recovery by reversing data flow through a new STP root.

This feature proposes a redundancy in the daisy chain, wherein two switches act as a redundant Designated Port, each connected to either end of the daisy chain. In case of a link failure, the link direction is reversed using a new STP root.

A redundant RAP Ethernet daisy chain shares capabilities with the mesh daisy chain feature. Both topologies provide redundancy and improve network resiliency. In the redundant RAP Ethernet daisy chain topology, each AP encapsulates a packet with a CAPWAP header and forwards it from its wireless client to the controller. The AP bridges the packet from its secondary Ethernet interface to its primary Ethernet interface, including CAPWAP packets sent from other APs' wireless clients. Both 2.4 GHz and 5 GHz radios are available for client access.

Redundant RAP Ethernet daisy chain is supported on the IW6300 AP model.

The daisy chain strict RAP configuration is applicable to Cisco IOS APs only.

If a CAPWAP loss occurs on the first RAP connected to the switch in an Ethernet daisy chain topology, the entire chain loses its uplink, causing a long recovery time. When the RAP Ethernet daisy chain is enabled, the CAPWAP data keepalive interval increases threefold.

> **Note**
>
> Only the wired uplink configuration is valid when an AP is configured as a Bridge or Flex Bridge mode Root AP.

Prerequisites for redundant RAP Ethernet daisy chaining support

This topic outlines the requirements you must meet before enabling redundant RAP Ethernet daisy chaining support.

To successfully configure redundant RAP Ethernet daisy chaining, ensure that you:

- Enable Ethernet bridging.
- Enable the strict-wired-uplink feature.

Configure redundant RAP Ethernet daisy chaining support (CLI)

Use this procedure to configure redundant RAP Ethernet daisy chaining using CLI commands.

Enable redundant RAP Ethernet daisy chaining on a mesh profile to provide link redundancy in wireless mesh networks.

Use this configuration to improve network robustness by allowing redundant paths in the RAP Ethernet daisy chain topology.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a mesh profile and enter mesh profile configuration mode.

Example

```
Device(config)# wireless profile mesh profile-name
```

3. Configure daisy chain STP redundancy.

Example

```
Device(config-wireless-mesh-profile)# daisychain-stp-redundancy
```

Redundant RAP Ethernet daisy chaining is enabled for the specified mesh profile.

```
Device# configure terminal
Device(config)# wireless profile mesh default-mesh-profile
Device(config-wireless-mesh-profile)# daisychain-stp-redundancy
```

Verify daisy chain redundancy

Verify daisy chain redundancy using commands.

To verify the ethernet daisy chain summary, use this command:

```
Device# show wireless mesh ethernet daisy-chain summary

AP Name      BVI MAC          BGN     Backhaul      Ethernet          STP Red
-------------------------------------------------------------------------------
RAP4        683b.78bf.15f0    IOT     Ethernet0     Up  Up  Dn  Dn   Enabled
RAP3        683b.78bf.1634    IOT     Ethernet0     Up  Up  Dn  Dn   Enabled
RAP1        6c8b.d383.b4d4    IOT     Ethernet0     Up  Up  Dn  Dn   Enabled
RAP2        6c8b.d383.b4e8    IOT     Ethernet0     Up  Up  Up  Dn   Enabled
```

To verify the ethernet daisy chain Bridge Group Name (BGN) details, use this command:

```
Device# show wireless mesh ethernet daisy-chain bgn <IOT>

AP Name   BVI MAC          BGN    Backhaul    Ethernet          STP Red
-------------------------------------------------------------------------------
RAP4     683b.78bf.15f0    IOT    Ethernet0   Up  Up  Dn  Dn   Enabled
RAP3     683b.78bf.1634    IOT    Ethernet0   Up  Up  Dn  Dn   Enabled
RAP1     6c8b.d383.b4d4    IOT    Ethernet0   Up  Up  Dn  Dn   Enabled
RAP2     6c8b.d383.b4e8    IOT    Ethernet0   Up  Up  Up  Dn   Enabled
```

To verify the mesh profile, use this command:

```
Device# show wireless profile mesh detailed default-mesh-profile

Mesh Profile Name : default-mesh-profile
-----------------------------------------------
Description : default mesh profile
Bridge Group Name : IOT
```

```
Strict match BGN : ENABLED
Amsdu : ENABLED
Background Scan : ENABLED
Channel Change Notification : ENABLED
Backhaul client access : ENABLED
Ethernet Bridging : ENABLED
Ethernet Vlan Transparent : DISABLED
Daisy Chain STP Redundancy : ENABLED
Full Sector DFS : ENABLED
IDS : ENABLED
Multicast Mode : In-Out
Range in feet : 12000
Security Mode : EAP
Convergence Method : Standard
LSC only Authentication : DISABLED
Battery State : ENABLED
Authorization Method : eap_methods
Authentication Method : eap_methods
Backhaul tx rate(802.11bg) : auto
Backhaul tx rate(802.11a) : auto
===============
```

Part XVII

URWB

Topics:

- URWB

195 URWB

Topics:

- Ultra-Reliable Wireless Backhaul Background
- Configure URWB using the controller
- URWB telemetry protocol
- URWB recovery
- Dynamically configure URWB parameters
- Controller-directed URWB traffic
- Support for 6 GHz Low Power Indoor mode
- NAT/PAT for URWB
- Digital I/O ports
- IOx frameworks
- URWB topology awareness
- Ethernet port authentication

Ultra-Reliable Wireless Backhaul Background

Ultra-Reliable Wireless Backhaul (URWB) is a wireless technology that delivers highly reliable, low-latency network connectivity for critical applications and devices, using IEEE 802.11 standards in the 5 GHz band.

Ultra-Reliable Wireless Backhaul (URWB) is a wireless technology developed for mission and business critical applications that demand high reliability, low latency, and robust performance. It operates in the 5 GHz frequency band and is based on the widely adopted IEEE 802.11 standards. Leveraging the 802.11 framework simplifies implementation for Information Technology (IT) and Operational Technology (OT) teams. As a result, URWB enables greater deployment flexibility, reduced infrastructure costs, and accelerated installation when connecting fixed and mobile devices.

URWB is designed for wireless connectivity needs that are an integral part of your network infrastructure, rather than connecting end users like Wi-Fi technology is commonly used for.

URWB provides scalability without compromising reliability, delivering low latency and high availability. These characteristics are essential for mobile device use cases where stable connections and minimal latency are critical to ensuring safety and maintaining operational efficiency.

URWB supports two distinct connectivity models:

- Mobility: Provides seamless and uninterrupted connectivity to moving devices. Commonly used for Autonomous Guided Vehicles (AGV), autonomous mobile robots (AMR), trains, and other mobile devices.
- Fixed infrastructure: Establishes connectivity between radios attached to stationary structures such as poles, towers, or buildings. This architecture is frequently used for wireless backhaul and physical surveillance.

Mobility roles

1. Base (Infrastructure – AP role)

   Behavior: Serves as a fixed access point that mobile clients connect to.

2. Base-Relay (Infrastructure – Relay role)

   Behavior: A stationary infrastructure node that extends wireless coverage by relaying traffic between bases and clients.

3. Client (Mobile/Vehicle)

   Behavior: A moving node (e.g., mounted on a vehicle/crane/AGV) that actively scans for, connects to, and roams between infrastructure nodes.

Topologies

URWB supports multiple network configurations, including:

- Fixed Mesh: Multiple nodes are interconnected, allowing data to be routed dynamically through the most efficient path. Ideal for large industrial sites or campuses where redundancy and self-healing capabilities are required.
- Point-to-Multipoint (PTMP): A single central node connects to multiple remote nodes. Common in scenarios like wireless backhaul for several field devices or edge networks connecting to a central hub.
- Mobility: Supports connectivity to moving assets such as vehicles or robots. Essential for use cases requiring continuous, low-latency communication while in motion.

Benefits

- ultra-low latency,
- seamless handoffs, even at high speed (220mph - 350Km/h),
- high-throughput,

- high-availability with ultra-fast failover, and
- supports both fixed and mobile devices.

Limitations and restrictions

The following limitations exist for this release:

- Compatibility- URWB on Cisco Wireless project supports select access points. For complete information on AP devices use case and country support, see the guide "URWB on Cisco Wireless Use Case and Country".
- Mobility architectures extending over Layer 3 networks are not supported. Mobility deployments must be deployed with all Base access points in the same network segment.
- 6 GHz Standard Power is not supported on URWB radios.
- URWB supports up to two radio slots per access point.
- 2.4 GHz radios are not supported.
- SD Access (Wireless Fabric) is not supported.
- URWB supports one wired network port on each access point. LAN ports, on applicable access points, are not supported.
- Throughput on URWB is subject to processing limitations, which varies per access point platform.

Configure URWB using the controller

Configures URWB network settings and parameters using the wireless controller interface. This process enables centralized management and deployment of ultra-reliable wireless backhaul profiles for optimized connectivity in mission-critical environments.

Create URWB network profile (GUI)

Describes how to create URWB network profile for APs in the GUI.

A URWB Network Profile is a configuration framework designed to enable the setup of an Ultra-Reliable Wireless Backhaul (URWB) network for APs. This profile allows APs to form a wireless mesh network with enhanced reliability, performance, and security.

Complete these steps to configure URWB network profile:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **URWB Network Profile**.
2. Click **Add**.

   The **Add URWB Network Profile** page is displayed.

3. In the **General** tab, complete the following:
   a) To edit the profile, enable the **URWB Status** check box.
   b) Enter the **URWB Network Profile Name**.
   c) Enter the description of the profile name.
   d) Check the **Strong Network Key** check box, to configure a strong password evaluation for the network key. If you enable strong network key, it means that the network key must contain atleast one uppercase, lowercase alphabets, and digits.

e) Select either **Unencrypted** or **AES** from the **Network Key Type** drop-down list. You can use only **Unencrypted** for configuration.
f) Enter the **Network-Key**.

The network key must be between 8 and 63 characters long. Only uppercase alphabets, lowercase alphabets, digits, or special characters are allowed in the network key.

4. Click the **Mobility** tab and select one of the following **Mobility Roles**:
   - Base
   - Base Relay
   - Client
5. To edit the MPO attributes, click the **MPO** tab.

> **Note**
>
> To edit the MPO attributes, select an option other than **Disabled** in the **Packet Duplication Mode**. MPO is not mandatory to make the system operate.

a) In the **Class of Service** field, enter a value from 0 to 6.

This value indicates the number of packets that need to be duplicated. The default value is 6.

b) In the **Number of Paths** field, enter a value from 1 to 4.

This value indicates the number of independent paths to be established for traffic duplication. The default value is 1.

c) In the **Minimum SNR** field, enter a value from 0 to 96.

This value indicates the minimum SNR for a wireless link to be considered for building a redundant path. The default value is 20.

d) To enable telemetry, check the **Enable Telemetry** check box.

6. Complete the following details in the **Advanced** tab:

> **Note**
>
> You can leave the advanced parameters at their default settings; you do not need to configure them.

a) In the **Ethernet Protocols** area, select the **Allow Ether Types** from the drop-down list.

Select **List** from the **Allow Ether Types** drop-down list to allow only custom Ether values.

b) Check the **Legacy Ethernet** check box to allow Ethernet I frames fowarding.
c) In the **Unicast** area, check the **Unicast Flood** check box to enable unicast flood and enter unicast flood limitation.
d) Check the **Unicast Flood Limitation** check box to flood one packet every 5 seconds.
e) Select the **Multicast** option from the drop-down list.

7. Select the **Multicast** option from the drop-down list.

8. Check the **High Availability Enabled** check box to enable the redundancy support for devices in the same class (coordinator, vehicle devices, and point-to-multi-point base).
9. Click **Apply to Device**.

The URWB network profile is created.

Configure URWB network profile (CLI)

Describes the CLI steps to create a URWB network profile

Configure the URWB network key in the URWB pofile. The URWB profile is a wireless profile type that contains the URWB configuration parameters. All the APs with the same site-tag are configured with the URWB configurations defined in the URWB profile.

Complete the following steps to create a URWB network profile and configure the URWB network key:

Procedure

---

1. Enter configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter a profile name to create a URWB profile.

   Example

   ```
   Device(config)# wireless profile urwb urwb-profile-name
   ```

   This command creates a URWB profile and enters into the wireless URWB profile configuration mode.

3. Optional: Configure a strong password evaluation for the network key.

   Example

   ```
   Device(config-wireless-urwb-profile)# strong-network-key
   ```

   If you enable strong network key, it means that the network key must contain at least one uppercase, lowercase alphabets, and digits.

4. Configure a network key for secure communication between the URWB nodes.

   Example

   ```
   Device(config-wireless-urwb-profile)# network-key key {0 | 8}<network-key>
   ```

   Here, **0** indicates that you must enter an unencrypted network key; and **8** indicates an AES encrypted network key.

> **Note**
>
> You can only use type 0 for configuration. If you configure the encryption command (**password encryption aes**) in the configuration mode, the output of the **show running-config** command displays the details of the encrypted network key (8 xxxx).
>
> For more information, see Setting a Private Configuration Key for Password Encryption.

> **Note**
>
> The network key supports all the printable ASCII characters except the following special characters: ~ | ; $ < > & ^ ` ? \ and ".

5. Enable the URWB profile in the wireless URWB profile configuration mode.

   Example

   ```
   Device(config-wireless-urwb-profile)# no shutdown
   ```

   To disable the URWB profile, use the **shutdown** command.

6. Exit the sub-mode.

   Example

   ```
   Device(config-wireless-urwb-profile)# exit
   ```

The URWB network profile is created and the network key is configured in the wireless profile.

Example

```
Device# configure terminal
Device(config)# wireless profile urwb urwb-profile-name
Device(config-wireless-urwb-profile)# strong-network-key
Device(config-wireless-urwb-profile)# network-key network-key-type 0
<network-key>
Device(config-wireless-urwb-profile)# no shutdown
Device(config-wireless-urwb-profile)# exit
```

Configure URWB mobility backhaul check (CLI)

Complete the following steps to configure mobility role and backhaul check:

Procedure

---

1. Enter configuration mode.

Example

```
Device# configure terminal
```

2. Enter a profile name to create a URWB profile.

   Example

   ```
   Device(config)# wireless profile urwb urwb-profile-name
   ```

   This command creates a URWB profile and enters into the wireless URWB profile configuration mode.

3. Configure URWB mobility role.

   Example

   ```
   Device(config-wireless-urwb-profile)# mobility role {base| base-relay | client}
   ```

4. Configure URWB mobility backhaul-check when coordinator is unreachable

   Example

   ```
   Device(config-wireless-urwb-profile)# [no] mobility backhaul-check
   coordinator-down {handoff-inhibition | relay-switch}
   ```

   This command configures the handoff and handoff-inhibition when coordinator is unreachable. The relay-switch configuration switches to Infrastructure Wireless Relay mode if the coordinator is unreachable.

   Use the **no** form of this command to disable URWB mobility backhaul-check with coordinator-down

5. Configure URWB mobility backhaul-check when Ethernet ports are down.

   Example

   ```
   Device(config-wireless-urwb-profile)# [no] mobility backhaul-check eth-down
   {handoff-inhibition|relay-switch}
   ```

   This command configures the handoff and handoff-inhibition when when all Ethernet ports are disconnected. The relay-switch configuration switches to Infrastructure Wireless Relay mode if all Ethernet ports are down.

   Use the **no** form of this command to disable URWB mobility backhaul-check when Ethernet ports are down.

The URWB mobility role and backhaul check is configured.

Example

```
Device# configure terminal
Device(config)# wireless profile urwb urwb-profile-name
Device(config-wireless-urwb-profile)# mobility role {base| base-relay | client}
Device(config-wireless-urwb-profile)# [no] mobility backhaul-check
coordinator-down {handoff-inhibition | relay-switch}
Device(config-wireless-urwb-profile)# [no] mobility backhaul-check eth-down
{handoff-inhibition|relay-switch}
```

URWB Backhaul Check

The Backhaul Check feature monitors the connectivity status of infrastructure (base/base_relay) units' wired Ethernet ports and/or mesh-end connectivity. When backhaul connectivity is lost, the feature prevents mobile units from handing off to that infrastructure unit. This process ensures that traffic is not routed to a 'dead end' in the network.

Purpose

In a Fluidity/URWB network, infrastructure units (track-side radios) provide wireless coverage to mobile units (vehicle radios). These infrastructure units typically connect to the wired network via Ethernet. If an infrastructure unit loses its wired backhaul connection, it becomes a 'black hole.' Mobile units could connect to it wirelessly, but their traffic would have no path to the network.

Standard (Ethernet Port) Backhaul Check vs. Mesh-End Backhaul Check

This feature supports two operational modes: Standard Backhaul Check and Mesh-End Backhaul Check.

Standard Backhaul Check

What it monitors: The physical Ethernet port status on the infrastructure unit itself.

Use case: The infrastructure unit has a direct wired Ethernet connection to the backbone network. If that cable or port fails, the unit has no backhaul.

Mesh-End Backhaul Check

What it monitors: The logical reachability to the mesh-end (coordinator) via the wired AutoTAP probing mechanism – essentially checking if a pseudowire path to the coordinator still exists.

Use case: The infrastructure unit does NOT have a direct Ethernet connection to the backbone. Instead it reaches the network through a chain of other infrastructure units (e.g., a base-relay daisy chain). Even if its own Ethernet port is operational or does not exist, the path to the coordinator or mesh-end may be broken further upstream.

Configure URWB mobility MPO parameters (CLI)

Complete the following steps to configure URWB mobility MPO parameters.

Procedure

1. Enter configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter a profile name to create a URWB profile.

   Example

   ```
   Device(config)# wireless profile urwb urwb-profile-name
   ```

   This command creates a URWB profile and enters into the wireless URWB profile configuration mode.

3. Enable URWB mobility multipath operation (MPO).

   Example

   ```
   Device(config-wireless-urwb-profile)# mobility mpo enable {rx-only | tx-rx}
   ```

   This command enables MPO Receive Only or MPO Bidirectional.

4. Configure URWB mobility MPO traffic CoS.

Example

```
Device(config-wireless-urwb-profile)# mobility mpo cos cos-value
```

Here, the CoS value is between 0 to 7.

5. Configure URWB mobility MPO maximum paths (maximum number of MPO links).

Example

```
Device(config-wireless-urwb-profile)# mobility mpo path number of links
```

Here, the number of links are between 1 to 4.

6. Configure URWB mobility MPO RSSI.

Example

```
Device(config-wireless-urwb-profile)# mobility mpo rssi rssi-value
```

This command configures minimum RSSI to establish MPO redundant links. Here, the RSSI value is between 0 to 96.

7. Configure URWB mobility MPO telemetry.

Example

```
Device(config-wireless-urwb-profile)# [no] mobility mpo telemetry
```

The URWB mobility MPO parameters are configured.

Example

```
Device# configure terminal
Device(config)# wireless profile urwb urwb-profile-name
Device(config-wireless-urwb-profile)# [no] mobility mpo enable rx-only
Device(config-wireless-urwb-profile)# mobility mpo path 4
Device(config-wireless-urwb-profile)# mobility mpo rssi 4
Device(config-wireless-urwb-profile)# [no] mobility mpo telemetry
```

Configure URWB mobility scan parameters (CLI)

Before you begin

The mobility scan parameters are strictly connected to the channel list configuration. Together with this parameter, you must also configure the channel list for scanning. To configure the channel list, see the Configure URWB channel list (CLI) section.

Procedure

---

1. Enter configuration mode.

Example

```
Device# configure terminal
```

2. Enter a profile name to create a URWB profile.

   Example

   ```
   Device(config)# wireless profile urwb urwb-profile-name
   ```

   This command creates a URWB profile and enters into the wireless URWB profile configuration mode.

3. Configure URWB mobility scan.

   Example

   ```
   Device(config-wireless-urwb-profile)# mobility scan {isolation scan-wait-time
    0-65535| periodic scan-period 0-65535 | rssi-threshold threshold-value 0-96}
   Device(config-wireless-urwb-profile)# no mobility scan isolation
   Device(config-wireless-urwb-profile)# no mobility scan periodic
   ```

   Use the **no** form of the command to disable isolation and periodic time for scanning.

The URWB mobility scan parameters are configured.

Example

```
Device# configure terminal
Device(config)# wireless profile urwb urwb-profile-name
Device(config-wireless-urwb-profile)# mobility scan {isolation scan-wait-time
 0-65535| periodic scan-period 0-65535 | rssi-threshold threshold-value 0-96}
Device(config-wireless-urwb-profile)# no mobility scan isolation
Device(config-wireless-urwb-profile)# no mobility scan periodic
```

Configure URWB MPLS parameters (CLI)

Complete the following steps to configure URWB MPLS dataframe packets for a specified device:

Procedure

---

1. Enter configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter a profile name to create a URWB profile.

   Example

   ```
   Device(config)# wireless profile urwb urwb-profile-name
   ```

   This command creates a URWB profile and enters into the wireless URWB profile configuration mode.

3. Configure MPLS Ethernet filter, filtering method.

Example

```
Device(config-wireless-urwb-profile)# [no] mpls ether-filter ethertypes
filtering {allow-all | use-allow-list}
```

This command allows all ethertypes or uses the configured ethertypes allowed list.

Use the **no** form of the command to disable MPLS ether-filter filtering method.

4. Configure Ethernet filter allow-list and add the Ethernet type value to the allow-list.

Example

```
Device(config-wireless-urwb-profile)# [no] mpls ether-filter ethertypes
allow-list add ether-type-value
```

Here, the value of the Ethernet type is from 0 to 65535.

Use the **no** form of the command to remove ether-type from URWB ether-filter. allow-list .

5. Configure ethernet I frames forwarding.

Example

```
Device(config-wireless-urwb-profile)# [no] mpls ether-filter ethernet-I
```

Here, the value of the Ethernet type is from 0 to 65535.

Use the **no** form of the command to disable ether-filter Ethernet-I frames.

6. Configure high-availability.

Example

```
Device(config-wireless-urwb-profile)# mpls high-availability
Device(config-wireless-urwb-profile)# [no] mpls high-availability
```

Use the **no** form of the command to disable high-availability.

7. Configure high-availability time out for device failure detection.

Example

```
Device(config-wireless-urwb-profile)# mpls high-availability timeout
timeout-value
```

Here, the *timeout-value* is the high-availability node failure detection timeout. The value is from 0 to 65535 ms.

The URWB MPLS parameters are configured.

Example

```
Device# configure terminal
Device(config)# wireless profile urwb urwb-profile-name
Device(config-wireless-urwb-profile)# [no] mpls ether-filter ethertypes
filtering allow-all
Device(config-wireless-urwb-profile)# [no] mpls ether-filter ethertypes
allow-list add 50
```

```
Device(config-wireless-urwb-profile)# [no] mpls ether-filter ethernet-I
Device(config-wireless-urwb-profile)# mpls high-availability
Device(config-wireless-urwb-profile)# mpls high-availability timeout 54
Device(config-wireless-urwb-profile)# [no] mpls high-availability
```

Configure multicast for URWB profile (CLI)

Complete the following steps to configure the multicast settings for URWB profile:

Procedure

1. Enter configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter a profile name to create a URWB profile.

   Example

   ```
   Device(config)# wireless profile urwb urwb-profile-name
   ```

   This command creates a URWB profile and enters into the wireless URWB profile configuration mode.

3. Configure multicast settings.

   Example

   ```
   Device(config-wireless-urwb-profile)# multicast {all | coordinator-only}
   Device(config-wireless-urwb-profile)# [no] multicast
   ```

   This command enables multicast rule to all devices or only to the coordinator device.

   Use the **no** form of the command to disable multicast.

The URWB multicast settings are configured.

Example

```
Device# configure terminal
Device(config)# wireless profile urwb urwb-profile-name
Device(config-wireless-urwb-profile)# multicast coordinator-only
Device(config-wireless-urwb-profile)# [no] multicast
```

Limitations and guidelines for secondary Ethernet ports

The limitations and guidelines for secondary Ethernet ports highlight key operational constraints and configuration considerations to ensure proper deployment and functionality of the feature.

When deploying secondary Ethernet ports, observe these constraints:

- **Redundancy Impact** : Activating the secondary Ethernet port disables dual-Ethernet redundancy (backup mode).
- **VLAN Configuration**: The primary and secondary Ethernet ports must share an identical VLAN configuration.

- **State Changes**: Any modification to the secondary port state (enabled or disabled) requires a device reboot for the change to take effect.
- **Model-Specific Restrictions**: For the Cisco Catalyst 9124AX Series model, the third port (LAN1) is limited to client connections or daisy-chaining and cannot operate as a root port.
- **Mobility Infrastructure Limitation**: In a mobility-configured network with two mesh-end devices, the mobility infrastructure must be deployed using a ring topology. The two mesh-end APs must be positioned as the first and last APs in the ring and must be directly connected to the controller. Additionally, both mesh-end APs are required to reach the controller through the same wired port, designated as the root port, to ensure proper network operation and failover capabilities.

Configure secondary Ethernet port (GUI)

Configure the secondary Ethernet port on supported access point models to enable wired client connectivity and optional PoE functionality.

Configure the secondary Ethernet port to enable wired client connectivity and optional PoE functionality.

From release 26.2.1, the URWB stack supports permanently enabling two wired interfaces, allowing independent management of Wired0 and Wired1 ports and enabling daisy-chaining or mesh ring deployments without external switches. This feature restores parity with standalone Industrial Wireless software and enhances flexibility and cost efficiency in URWB deployments. Enable optional PoE on the secondary Ethernet port to provide wired client support and power to downstream devices.

Before you begin

Ensure that your access point model supports configuration of a secondary Ethernet port and Power over Ethernet (PoE). Only the Cisco Catalyst 9124AX series supports PoE on port ID 2.

Procedure

1. Log in to the Controller web interface.
2. Choose **Configuration > Tags & Profiles > URWB Network Profile**.
3. Select the **Wired Clients** tab.
4. Go to the **Ethernet settings** section.
5. Choose the physical port ID to map as the secondary logical interface or disable the secondary Ethernet interface.
6. Enable the **LAN Port PoE** toggle if your device model supports powering downstream devices.

   > **Note**
   >
   > PoE functionality is supported only on LAN1 port of Cisco Catalyst 9124AX series.

7. Click **Save**.
8. Reboot the device.

What to do next

1. Navigate to **Monitoring > Wireless > AP Statistics** on the wireless controller interface.
2. Select the specific AP under the **General** tab to access its detailed statistics.

3. Click the **URWB** sub-tab to confirm that the secondary Ethernet configuration changes have been correctly applied after the device reboot.

> **Note**
>
> Hot configuration support is limited to changing the mapping of the secondary URWB port to a different physical port (e.g., from LAN1 to LAN2).

Configure secondary Ethernet port (CLI)

Configure the secondary Ethernet port on supported Cisco access points using CLI commands to enable wired client connectivity and optional PoE functionality.

Configure the secondary LAN port ID on compatible Cisco access points. Enable optional PoE on the secondary Ethernet port to provide wired client support and power to downstream devices.

Before you begin

- Identify the correct LAN Port ID for your hardware (supported range: 1 to 4). You can use the **show interface status** command in privileged EXEC mode to list the LAN Port IDs of the device.
- Ensure that your access point model supports configuration of a secondary Ethernet port and Power over Ethernet (PoE). Only the Cisco Catalyst 9124AX series supports PoE on port ID 2.

Procedure

1. Use the **wireless profile urwb** *profile-name* command to enter the URWB profile configuration mode.

   Example

   ```
   Device# wireless profile urwb Cisco-AP
   ```

2. Use the **ethernet secondary lan id** *port-id* command to configure the secondary LAN.

   Example

   ```
   Device# ethernet secondary lan id 1
   ```

   The supported port ID range is 0 to 4. On most supported access point models, Port ID 1 corresponds to the secondary Ethernet port. Setting the port ID to zero effectively disables the secondary port in URWB.

3. (Optional) Use the **ethernet secondary lan poe** command to enable PoE-out on the secondary port.

   Example

   ```
   Device# ethernet secondary lan poe
   ```

> **Note**
>
> PoE functionality is supported only for the Cisco Catalyst 9124AX access point series on port ID 2.

4. (Optional) Use the **show ap name** *ap_name* **urwb info** command to verify that the secondary Ethernet port is active and configured correctly.

Example

```
Device# show ap name Cisco-AP urwb info
```

```
Secondary Ethernet
  Port state                          : Disabled
  PoE out                             : Disabled
```

Create URWB radio profile (GUI)

Describes the steps to create a URWB radio profile.

The URWB wireless radio profile is used to configure a particular radio slot for the URWB backhaul operation. The URWB wireless radio profile, when applied to a radio interface on a device, is relevant only if the device is in the URWB mode.

To set up a backhaul link with URWB, configure the URWB radio profile to a radio interface. Follow these steps to create a URWB radio profile:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF/Radio**.
2. Click the **Radio** tab.
3. Click **Add**.

   The **Add Radio Profile** window is displayed.
4. In the **Add Radio Profile** window, click the **General** tab and enter the radio profile name.
5. Click the **URWB** tab.
6. From the **URWB Radio Role** drop-down list, choose one of the options:
   - Disabled: Disables the URWB radio role. This is the default value.
   - Fixed
   - Mobility
   - Fixed Point to Multi Point
7. From the **Radio Band** drop-down list, choose one of the radios:
   - 5 GHz
   - 6 GHz

8. From the **Channel** drop-down list, choose the appropriate channel.
9. From the **Channel width** drop-down list, choose one of the channel widths:
   - 20 MHz
   - 40 MHz
   - 80 MHz
   - 160 MHz
10. In the **Encryption** section, choose a type of encryption from the **AES** drop-down list.
   - Disabled: Disables the encryption. This is the default value.
   - Fixed Key
   - Rotating Key: When you choose the Rotating Key encryption, enter the **Key Rotation Timeout** in seconds. The value must be in the range of 1 to 65,535. The default is 15.

> **Note**
>
> Configuring the rotating key can impact mobility performance.

11. In the **Fixed Point to Multi Point** section, configure the following:
   a) Check the **Auto Scan** check box to allow clients to perform frequency scanning when the RSSI from the base is below the threshold or when there is no signal.
   b) Enter the cluster ID to identify a cluster formed of base(s) and client(s).

> **Note**
>
> Devices with different cluster IDs do not connect in one topology.

   c) Enter the tower ID.

> **Note**
>
> Redundant pair of base(s) must be assigned the same unique tower ID.

   d) Enter the **RSSI Threshold** to build a path to the base. The range is from 0 to 100. The default is 0.
12. Click **Apply to Device** to save the configuration.

The URWB radio profile has been created.

What to do next

Assign the URWB profiles to the wireless RF tag.

Create RF tags and assign profiles (GUI)

Complete these steps to create RF tags and assign profiles:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**, click the **RF** tab, and then click **Add**.

   The **Add RF Tag** page is displayed.
2. In the **Add RF Tag** page, enter the name and description of the RF tag.
3. Select the required **6 GHz Band RF Profile**, **5 GHz Band RF Profile**, and **2.4 GHz Band RF Profile** to be associated with the RF tag.
4. Select an appropriate **URWB Network Profile** for the RF tag.
5. Choose the required **6 GHz Radio Profile**, **5 GHz Radio Profile**, and **2.4 GHz Radio Profile** to be associated with the RF tag.
6. Click **Apply to Device** to save the configuration.

The RF tags are created and profiles are assigned.

Create URWB radio profile (CLI)

Describes the CLI steps to create a URWB radio profile.

Complete the following steps to configure the URWB radio profile:

Procedure

1. Enter configuration mode to start creating a URWB radio profile.

   Example

   ```
   Device# configure terminal
   ```
2. Enter a profile name to create a wireless radio profile.

   Example

   ```
   Device(config)# wireless profile radio urwb-radio-profile-name
   ```

   This command creates a wireless radio profile and enters into the wireless profile radio configuration mode.
3. Configure URWB channel radios in the 5-GHz or 6-GHz bands.

   Example

   ```
   Device(config-wireless-radio-profile)# urwb channel {5ghz | 6ghz} channel
   ```

   The channel range is from 1 to 233.
4. Configure URWB channel width.

Example

```
Device(config-wireless-radio-profile)# urwb cwidth {160MHz | 20MHz | 40MHz |
 80MHz}
```

The valid channel widths are 160MHz, 20MHz, 40MHz, and 80MHz.

5. Configure URWB role of the radio, as fixed, mobility, or point-to-multi-point.

Example

```
Device(config-wireless-radio-profile)# urwb role {fixed | point-to-multi-point
 | mobility}
```

6. Exit the sub-mode.

Example

```
Device(config-wireless-radio-profile)# exit
```

The URWB radio profile is created.

Example

```
Device# configure terminal
Device(config)# wireless profile radio urwb-radio-profile-name
Device(config-wireless-radio-profile)# urwb channel 5ghz 120
Device(config-wireless-radio-profile)# urwb cwidth 160MHz
Device(config-wireless-radio-profile)# urwb role fixed
Device(config-wireless-radio-profile)# exit
```

Configure URWB point-to-multi-point parameters (CLI)

Complete the following steps to configure the URWB fixed point-to-multi-point (PTMP) parameters:

Procedure

---

1. Enter configuration mode to create a URWB radio profile.

Example

```
Device# configure terminal
```

2. Enter a profile name to create a wireless radio profile.

Example

```
Device(config)# wireless profile radio urwb-radio-profile-name
```

This command enters into the wireless profile radio configuration mode.

3. Configure URWB PTMP cluster ID to identify a cluster formed of base(s) and client(s).

Example

```
Device(config-wireless-radio-profile)# urwb ptmp cluster-id cluster-id
```

The maximum length of the cluster ID is 64 characters long.

> **Note**
>
> You must not set the cluster ID as an empty value or with ''.

Cluster ID: When more than one point-to-multipoint cluster operates in the same location, each cluster can be logically separated from every other cluster by using a cluster ID value that is unique to that cluster. A Fixed client radio will only communicate wirelessly with a Fixed Base radio that shares the same cluster ID.

> **Note**
>
> This setting is applicable only if the URWB role is set to point-to-multi-point.

4. Enable URWB PTMP autoscan for radio slot.

Example

```
Device(config-wireless-radio-profile)# [no] urwb ptmp autoscan
```

Use the **no** for of this command to disable URWB PTMP autoscan.

> **Note**
>
> This setting is applicable only if the URWB role is set to point-to-multi-point and if the URWB PTMP role is secondary.

5. Set the URWB PTMP tower ID for radio slot.

Example

```
Device(config-wireless-radio-profile)# [no] urwb ptmp tower-id tower-id
```

Use the **no** for of this command to disable URWB PTMP tower ID.

> **Note**
>
> This setting is applicable only if URWB role is set to point-to-multi-point and if URWB PTMP role is Fixed Base.

Tower ID: To avoid connectivity issues when multiple towers are serving the same cluster, the **tower-id** parameter can be enabled on Base radios.

All radios mounted on the same tower share the same tower ID. Even if two base radios see each other through a physical link and wirelessly, they know that they are on a different tower and will continue to advertise themselves.

When a client is connected to one tower (Tower 1) and moves close to another one (Tower 2), it automatically connects to a new base on Tower 2. This gives a wider coverage area and continuous connectivity. The same tower design, with different tower IDs, can be replicated to provide more coverage zones.

6. Configure URWB PTMP RSSI threshold for radio slot.

   Example

   ```
   Device(config-wireless-radio-profile)# [no] urwb ptmp rssi-threshold
   threshold-value
   ```

   Use the **no** form of this command to disable URWB PTMP RSSI threshold.

   Note

   This setting is appliable only if URWB role is set to point-to-multi-point and if URWB PTMP role is Fixed Client.

The URWB PTMP parameters are configured.

Example

```
Device# configure terminal
Device(config)# wireless profile radio urwb-radio-profile-name
Device(config-wireless-radio-profile)# urwb ptmp cluster-id cluster-id
Device(config-wireless-radio-profile)# [no] urwb ptmp autoscan
Device(config-wireless-radio-profile)# [no] urwb ptmp tower-id tower-id
Device(config-wireless-radio-profile)# [no] urwb ptmp rssi-threshold
threshold-value
```

Configure URWB encryption parameters (CLI)

Complete the following steps to configure the URWB encryption parameters:

Procedure

---

1. Enter configuration mode to create a URWB radio profile.

   Example

   ```
   Device# configure terminal
   ```

2. Enter a profile name to create a wireless radio profile.

   Example

   ```
   Device(config)# wireless profile radio urwb-radio-profile-name
   ```

   This command creates a wireless radio profile and enters into the wireless profile radio configuration mode.

3. Configure AES-based encryption policy with fixed key or rotating key.

Example

```
Device(config-wireless-radio-profile)# urwb crypto policy {none | fixed-key
| rotating-key}
```

Use the **none** command to disable the AES-based encryption.

4. Configure AES-based encryption rotating key timeout.

Example

```
Device(config-wireless-radio-profile)# urwb crypto rotating-key-timeout timeout
```

Here, the *timeout* range is from 0 to 65535 seconds.

> **Note**
>
> This setting is applicable only if the URWB crypto policy is set to **rotating-key**.

The URWB encryption parameters are configured.

Example

```
Device# configure terminal
Device(config)# wireless profile radio urwb-radio-profile-name
Device(config-wireless-radio-profile)# urwb crypto policy {none | fixed-key |
 rotating-key}
Device(config-wireless-radio-profile)# urwb crypto rotating-key-timeout 120
```

Configure URWB channel list (CLI)

Complete the following steps to configure the URWB channel list for mobility scanning:

Procedure

1. Enter configuration mode to create a URWB radio profile.

Example

```
Device# configure terminal
```

2. Enter a profile name to create a wireless radio profile.

Example

```
Device(config)# wireless profile radio urwb-radio-profile-name
```

This command creates a wireless radio profile and enters into the wireless profile radio configuration mode.

3. Add channel or channel-width to URWB channel list for radio slot.

Example

```
Device(config-wireless-radio-profile)# [no] urwb chan-list channel channel
cwidth channel_width
```

Use the **no** for of this command to remove channel or channel-width from the URWB channel list for radio slot.

The URWB channel list is configured for mobility scanning.

Example

```
Device# configure terminal
Device(config)# wireless profile radio urwb-radio-profile-name
Device(config-wireless-radio-profile)# [no] urwb chan-list channel 100 cwidth
 40MHz
```

Create RF tags and assign profiles (CLI)

Complete the following steps to map the radio profile to RF tag and radio slot:

Procedure

---

1. Enter configuration mode to create wireless RF tags.

   Example

   ```
   Device# configure terminal
   ```

2. Enter a profile name and create a wireless RF tag.

   Example

   ```
   Device(config)# wireless tag rf rf-profile-name
   ```

   This command configures the wireless RF tag and goes into the wireless RF tag profile configuration mode.

3. Configure URWB profile.

   Example

   ```
   Device(config-wireless-rf-tag)# urwb-profile urwb_profile
   ```

4. Configure the URWB radio profile to the RF tag.

   Example

   ```
   Device(config-wireless-rf-tag)# dot11 5ghz slot1 radio-profile
   urwb-radio-profile-name
   ```

> **Note**
>
> Use the `dot11 6ghz <slot> radio-profile radio-profile-name` command to attach radio profile to a 6 GHz radio configuration. For example, `Device(config- wireless-rf-tag)# dot11 6ghz slot3 radio-profile urwb_radio_profile`

5. Exit the sub-mode.

Example

```
Device(config-wireless-rf-tag)# exit
```

The RF tags are created and assigned to the URWB profiles.

Example

```
Device# configure terminal
Device(config)# wireless tag rf rf-profile-name
Device(config-wireless-rf-tag)# urwb-profile urwb_profile
Device(config-wireless-rf-tag)# dot11 5ghz slot1 radio-profile
urwb-radio-profile-name
Device(config-wireless-rf-tag)# exit
```

> **Note**
>
> In the RF tag configuration, if you assign radio profiles on 2 radio slots in 5-GHz band (**dot11 5ghz slot# radio-profile** *profile-name*. Following restrictions apply:
>
> - The channels configured are separated by 100 MHz.
> - The channels cannot be in the same UNII2 Ext channels.

What to do next

- Assign the URWB coordinator role to one of the APs.
- Configure URWB DFS parameters.
- Assign the URWB Fixed Base role to the selected APs.

Configure URWB parameters in the privileged EXEC mode

Describes the process to configure URWB parameters in the Privileged EXEC mode.

The URWB parameters in the EXEC mode are Fixed Base and Fixed Point-to-Multipoint.

- Fixed Base in a point-to-multipoint topology, is the radio selected by the clients to connect to. Clients only connect to Fixed Base radios.
- If Fixed Point-to-Multipoint is enabled, at least one AP has to be configured as a Fixed Base.

Configure URWB parameters per radio (GUI)

Describes the steps to configure the URWB parameters per radio using GUI steps.

Complete the following steps to configure URWB parameters per radio:

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click the **5 GHz Radios** section.
3. Select an AP from the list of APs displayed.

   The edit page is displayed.
4. In the **URWB** section, use the toggle button to enable or disable the **Fixed Point to Multipoint Base [Config on WLC]** parameter.

   > **Note**
   >
   > Reboot the AP for F**ixed Point to Multipoint Base** configuration on the controller to synchronize with the configuration on the AP.
5. Click **Update & Apply to Device**.
6. Click **All Access Points** and select the required AP.

   The **Edit AP**page is displayed.
7. Click the **URWB** tab.
8. In the **URWB Coordinator [Config on WLC]** section, check the **Coordinator** check box to configure the AP as a coordinator.

   This option tags a device to the coordinator role. Coordinators are the entry or exit points of the traffic to or from the infrastructure side.

   Reboot the AP to synchronize configurations.
9. Optional: Check the **Wired-Only** coordinator, and select either **Fixed** or **Mobility**.

   The **URWB Coordinator [Config on AP]** section displays the coordinator status on the AP.
10. Click **Update & Apply to Device** to save the configuration.

The URWB parameters are configured per radio.

Configure URWB parameters in the privileged EXEC mode (CLI)

Describes the steps to configure URWB parameters in the privileged EXEC mode.

Procedure

1. Configure URWB PTMP parameters.

Example

```
Device# ap name Cisco-AP dot11 5ghz slot slot-number urwb ptmp base
Device# ap name Cisco-AP no dot11 5ghz slot slot-number urwb ptmp base
```

2. Configure URWB DFS parameters.

Example

```
Device# ap name Cisco-AP dot11 5ghz slot slot-number urwb dfs {auto | primary
 | secondary}
```

3. Configure URWB coordinator mode.

Example

```
Device# ap name Cisco-AP urwb mode coordinator
Device# ap name Cisco-AP no urwb mode coordinator
```

4. Configure URWB coordinator radio-off mode.

Example

```
Device# ap name Cisco-AP urwb mode coordinator radio-off {fixed | mobility}
```

The URWB parameters are configured in the privileged EXEC mode.

Example

```
Device# ap name Cisco-AP dot11 5ghz slot slot-number urwb ptmp base
Device# ap name Cisco-AP no dot11 5ghz slot slot-number urwb ptmp base
Device# ap name Cisco-AP dot11 5ghz slot slot-number urwb dfs {auto | primary
 | secondary}
Device# ap name Cisco-AP urwb mode coordinator
Device# ap name Cisco-AP no urwb mode coordinator
Device# ap name Cisco-AP urwb mode coordinator radio-off fixed
```

Verify URWB details

Describes the verification section of the URWB feature.

Verify URWB profile details

To display the profile details specific to a URWB profile, use the following command:

```
Device# show wireless profile urwb detailed urwb-profile
```

Verify details about specific wireless radio profile

To display details about specific wireless radio profile, including the URWB information, use the following command:

```
Device# show wireless profile radio detailed radio_profile_name
```

Verify URWB details for a specific AP

To display the URWB details for a specific AP, use the following command:

```
Device# show ap name ap_name urwb info
```

Verify the configuration status from a URWB AP

To display the configuration status from a URWB AP, use the following command:

```
Device# show ap urwb status
```

Verify URWB details for an AP in a radio band

To display the URWB details for an AP in a radio band, use the following command:

```
Device# show ap name Cisco-AP dot11 band slot slot-id urwb detail
```

URWB telemetry protocol

URWB telemetry protocol provides real-time wireless network performance monitoring by sending structured UDP packets containing key metrics such as signal strength, latency, jitter, and packet loss.

The URWB telemetry protocol feature performs external monitoring of real-time wireless performance. This feature sends pre-defined, structured UDP packets at regular intervals, and these packets contain network metrics. Third-party and custom applications can use telemetry data. An application that receives this data can interpret it live or capture and process it later. The telemetry UDP packets includes:

- signal strength and Link Error Rate,
- latency,
- jitter,
- packet throughput,
- number of transmissions and retransmissions,
- modulation rate,
- details of packet loss,
- operating frequency of each radio, and
- information about the events that record the network.

Configure URWB telemetry protocol using AP CLI

Enable or disable URWB telemetry protocol

Perform this task to manage the URWB telemetry export functionality on the device. Enabling or disabling the protocol ensures the telemetry export aligns with your network management requirements.

Procedure

Use the configure **urwb telemetry export enable** command to enable the URWB telemetry export.

```
Device#configure urwb telemetry export enable
```

Or

Use the **configure urwb telemetry export disable** command to disable the URWB telemetry export.

```
Device#configure urwb telemetry export disable
```

Configure URWB telemetry server

Before you begin

Perform this task to configure the address and port number of the URWB telemetry server, ensuring successful communication between the device and the server.

Procedure

1. Use the **configure urwb telemetry server address** command to configure the server address.

   ```
   Device#configure urwb telemetry server a.b.c.d
   ```

2. Use the **configure urwb telemetry server address Port number** command to configure port number of telemetry server.

   ```
   Device#configure urwb telemetry server a.b.c.d 30000
   ```

   Example

   ```
   Device#configure urwb telemetry server a.b.c.d <0-65535>
   ```

   Example

   ```
   Device#configure urwb telemetry server
     x.x.x.x  IP address of telemetry server (A.B.C.D)
   Device#configure urwb telemetry server a.b.c.d
     <0-65535>  Port number of telemetry server (NUM)
     <cr>
   Device#configure urwb telemetry server a.b.c.d 9
     <cr>
   ```

> **Note**
>
> Port number valid range is from 0 to 65535.

Verify telemetry configuration

This task ensures that the telemetry configuration is correctly set up on the device, allowing for proper monitoring and reporting of telemetry data.

Procedure

Use the **show urwb telemetry config** command to view the telemetry configuration.

Example

```
Device#show urwb telemetry config

Telemetry export: disabled, current (live): disabled

Telemetry server: 0.0.0.0 30000, current (live): 0.0.0.0 30000

Telemetry level: default, current (live): default
```

Monitor URWB network information using Controller (GUI)

Describes the GUI steps to monitor URWB network information using the controller.

Complete these steps to monitor the URWB network information:

Procedure

1. Choose **Monitoring** > **Wireless** > **URWB Monitoring** to view the URWB network statistics.

   The **URWB Network Topology** page is displayed.

   The **URWB Network Topology** page displays a graphical representation of the network that includes AP nodes. The page also features the **URWB Coordinator** drop-down list.

   > **Note**
   >
   > The **Coordinator** or mesh-end is responsible for collecting and sending the URWB network statistics to the controller.

2. From the **URWB Coordinator** drop-down list, select the coordinator AP Name to view the network topology graph associated with the selected coordinator.

The page features the total number of APs connected, and the type of link used to connect to the AP nodes.

3. Click individual nodes or APs in the graph to view specific details, such as:
   - **Mesh ID**
   - **AP Name**
   - **IP Address**
   - **Device Model**
   - **AP Role**
   - **Connected Devices Count**

> **Note**
>
> The connections or links between APs could be either wired or wireless. Wireless links are represented in blue and wired links are represented in black.

4. Click the wireless link to view the **Uplink** and **Downlink** statistics for the link, such as:
   - **Throughput(Mbps)**
   - **LER**: Link Error Rate - Measurement of link errors
   - **PER**: Packet Error Rate - Percentage of packets with errors
   - **RSSI**: Received Signal Strength Indicator
   - **MCS | Rate**: Modulation and coding scheme

The refresh interval for monitoring statistics is two seconds

Verify AP URWB information

To display the AP URWB information, use these commands:

Table 170: AP URWB commands and description

Show commands	Description
show ap name *ap-name* **urwb coord-routes**	Displays the URWB coordinator routes.
show ap name *ap-name* **urwb fixed-links**	Displays the URWB fixed links reported by the coordinator AP.
show ap name *ap-name* **urwb mobility-links**	Displays the URWB mobility links reported by the coordinator AP.
show ap name *ap-name* **urwb network**	Displays the URWB network nodes reported by the coordinator AP.
show ap name *ap-name* **wireless-stats**	Displays the URWB wireless link statistics reported by the coordinator AP.

URWB recovery

URWB recovery enables access points to automatically troubleshoot and restore their connection to the network controller, ensuring seamless connectivity and efficient operations in wireless networks.

The URWB recovery commands are supported on access point (APs) to troubleshoot and restore connectivity when communication with the controller is lost. These commands facilitate the recovery process and ensure that APs can reconnect and resume normal operations.

URWB recovery includes these main features:

- Helps APs troubleshoot connectivity issues.
- Automatically downloads the correct configuration after the connection is restored and eliminates the need for manual intervention.
- Returns the AP to a fully functional state after recovery.

Configure URWB recovery to maintain network efficiency and seamless connectivity.

Prerequisites

- If the device has a CAPWAP connection, the URWB recovery commands are rejected.
- Ensure you have physical access to the AP's console.
- Know the correct alpha-2 country code for the AP's deployment location.

Configure the URWB recovery using AP CLI

Complete these steps to configure the URWB recovery using AP commands.

Procedure

1. Use the **configure urwb recovery conf-reset** command to reset the URWB configuration and disable the URWB mode.

   Example

   ```
   Device#configure urwb recovery conf-reset
   ```

   Warning

   In URWB mode, this command erases all configurations and resets the device to factory default settings in CAPWAP mode.

2. Use the **configure urwb country-code** *country-code* command to configure the country code for initial provisioning.

   Example

   ```
   Device#configure urwb recovery country-code US
   ```

> **Note**
>
> - Enter the correct Alpha-2 country code (for example, US, IN, GB).
> - Set the country code before changing the radio slot to ensure correct application of the parameters.
> - For WiFi-7 PIDs, the device will prompt you to reload the AP.

3. Use the **configure urwb recovery network-key** *key* command to configure the network-key on the radio interface.

   Example

   ```
   Device#configure urwb recovery network-key 123456789
   ```

   > **Note**
   >
   > The network-key must contain at least 8 characters and can be up to 64 characters. Special characters such as "~|;$<>&^`?/ and "" are not permitted.

4. Use the **configure urwb recovery dot11Radio** *interface* **role {fixed | mobility-client}** command to configure the role on the radio interface. It can be either fixed or mobility-client.

   Example

   ```
   Device#configure urwb recovery dot11Radio 1 role fixed
   ```

   > **Note**
   >
   > If you need to change both the country code and the radio slot role, set the country code first to ensure the correct application of the parameters.

5. Use the **configure urwb recovery dot11Radio** *interface* **band** *band* **channel** *channel* **bandwidth** *bandwidth* command to configure the radio interface settings.

   Example

   ```
   Device#configure urwb recovery dot11Radio 1 band 5ghz channel 36 bandwidth
   20
   ```

   - The radio *interface* supports 5 GHz and 6 GHz bands. Set the band ID is correctly before configuration.
   - Operating *channel* range is 1 to 256.
   - Supported radio *bandwidth* options are 20 MHz, 40 MHz, 80 MHz, and 160 MHz.

6. Use the **configure urwb recovery dot11Radio** *interface* **encryption {aes-fixed-key | aes-rotating-key | disable}** command to assign an encryption to the selected radio interface.

   Example

   ```
   Device#configure urwb recovery dot11Radio 1 encryption aes-fixed-key
   ```

7. Use the **reload** command to apply the changes.

   Example

   ```
   Device#reload
   ```

   Note

   The configuration is saved automatically.

8. Use **configure urwb recovery dot11Radio** *interface* **dual-band** *band* **{5GHz | 6GHz | 24GHz}**command to configure the URWB dual-band mode.

   Example

   ```
   Device#configure urwb recovery dot11Radio 1 dual-band 5ghz
   ```

9. Use **configure urwb recovery dot11Radio** *interface* **dual-radio mode** *mode* **{enable disable}**command to configure the URWB dual-radio mode.

   Example

   ```
   Device#configure urwb recovery dot11Radio 1 dual-radio mode enable
   ```

Country code behavior on AP models

The behavior of the country code, when configured through the recovery CLI, depends on the Access Point (AP) model.

Country code behavior on WiFi-6 PIDs (IW916x -ROW models)

Note

The recovery CLI does not support country code configuration for non-ROW Wi-Fi 6 PIDs.

- Country code configuration is applicable for ROW PIDs.
- The system prevents configuration for non-ROW PIDs.
- The country code you configure determines the valid frequencies and bandwidths allowed in the regulatory domain.

Country code behavior on WiFi-7 PIDs

> **Note**
>
> For WiFi-7 PIDs, reboot the AP after configuring the country code through the recovery CLI. This ensures the changes take effect.

- The configured country code is used only for the initial connection.
- After the AP joins the controller, the regulatory configuration received from the controller is automatically applied to the AP.
- If the country code has already been programmed through the controller, this initial configuration step is not necessary.

URWB recovery configuration example

These commands allow users to quickly configure and customize the URWB recovery settings for optimal wireless network operation during recovery scenarios.

Procedure

1. Use the **configure urwb recovery dot11Radio 1** command to select the radio interface for configuration.

   Example

   ```
   Device#configure urwb recovery dot11Radio 1
   ```

2. Use the **configure urwb recovery dot11Radio 1 role** *radio-role* command to set the radio role.

   Example

   ```
   Device#configure urwb recovery dot11Radio 1 role ptmp
   ```

   The radio role is set to Point-to-Multipoint (PtMP)

3. Use the **configure urwb recovery dot11Radio 1 role** *radio-role* **cluster-id** *id* command to assign a cluster ID to the radio interface.

   Example

   ```
   Device#configure urwb recovery dot11Radio 1 role ptmp cluster-id test
   ```

4. Use the **configure urwb recovery dot11Radio 1 band** *band* **channel** *channel* **bandwidth** *bandwidth* command to configure the frequency and channel parameters.

   Example

   ```
   Device#configure urwb recovery dot11Radio 1 band 5ghz channel 36 bandwidth
   20
   ```

5. Use the **reload** command to apply the configuration and restart the access point.

Example

```
Device#reload
```

The access point restarts and the new recovery settings take effect.

Dynamically configure URWB parameters

Adjust URWB parameters dynamically on APs without rebooting, improving network management and reducing service interruptions.

This feature allows you to adjust certain Ultra-Reliable Wireless Backhaul (URWB) configuration parameters on supported access points (APs) without needing to reboot the device. This makes network management smoother and reduces service interruptions.

Hot-configurable URWB parameters

These URWB parameters can be updated dynamically, without requiring a device reboot.

- Radio frequency and bandwidth
- Radio DFS role
- Radio DFS backup channel list
- Radio crypto and key-control rotation timeout
- Network key
- MPLS ethertype
- MPLS allow Ethernet I
- MPLS ether filter method
- MPLS unicast flood
- MPLS unicast flood limitation
- Mobility role (base / base-relay)
- Mobility scan parameters (idle, after, RSSI threshold)
- Mobility RSSI hysteresis
- Mobility times (warmup, timeout)
- Mobility backhaul checks
- P2MP primary
- P2MP parameters (RSSI threshold, autoscan, cluster ID, tower ID)
- High-availability enablement
- High-availability timeout
- Multicast
- MPO enablement
- MPO fine-tuning parameters (maximum links, minimum RSSI, QoS classification)

- MPO telemetry enablement

Parameters requiring a device reload

These URWB parameters cannot be changed dynamically. The device must be rebooted for the changes to take effect. You can use either the **reload** command or use the **Reload** button in the **URWB** tab of the **Monitoring > Wireless > AP Statistics** page.

- URWB global enablement
- Coordinator mode
- Switch between P2MP base and P2MP client
- Radio role (disabled, fixed, P2MP, mobility)
- Band ID

> **Note**
>
> Changing the country code requires a device reload.

Configure URWB parameters using CLI

This task helps you modify hot-configurable URWB parameters and safely apply them to the network device.

> **Note**
>
> To prevent partial or inconsistent configurations, changes are first staged, and then explicitly applied.

Before you begin

- Confirm the parameters you intend to update are hot-configurable. See the Hot-configurable URWB parameters section for the supported list of parameters.
- Ensure that the required APs are operating in URWB mode.

Follow these steps to apply configuration changes to URWB parameters:

Procedure

1. Select the AP to which the configuration is to be applied.

> **Note**
>
> When updating configurations in a wireless mesh network, apply the configuration to the remote devices first, and then to the Coordinator AP. This helps prevent loss of connectivity to the rest of the network during the update.

2. After making the required configuration changes, verify that the Config Status field is set to 'Live update required.'

   See the View Configuration Update Status Summary section on how to view the summary of the configuration updates.

3. Apply the configuration changes using the **ap name** *ap-name* **urwb apply-running-config** command.

   > **Note**
   >
   > The Config Status field transitions to 'In progress.'

4. After the configuration is successfully applied, confirm that the Config Status field is set to 'Up to date.'

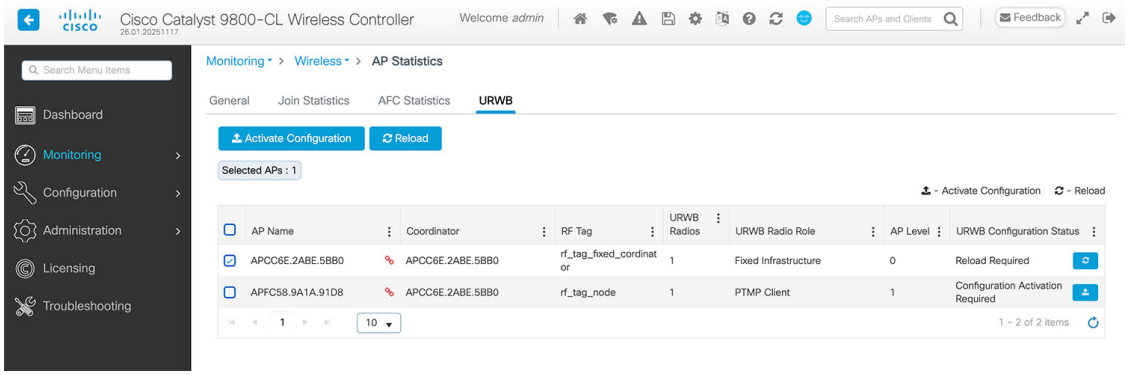
   See the View Configuration Update Status Summary section on how to view the summary of the configuration updates.

Configure URWB parameters using GUI

This task helps you modify hot-configurable URWB parameters and apply them to the network device.

Procedure

1. Choose **Configuration > Tags & Profiles > URWB Network Profile**.
2. Click the URWB Network Profile name to open the **Edit URWB Network Profile** page.
3. Select the appropriate tab and make the necessary changes.
4. Click **Update and Apply on Device**. The system displays a message confirming successful configuration.
5. Choose **Monitoring > Wireless > AP Statistics**, and select the **URWB** tab.



6. Activate the configuration using one of these methods:

   - Click the  buttom displayed on the URWB Configuration Status field of the AP.

- Select the AP and click the 

button on the page.

7. (Optional) Reload the device. Use this option only if URWB parameter changes require a device reboot. Use one of these methods:
   - Click the  button displayed on the URWB Configuration Status field of the AP.
   - Select the AP name and click the 

button on the page.

View configuration update summary

You can view the summary of the configuration updates status using the command given here.

- show ap urwb status

```
controller# show ap urwb status

*If configuration error is 'Present' please run 'show ap urwb err-status' for
 details

AP name          Coordinator name Config status Config error RF tag
   Radio roles
-----------------------------------------------------------------------------------------

AP345D.A812.0B6C AP345D.A812.0B6C Up to date                cm_p2mp_rf_tag_me
  PTMP Base
APC828.E536.7920 AP345D.A812.0B6C Up to date               cm_p2mp_rf_tag_mp2
 PTMP Secondary
APE4A4.1CD8.1700 AP345D.A812.0B6C Up to date               cm_p2mp_rf_tag_mp1
 PTMP Secondary
```

Controller-directed URWB traffic

Introduces controller-directed URWB traffic, enabling communication across broadcast domains and infrastructure subnets, centralizing traffic forwarding, preserving seamless mobility, and maintaining vehicle location and VLAN context for Layer 3 mobility.

This feature enables communication between different broadcast domains and infrastructure subnets.

It allows URWB vehicles and their wired clients to communicate across multiple infrastructure subnets and reduces broadcast domains while preserving seamless mobility.

In controller-directed mode, the controller maintains awareness of vehicle location, client association, and VLAN context as vehicles move. This enables Layer 3 mobility across infrastructure segments while keeping forwarding decisions centralized.

URWB clients

Introduces URWB clients as controller-managed entities representing vehicle APs and wired endpoints behind them, and explains how they enable consistent mobility, policy enforcement, and VLAN handling as vehicles move across network segments.

A URWB client is a controller-managed client type that represents either:

- a vehicle AP participating in the URWB network, or
- a wired endpoint connected behind a vehicle AP.

On the controller, URWB clients are tracked similarly to wireless clients for mobility, policy, and forwarding purposes. This allows the controller to apply consistent VLAN, policy, and roaming behavior as the vehicle moves between segments.

URWB clients

- Vehicle AP: A mobile AP mounted on a vehicle and joined to the URWB network.
- Wired URWB client: A camera, sensor, or other Ethernet-connected device behind the vehicle AP.

URWB traffic egress point

Outlines the configuration of the URWB traffic egress point, including profile setup, WLAN and policy profile creation, and policy tag mapping for proper mobility, policy handling, and traffic forwarding.

To enable the Controller-directed URWB Traffic feature, the URWB profile must be configured with the appropriate traffic egress point. When the traffic egress point is set to controller, it automatically generates the "cisco-urwb-client "WLAN and the corresponding "cisco-urwb-client" policy profile.

The policy tag must then be configured correctly to map the "cisco-urwb-client" WLAN to the "cisco-urwb-client" policy profile. This mapping is required for URWB client mobility, policy handling, and traffic forwarding to work as expected.

URWB client WLANs

Introduces system-generated URWB client WLANs, explaining their automatic creation, controller-based mobility and policy handling, and the use of the associated policy profile for mapping policies and forwarding contexts.

A URWB client WLAN (cisco-urwb-client) is a system-generated controller construct used to manage URWB clients. The system creates this WLAN automatically when a user changes the traffic egress point in a URWB profile to controller.

It supports controller-based mobility and policy handling for URWB clients and binds URWB clients to the required policy and forwarding context.

In addition to the WLAN, the system also creates a policy profile named "cisco-urwb-client" automatically. Use this policy profile to create the required mapping between the policy profile and the WLAN.

URWB mobility across subnets

Introduces URWB mobility across subnets, explaining how vehicle APs and wired devices maintain connectivity during intra-segment and inter-segment handoffs, with controller-managed forwarding updates ensuring uninterrupted client access as the AP roams between infrastructure segments.

URWB mobility across subnets allows a vehicle AP and the wired devices behind it to remain reachable while moving between infrastructure segments.

Mobility cases:

- **Intra-segment handoff**: Roaming within the same subnet, with traffic continuing through the same segment.
- **Inter-segment handoff**: Roaming across different subnets, requiring controller-directed traffic handling and forwarding updates.

When inter-segment handoff occurs, the controller updates the forwarding path based on the vehicle AP's new location while preserving client connectivity.

Limitations of Controller-directed URWB traffic

Outlines the limitations of controller-directed URWB traffic in release 26.2.1, including controller, mode, high availability, IPv6, IGMP snooping requirements, and lack of wireless relay support.

This feature has the following limitations in release 26.2.1:

- Only a single controller is supported.
- Only local mode is supported, FlexConnect is not supported.
- Controller high availability for controllers are not supported.
- IPv6 for the management network is not supported.
- IGMP snooping must be disabled on VLANs used by mobile APs and URWB wired clients. You can use the **no ip igmp snooping vlan** *vlan* command for this.
- Wireless relay is not supported.

How URWB client mobility works

Explains how URWB client mobility enables vehicle APs and their wired clients to stay connected while moving across network segments, describing the roles of the vehicle AP, coordinator AP, and controller, and outlining the stages that maintain uninterrupted connectivity.

URWB client mobility allows vehicle APs and their downstream wired clients to maintain connectivity while moving between network segments.

The key components involved in the process are:

- **Vehicle AP**: The mobile AP that moves between infrastructure subnets.
- **Coordinator AP**: The AP that aggregates URWB traffic within a local segment.
- **Wireless LAN Controller**: The device that tracks client location and updates forwarding behavior.

The process involves the following stages:

1. The vehicle AP connects through a coordinator AP and is identified by the controller as a URWB client.
2. The controller tracks the vehicle AP location, client state, and VLAN context.
3. When the vehicle AP roams to a new segment, the controller updates the forwarding path.
4. Traffic for the vehicle AP and wired clients continues without manual reconfiguration.

Result: Clients remain connected and reachable as vehicles move across subnets.

Create VLAN for vehicle AP and wired client (GUI)

This task creates the necessary VLANs on the controller to segment and manage traffic for URWB clients and mobile APs, ensuring proper routing and mobility support within the network.

Creating dedicated VLANs ensures that traffic from vehicle-based wired clients (such as VLAN 110) and mobile AP management traffic (such as VLAN 111) are correctly identified, segmented, and forwarded by the controller.

Procedure

1. Choose **Configuration** > **Layer2** > **VLAN**.
2. Click the **VLAN** tab and click the **Add** button.

The **Create VLAN** page is displayed.

3. Select the **Create a single VLAN** option and enter these parameters:
   a) **VLAN ID**: Create separate VLAN IDs for vehicle AP and wired clients. For example, enter 111 as the VLAN ID for vehicle AP and 110 for the wired client.
   b) **Name**: Enter appropriate names for the VLAN. Ensure that the length of the VLAN name is less than 20 characters.

   > **Note**
   >
   > Disable VTPv1 and VTPv2 or switch to VTPv3 before configuring a VLAN name longer than 32 characters.

4. Click **Apply to Device**.

The VLANs are created on the controller, segmenting traffic for vehicle APs and wired clients and enabling proper routing and mobility support.

Create VLAN for vehicle AP and wired client

This task creates the necessary VLANs on the controller to segment and manage traffic for URWB clients and mobile APs, ensuring proper routing and mobility support within the network.

Establish the necessary network segments on the controller to support URWB client data and mobile AP management traffic.

When implementing controller-directed URWB traffic, the controller must be aware of the specific subnets used by mobile devices. Creating dedicated VLANs ensures that traffic from vehicle-based wired clients (such as VLAN 110) and mobile AP management traffic (such as VLAN 111) are correctly identified, segmented, and forwarded by the controller.

Procedure

---

1. Use the **configure terminal** command to enter configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Use the **vlan** *vlan-id* command to create VLAN for the vehicle AP.

   Example

   ```
   Device(config)# vlan 111
   ```

3. Use the **vlan** *vlan-id* command to create VLAN for wired clients behind the vehicle AP.

   Example

   ```
   Device(config)# vlan 110
   ```

4. Use the **end** command to return to the privileged EXEC mode.

Example

```
Device(config)# end
```

Create URWB profile for mobility role base and client (GUI)

This task configures the URWB network profiles to establish the controller as the central traffic egress point and defines the necessary mobility roles for mobile clients.

Set up URWB network profiles to ensure centralized traffic management and assign correct mobility roles for mobile clients.

Use this task when you want to establish the controller as the central traffic egress point for mobile clients by defining appropriate URWB profiles and mobility roles.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **URWB Network Profile**.
2. In the **General** tab, enable the **URWB Status** and enter the **URWB Network Profile Name**.
3. Click the **Mobility** tab, and complete these steps:
   a) From the **Mobility Role** drop-down list, select either **Base**, **Base Relay**, or **Client**.

   > **Note**
   >
   > When you select **Base** or **Base Relay** as the mobility role, the **Wired Ports** tab is displayed. When you select **Client** as the mobility role, the **Wired Clients** tab is displayed.

   b) From the **Traffic Egress Point** drop-down list select one of these egress points:
      1. **Coordinator**: Specifies coordinator as the URWB traffic egress point.
      2. **WLC**: Specifies wireless LAN controller as the URWB traffic egress point.

   c) In the **Fast Drop Count** field, configure the fast wireless disconnect feature by entering the drop count. The range is 0 to 65535. The default value is 0.
4. In the **Advanced** section, under the **Mobility Backhaul Check** area, complete these steps:
   a. From the **Coordinator Down** drop-down list, select either **Disabled**, **Handoff-Inhibition**, or **Relay-Switch**, to configure hand-off when the coordinator is unreachable.
   b. From the **Ethernet Down** drop-down list, select either **Disabled**, **Handoff-Inhibition**, or **Relay-Switch**, to configure hand-off when all the Ethernet ports are disconnected.
5. In the **Mobility Scan** section, configure the scan parameters:
   a. **Isolation**: Scanning takes place when the device is disconnected from the infrastructure. The maximum waiting time before the scanning is triggered is 0 to 65535 ms. The default value is 0.
   b. **Periodic**: Scanning takes place periodically. The value range is 0 to 65535 seconds. The default value is 0.

c. **RSSI Threshold**: Scanning takes place when the RSSI value falls below threshold. The value range is 0 to 96. The default value is 0.

6. Click **Apply to Device**.

The URWB network profiles are configured, and the controller functions as the central traffic egress point. Mobility roles for mobile clients are defined, ensuring proper traffic routing and client management.

Configure wired ports (GUI)

Configure which physical Ethernet interface on the URWB device operates as the Secondary LAN Port. This provides an additional wired connection for the network profile.

When you select **Base** or **Base Relay** as the mobility role, the **Wired Ports** tab is displayed. Under Ethernet settings, the secondary port can be disabled or assigned to LAN 1, LAN 2, LAN 3, or LAN 4. Changes may require configuration activation or a reload of the associated AP(s).

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **URWB Network Profile**.
2. Select the required URWB profile from the displayed list.
3. Click the **Wired Ports** tab
4. In the **Ethernet Settings** section, complete these steps:
   a) From the **Secondary LAN Port** drop-down list, select a secondary LAN port to enable LAN port PoE.
   b) Check the **LAN Port PoE** check box to enable the LAN port PoE.

   > **Note**
   >
   > The Secondary LAN Port and LAN Port PoE configuration has no effect on unsupported APs. This feature is currently supported on Cisco Catalyst 9124AX series APs.

5. Click **Apply to Device** to save the configuration.

The selected LAN interface is configured as the secondary LAN port and applied to the device, potentially requiring configuration activation or an AP reload.

Configure wired clients (GUI)

Configure devices directly connected to URWB access point LAN ports, including cameras, sensors, and industrial controllers.

Enable and manage connectivity, authentication, and NAT configuration for wired devices connected to URWB access points.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **URWB Network Profile**.
2. Select the required URWB profile from the displayed list.
3. Click the **Wired Ports** tab

4. In the **Ethernet Settings** section, complete these steps:
   a) From the **Secondary LAN Port** drop-down list, select a secondary LAN port to enable LAN port PoE.
   b) Check the **LAN Port PoE** check box to enable the LAN port PoE. The Secondary LAN Port and LAN Port PoE configuration has no effect on unsupported APs.
5. In the **Local Authentication** section, select the RADIUS server group used to authenticate wired clients. The blue shortcut icon opens the RADIUS configuration area if a group must be created or reviewed.

   The selected RADIUS servers must be reachable from the network and already contain the appropriate client credentials or policies.
6. In the **NAT** section, check the **Enable NAT** check box to enable the NAT feature.
7. In **NAT Address**, enter the IPv4 address of the AP facing the on-board local network and then enter the subnet mask associated with that address.
8. Enter the **Outside Port Range**. The NAT outside port range is from 1 to 65535 unless both are 0.
9. In the **NAT Rules** section, click **Add** and complete these steps to add new rules:
   a) Select the required protocol, normally TCP or UDP.
   b) Enter the outside port on which the NAT address receives the connection. The range for NAT rule outside port is from 1 to 65535.
   c) Enter the IPv4 address of the wired device as the inside IPv4 address.
   d) Enter the inside port on that wired device.
10. Click **Save**. The newly added rule is displayed in a table.

Wired devices are configured and connected, authenticated (if required), and NAT rules are applied. The configuration is saved and displayed in the profile.

Create URWB profile for mobility role base and client

This task configures the URWB network profiles to establish the controller as the central traffic egress point and defines the necessary mobility roles for mobile clients.

Configure URWB network profiles so that the controller is the traffic egress point for both the base role and the client role.

This configuration is required to shift traffic handling from the local coordinator to the controller, enabling centralized traffic management for mobile subnets. Both the mobility role base and client need to configure the egress point to controller.

Before you begin

- In controller-directed deployments, both profiles must use the traffic egress point `WLC`.
- The base profile must use mobility role `Base` and the client profile must use mobility role `Client`

Procedure

---

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Use the **wireless profile urwb** *profile* command to create or edit the base URWB profile.

Example

```
Device(config)# wireless profile urwb mb-base-urwb-profile
```

3. Use the **mobility traffic-egress wlc** command to set the controller as the traffic egress point.

   Example

   ```
   Device(config-wireless-urwb-profile)# mobility traffic-egress wlc
   ```

4. Use the **exit** command to return to the global configuration mode.

   Example

   ```
   Device(config-wireless-urwb-profile)# exit
   ```

5. Use the **wireless profile urwb** *profile* command to create or edit the client URWB profile.

   Example

   ```
   Device(config)# wireless profile urwb mb-client-urwb-profile
   ```

6. Use the **mobility traffic-egress wlc** command to set the controller as the traffic egress point for the client.

   Example

   ```
   Device(config-wireless-urwb-profile)# mobility traffic-egress wlc
   ```

7. Use the **mobility role client** command to set the mobility role for the client profile.

   Example

   ```
   Device(config-wireless-urwb-profile)# mobility role client
   ```

8. Use the **end** command to return to the privileged EXEC mode.

   Example

   ```
   Device(config-wireless-urwb-profile)# end
   ```

9. (Optional) Use the **show wireless profile urwb detailed** *profile* command to verify the configuration and operating status.

   Example

   ```
   Device# show wireless profile urwb detailed mb-base-urwb-profile

   Profile Name                        : mb-base-urwb-profile
   Description                         :
   Status                              : Disabled
   Multicast                           : Disabled
   RADIUS server group name            :

   Mobility
   ```

```
  Role                                : Base
  Traffic Egress Point                : WLC
  Warm up time                        : 30000
  Timeout for candidate               : 800
  High threshold value                : 6
  Low threshold value                 : 3
  Low/High threshold value            : 35
  Advertise signal redundancy         : 1
  Backhaul check ethernet             : Disabled
  Backhaul check Coordinator          : Disabled
  Scanning
    Periodic (s)                      : Disabled
    Isolation (ms)                    : Disabled
    RSSI threshold                    : Disabled
  Fast Drop Count                     : 0

MPLS
  High Availability                   : Disabled
  High Availability Timeout           : 100
  Unicast flood                       : Disabled
  Unicast flood limitation            : Disabled
  Allow ethernet 1                    : Disabled
  Ethernet Filter Method              : Not Configured
  Ethernet Types List                 : Not Configured

MPO
  Status                              : Disabled
  Maximum links                       : 2
  Minimum RSSI to establish           : 20
  CoS                                 : 6
  Telemetry status                    : Disabled

Secondary Ethernet
  Port state                        : Disabled
  PoE out                           : Disabled

Reliability
  Gratuitous ARP Status               : Disabled
  QOS Mapping for Priority 0          : 0
  QOS Mapping for Priority 1          : 1
  QOS Mapping for Priority 2          : 2
  QOS Mapping for Priority 3          : 3
  QOS Mapping for Priority 4          : 4
  QOS Mapping for Priority 5          : 5
  QOS Mapping for Priority 6          : 6
  QOS Mapping for Priority 7          : 7

NAT
  Status                              : Disabled
```

Create policy tag profile (GUI)

This task creates and configures a policy tag to associate the default URWB WLAN and policy profile with the coordinator APs through the WebUI.

Associate the automatically generated URWB WLAN and policy profile with a policy tag.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
   The **Policy Tag** page is displayed.
2. Click **Add**.

   The **Add Policy Tag** page is displayed.
3. Enter the policy tag name and the description.
4. In the **WLAN-POLICY Maps** section, click **Add**.

   The **Map WLAN and Policy** section is displayed.
5. From the **WLAN Profile** and **Policy Profile** drop-down lists, map the WLAN and policy profile names, and click the tick-mark button.
6. Click **Apply to Device**.

The policy tag profile is successfully created and associated with the default URWB WLAN and policy profile, enabling efficient management of coordinator APs.

Create policy tag profile

This task creates and configures a policy tag to associate the default URWB WLAN and policy profile with the coordinator APs.

Associate the automatically generated URWB WLAN and policy profile with a policy tag.

After enabling the controller-directed URWB feature, the system generates a default WLAN and policy profile (cisco_urwb_client). You cannot modify these parameters. These must be mapped to a policy tag so they can be assigned to the coordinator APs.

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Use the **wireless tag policy** *policy-tag* command to create or edit the policy tag.

   Example

   ```
   Device(config)# wireless tag policy urwb-base-policy-tag
   ```

   When the feature is enabled in the URWB profile, the system automatically creates a policy profile (cisco_urwb_client) and a WLAN profile (cisco_urwb_client).
3. Use the **wlan** *wlan-profile* **policy** *policy-profile* command to map the system-created URWB WLAN and policy profile to the policy tag.

   Example

   ```
   Device(config-policy-tag)# wlan cisco-urwb-client policy cisco-urwb-client
   ```

4. Use the **end** command to return to privileged EXEC mode.

Example

```
Device(config-policy-tag)# end
```

Bind policy tag to coordinator APs (GUI)

This task binds the created URWB policy tag to the coordinator APs to apply the necessary WLAN and policy configurations using GUI.

Assign the configured URWB policy tag to the coordinator APs to activate the URWB feature on those devices

After creating the policy tag containing the required URWB WLAN and policy profile, you must bind it to the specific coordinator APs to ensure they apply the correct configuration settings.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click an AP from the list of displayed APs.
   The **Edit AP** page is displayed.
3. Click the **General** tab.
4. In the **General** section, enter the AP name and specify location at which the AP is present.
5. In the **Tags** section, select the appropriate **Policy**, **Site**, and **RF** tags that you created in the **Configuration** > **Tags & Profiles** > **Tags** page.
6. Click **Update & Apply to Device**.

The selected coordinator APs now apply the URWB policy tag and activate their WLAN and policy configurations.

Bind the policy tag to coordinator APs

This task binds the created URWB policy tag to the coordinator APs to apply the necessary WLAN and policy configurations.

Assign the URWB policy tag to the coordinator APs so they apply the required URWB WLAN and policy configuration.

After creating the policy tag containing the required URWB WLAN and policy profile, you must bind it to the specific coordinator APs to ensure they apply the correct configuration settings.

Procedure

1. Use the **configure terminal** command to enter configuration mode.

Example

```
Device# configure terminal
```

2. Use the **ap** *ap-mac* command to enter AP configuration mode for the target coordinator AP.

Example

```
Device(config)# ap 0011.2233.4455
```

3. Use the **policy-tag** *urwb-policy-tag* command to apply the policy tag to the AP.

Example

```
Device(config-ap-tag)# policy-tag urwb-base-policy-tag
```

4. Use the **end** command to return to privileged EXEC mode.

Example

```
Device(config-ap-tag)# end
```

Configure VLAN tagging for vehicle AP (GUI)

Configure VLAN tagging on a vehicle access point to enable segmented network traffic for different applications.

Enable and assign a VLAN tag to a vehicle access point (AP) so that it can support separate network segments for improved traffic management.

Use this task when you need to assign VLAN tags to vehicle access points to support network segmentation, which can be necessary for isolating traffic types.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Click an AP from the list of displayed APs.
   The **Edit AP** page is displayed.
3. Click the **Advanced** tab.
4. In the **VLAN Tag** section, check the **VLAN Tag** check box to enable the VLAN tag.

> **Note**
>
> When changing the VLAN tag, there is a momentary loss of association of the AP with the controller.

   The **VLAN Tag ID** is displayed.
5. In the **VLAN Tag ID** field, enter the VLAN tag ID. The range is from 1 to 4094.
6. Click **Update & Apply to Device**.

The vehicle AP is updated with the new VLAN tag.

Configure VLAN tagging for the vehicle AP

This task assigns a specific VLAN tag to a vehicle AP to ensure it operates on the correct subnet for controller-directed traffic.

Enable VLAN tagging on the vehicle AP so it joins the correct infrastructure VLAN for mobility and traffic handling.

In controller-directed URWB traffic scenarios, vehicle APs operate on different subnets and VLANs than infrastructure coordinators. You must enable VLAN tagging for these APs to maintain correct network segmentation and roaming capabilities.

The configured tag VLAN must not be the same as the wireless management VLAN configured on the controller.

Procedure

Use the **ap name** *ap_name* **vlan-tag** *vlan-number* command to assign the specific VLAN tag to the vehicle AP.

Example

```
Device# ap name Vehicle-AP-01 vlan-tag 111
```

View the client type information

This section provides diagnostic commands to verify URWB client identification, tunnel status, and hardware/software processing states on the controller.

Use these commands to verify that the controller has correctly identified and programmed the URWB client.

Procedure

1. Use the **show wireless client mac-address** *mac* **detail** command to confirm that the device is correctly identified as a URWB client. You can view the general client details, including the client type and authentication state.

   Example

   ```
   Device# show wireless client mac-address 5x13.d5f6.264a detail | sec Type

   Client MAC Type              : Universally Administered Address
   Client Type                  : URWB Client
   Mobility Roam Type           : None
   Policy Type                  : N/A
   EAP Type                     : Not Applicable
   ```

2. Use the **show platform hardware chassis active qfp feature wireless wlclient cpp-client mac** *mac* **details** command to verify the data path tunnel type and confirm that the client is correctly processed by the hardware.

   Example

   ```
   Device# show platform hardware chassis active qfp feature wireless wlclient
   cpp-client mac 5x13.d5f6.264a details | sec Type

     POA Tunnel Type     : DATA
     Client Type         : URWB Client
     P2P Type            : BLOCKING DISABLE
   ```

3. Use the **show platform software wireless-client chassis active f0 mac-address** *mac* command to view the global WLAN ID, SSID, and mobility state for a specific URWB client.

   Example

```
Device# show platform software wireless-client chassis active f0 mac-address
 5x13.d5f6.264a

ID                            : 0xa0000003
MAC address                   : 5x13.x5x6.264a
Type                          : URWB Client
Global WLAN ID                : 5
SSID                          : cisco_urwb_client
Client index                  : 0
Mobility state                : Local
Authentication state          : Run
```

Support for 6 GHz Low Power Indoor mode

Supports the operation of Catalyst IW916x access points in 6 GHz Low Power Indoor (LPI) mode for enhanced wireless performance and regulatory compliance in indoor environments.

LPI mode is a specific operating class within the 6 GHz band. In this mode, access points are restricted to lower power levels and are intended exclusively for indoor use.

From Release 26.2.1, these platforms support 6-GHz LPI mode in Korea domain:

- IW9167EH - LPI AP
- IW9167I - LPI AP
- IW9165D - LPI AP
- IW9165E - LPI AP and client

Introduction of 6 GHz LPI mode

Starting with Cisco IOS XE Release 17.18.1, this feature allows the APs to operate exclusively in the 6 GHz frequency band in LPI mode as a mobility client.

In this configuration, the AP serves as a mobility client, connecting to a mobility base role on the infrastructure side. You must set up the URWB profile and radio profile for both the infrastructure device and the client device.

An infrastructure device refers to a controller that provides centralized management, routing, and coordination for the network.

A client device refers to an AP that connects to the network through an infrastructure device.

Default band operation

By default, the APs operates in the 5 GHz band but can be configured for 6 GHz LPI operations.

6GHz LPI on radio2 for URWB uplink

The 6 GHz LPI radio operates on Slot 2 as a dual-band XOR Radio. Activating the 6 GHz band of Radio 2 allows for URWB operation and enhances the performance and capabilities of the AP.

URWB role for LPI mode

To enable the 6 GHz LPI mode, the AP must be assigned the URWB role via the radio profile. Once set to the URWB role, the URWB stack manages the radio by disabling all Radio Resource Management (RRM) functions and automatically selecting the LPI mode.

Supported sub-channels and frequency range

In 6 GHz LPI mode, the AP supports the U-NII-5, U-NII-6, U-NII-7, and U-NII-8 sub-channels. These sub-channels cover a frequency range of 5925 MHz to 7125 MHz, aligning with the Wi-Fi 6E band.

Restrictions

Outlines operational limitations and compliance requirements for APs operating as 6 GHz mobility clients.

When operating as a 6 GHz mobility client, the AP is subject to specific restrictions to ensure compliance with regulatory requirements and operational guidelines:

- Supports only LPI mode as a 6 GHz AP.
- URWB mobility clients do not support manual configuration of 6 GHz TX power, or RRM and 6 GHz SP mode.

How 6 GHz mobility clients operate

Describes how Cisco Catalyst APs operate as 6 GHz mobility clients to establish efficient, interference-free wireless connections, ensuring compliance with spectrum regulations.

The AP functions as a 6 GHz mobility client. It uses advanced wireless protocols and scanning mechanisms to establish interference-free and efficient connections with parent APs. This process ensures both seamless operation and adherence to spectrum regulations.

The key components involved in this process are:

- Cisco Catalyst APs: Operates as the 6 GHz mobility client, scanning channels and connecting to a parent AP.
- Carrier-Sense Multiple Access with Collision Avoidance (CSMA/CA): This protocol prevents interference by detecting channel availability before the device transmits.
- Listen Before Talk (LBT): Ensures compliance by avoiding interference with licensed services in the spectrum.
- Parent AP: Acts as the central access point to which the AP connects for continued operation.
- Received Signal Strength Indicator (RSSI): Evaluated by the AP to determine signal quality for connection.
- URWB network key: Used by the AP to authenticate and select the optimal parent AP.

These stages describe how 6 GHz mobility clients operate:

1. Scanning channels: The AP scans dedicated 6 GHz channels, listens for beacons, and evaluates the RSSI and URWB network key.
2. Protocol compliance: The device uses CSMA/CA and LBT protocols to detect and avoid interference with nearby licensed services.
3. Parent AP selection: The AP evaluates the RSSI and other parameters to select the optimal parent AP for connection.

> **Note**
>
> When the parent AP is in LPI mode, the maximum EIRP is limited to 24 dBm.

4. Connection establishment: The AP refrains from transmitting any packets until a connection with the parent AP is successfully established.
5. Transition to legacy mode: After connecting to the parent AP, the device transitions to legacy mode and continues operation under the parent AP.

Result

The process enables the AP to operate efficiently and interference-free as a 6 GHz mobility client, ensuring reliable connections with parent APs and adherence to spectrum regulations.

Configure the 6-GHz LPI mode

The 6-GHz LPI mode enables access points to provide secure, low-power wireless connectivity in indoor environments and supports advanced radio profiles and tag assignments.

Perform these steps on the wireless controller to configure the 6-GHz LPI mode.

Before you begin

Ensure that the AP is connected to the controller through a wired link in the provision mode.

Procedure

1. Configure the URWB network profile on the infrastructure device and on the client device.
2. Configure the radio profile on the infrastructure device and on the client device..
3. Create an RF tag and assign radio profile and URWB profiles on the infrastructure device and on the client device.
4. Apply the RF tag of the infrastructure device and the client device to the access point.
5. (Optional) Verify 6-GHz LPI mode on the AP in mobility client role.

Configure the URWB radio profile

Procedure

Create a radio profile which specifies URWB 6-GHz channel and URWB channel width and URWB role. Ensur that you use the same URWB profile id that you create here, when attaching the wireless RF tag.

For CLI configuration steps, refer to Create URWB radio profile (CLI) and for GUI configuration stepsrefer to Create URWB radio profile (GUI).

Configure URWB network profile

Complete the following steps to configure the URWB network profile:

Before you begin

Ensure that you use the same URWB profile id that you create here, when attaching the wireless RF tag.

Procedure

1. Enter configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter a profile name to create a URWB profile.

   Example

   ```
   Device(config)# wireless profile urwb urwb-profile-name
   ```

   This command creates a URWB profile and enters into the wireless URWB profile configuration mode.

3. Configure a network key.

   Example

   ```
   Device(config-wireless-urwb-profile)# network-key network-key-type {0 |
   8}<network-key>
   ```

   This command configures the shared secret for secure communication between the URWB nodes. The network key can be saved as encrypted or unencrypted type.

   Here, **0** indicates that you must enter an unencrypted network key; and **8** indicates that you must enter an AES encrypted network key.

   > **Note**
   >
   > The network key must not contain special characters.

4. Configure URWB mobility role to base for the AP.

   Example

   ```
   Device(config-wireless-urwb-profile)# mobility role base
   ```

5. Enable the URWB profile.

   Example

   ```
   Device(config-wireless-urwb-profile)# no shutdown
   ```

   This command enables the URWB profile in the wireless URWB profile configuration mode.

   To disable the URWB profile, use the **shutdown** command.

6. Optional: Configure a strong password evaluation for the network key.

Example

```
Device(config-wireless-urwb-profile)# strong-network-key
```

Configure the radio profile for infrastructure device

Use this procedure to configure the radio profile for infrastructure device.

Before you begin

Ensure that you use the same radio profile name that you create here, when attaching the wireless RF tag.

Procedure

---

1. Use the **wireless profile radio** *radio-profile-name* command to create the radio profile for the infrastructure device.

Example

```
Device(config)#wireless profile radio 6g-radio-infra
```

2. Use the **urwb role** *mobility* command to configure the URWB radio role.

Example

```
Device(config-wireless-radio-profile)#urwb role mobility
```

3. Use the **urwb channel** *radio-band channel-number* command to configure the 6 GHz radio band and the channel number.

Example

```
Device(config-wireless-radio-profile)#urwb channel 6ghz 145
```

4. Use the **urwb cwidth** *channel-width* command to configure the channel width.

Example

```
Device(config-wireless-radio-profile)#urwb cwidth 160MHz
```

5. Use the **urwb crypto policy** *policy-name* command to configure the encryption.

Example

```
Device(config-wireless-radio-profile)#urwb crypto policy none
```

6. Use the **exit** *policy-name* command to return to global configuration.

Example

```
Device(config-wireless-radio-profile)#exit
```

Configure the radio profile for the client device

Use this procedure to configure the radio profile for client device.

Before you begin

Ensure that you use the same radio profile name that you create here, when attaching the wireless RF tag.

Procedure

---

1. Use the **wireless profile radio** *radio-profile-name* command to create the radio profile for the client device.

   Example

   ```
   Device(config)#wireless profile radio 6g-radio-vehicle
   ```

2. Use the **urwb role** *mobility* command to configure the URWB radio role.

   Example

   ```
   Device(config-wireless-radio-profile)#urwb role mobility
   ```

3. Use the **urwb channel** *radio-band channel-number* command to configure the 6 GHz radio band and the channel number.

   Example

   ```
   Device(config-wireless-radio-profile)#urwb channel 6ghz 145
   ```

4. Use the **urwb cwidth** *channel-width* command to configure the channel width.

   Example

   ```
   Device(config-wireless-radio-profile)#urwb cwidth 160MHz
   ```

5. Use the **urwb crypto policy** *policy-name* command to configure the encryption.

   Example

   ```
   Device(config-wireless-radio-profile)#urwb crypto policy none
   ```

6. Use the **exit** *policy-name* command to return to global configuration.

Example

```
Device(config-wireless-radio-profile)#exit
```

Create the RF tag and assign radio profile and URWB profiles for the infrastructure device

Use this procedure to create the RF tag and attach radio profile and URWB profiles for the infrastructure device.

Procedure

1. Use the **wireless tag rf** *rf-profile-name* command to create an RF tag for the infrastructure device.

   Example

   ```
   Device(config)#wireless tag rf 6g-infra
   ```

2. Use the **dot11 6ghz slot2 radio-profile** *radio-profile-name* command to attach radio profile to a 6 GHz radio configuration.

   Example

   ```
   Device(config-wireless-rf-tag)#dot11 6ghz slot2 radio-profile 6g-radio-infra
   ```

3. Use the **urwb-profile***urwb-network-profile-name* command to attach URWB profile to a 6 GHz radio configuration.

   Example

   ```
   Device(config-wireless-rf-tag)#urwb-profile mobility-VE-infra
   ```

4. Use the **exit** command to return to global configuration.

   Example

   ```
   Device(config-wireless-urwb-profile)#exit
   ```

Create an RF tag and assign radio profile and URWB profiles for the client device (CLI)

Complete the following steps to create the RF tag and attach radio profile and URWB profiles for the client device.

Procedure

For CLI configuration steps, refer to Create RF tags and assign profiles (CLI) on page 2470.

Apply RF tag of the infrastructure device to access point

Use this procedure to apply the RF tag and attach radio profile and URWB profiles of the client device to access point.

Procedure

1. Use the **ap** *mac-address* command to configure the MAC address of the infrastructure device.

   Example

   ```
   Device(config)#ap fc58.9a16.e570
   ```

2. Use the **rf-tag** *rf-tag-id* command to apply RF tag of the infrastructure device to access point.

   Example

   ```
   Device(config)#rf-tag 6g-infra
   ```

Apply RF tag of the client device to access point

Use this procedure to apply the RF tag and attach radio profile and URWB profiles of the client device to access point.

Procedure

1. Use the **ap** *mac-address* command to configure the MAC address of the client device.

   Example

   ```
   Device(config)#ap  fc58.9a17.xxxx
   ```

2. Use the **rf-tag** *rf-tag-id* command to apply RF tag of the client device to access point.

   Example

   ```
   Device(config)#rf-tag 6g-vehicle
   ```

Verify 6 GHz Low Power Indoor on AP in mobility client role

Verify the 6 GHz Low Power Indoor (LPI) mode on an access point operating as a mobility client.

Use this procedure to verify 6 GHz LPI on AP in mobility client role.

Procedure

Use the **show urwb dot11Radio 2 config** command to verify the 6 GHz LPI mode on AP in mobility client role.

Example

```
Device# show urwb dot11Radio 2 config
Interface:              enabled

Mode:                   fluidity
```

```
Frequency:              6675 MHz
Channel:                145
Channel width:          160 MHz
Antenna number:         2
TX power level:         4
TX power:               11 dBm
Antenna gain:           6 dBi
Maximum tx mcs:         9
High-efficiency:        enabled
Maximum tx nss:         2
RTS protection:         disabled
guard-interval:         800 ns
ampdu max length:       255
distance:               3000 m

The ampdu Tx
priority 0:             enabled
priority 1:             enabled
priority 2:             enabled
priority 3:             enabled
priority 4:             enabled
priority 5:             enabled
priority 6:             disabled
priority 7:             disabled

Enhanced Distributed Channel Access (EDCA) configuration
vo: aifs=1 cw_min=2 cw_max=3 txop=15
vi: aifs=1 cw_min=3 cw_max=4 txop=31
be: aifs=3 cw_min=4 cw_max=6 txop=31
bk: aifs=7 cw_min=3 cw_max=4 txop=0

Passphrase:
80fbe236dae4d38893bdde3359e625256ac5d05f91ac68537a02c3f2ef20e032
AES encryption:         disabled
AES key-control:        disabled
Key rotation:           disabled
Key rotation timeout:   0(second)

DFS region:             B
DFS radar role:         auto
Radar detected:         0
Indoor deployment:      enable
Rx-SOP Threshold:       0 dBm(AUTO)
Max packet retries:     32
High throughput 4.9Ghz: disabled
```

NAT/PAT for URWB

Outlines NAT/PAT functionality for URWB networks, including support for AGV mobility, IP and port mapping to WAN addresses, traffic flow management, policy-based configuration, and port address translation for accessing multiple services via single external IPs.

Network address translation for URWB is a networking feature that:

- provides NAT/PAT functionality for AGV use cases when the URWB mobility feature is enabled,
- allows network administrators to map local IP addresses and ports to wide area network (WAN) addresses, and

- manages traffic flow between on-board local networks and external systems.

Key features of NAT/PAT for URWB

Policy-based management: NAT/PAT configuration is integrated into the URWB network profile, ensuring consistent application across devices.

Port address translation: Enables multiple internal services to be reached via a single external IP address using different port mappings.

Guidelines for NAT/PAT configuration

Outlines key guidelines for effective NAT/PAT configuration, including avoiding port overlap with reserved system ports, adhering to rule limits per device, and monitoring translation table size to ensure stable performance.

Follow these guidelines to ensure stable NAT/PAT performance:

- Avoid port overlap: When defining the NAT outside port range, ensure that the selected range does not overlap with reserved system ports. The following ports are strictly reserved for internal system operations and must be excluded from your configuration: 651, 5246, 5247, 6600, and 6610. Configuring a range that includes these ports will result in a validation error.
- Observe rule limits: You can configure a maximum of 100 rules per device.
- Monitor table size: To prevent resource exhaustion, the NAT translation table is limited to 4096 entries.

Configure NAT/PAT for URWB (GUI)

Provides instructions for configuring NAT and PAT on URWB systems using the graphical user interface, including step-by-step guidance for setup and management.

Enable and configure NAT/PAT translation rules for URWB mobile units.

Use this task to allow communication between devices on an on-board local network and the WAN. This configuration is managed within the URWB network profile.

Before you begin

- Ensure the URWB mobility feature is enabled on the access point.
- Identify the IP address and netmask for the local network.

Follow these steps to configure NAT/PAT for URWB:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **URWB Network Profile**.
2. Click **Add** or select an existing profile.
3. In the **General** tab, enable the URWB status to edit the profile.
4. In the **Mobility** tab, from the **Mobility Role** drop-down list, choose **Client**.
    - **Base**
    - **Base Relay**
    - **Client**

    When you select **Base** or **Base Relay** as the mobility role, the **Wired Ports** tab is displayed. When you select **Client** as the mobility role, the **Wired Clients** tab is displayed.

5. In the **Wired Clients** tab, under the **NAT** section, check the **NAT** check box to enable Network and Port Address Translation (NAT/PAT) feature.
6. In **NAT Address**, enter the IPv4 address of the AP facing the on-board local network and enter the subnet mask associated with the NAT address.
7. Enter the **Outside Port Range**. Specify the minimum and maximum port range for outside port rule value. The range is from 1 to 65535 unless both are 0.
8. In the NAT rules section, click the **Add Rule** button to add a NAT/PAT configuration rule that contains protocol information, **Inside IP Address**, **Inside Port**, and **Outside Port**. The range for the port numbers is from 1 to 65535.

   Note

   You can configure up to 100 rules. The target host shall be located within the configured inside network address space.

9. Click **Save** to add the new rules to the table and then click **Update & Apply to Device**.

NAT/PAT translation rules are applied, enabling devices on the local on-board network to communicate with external networks through the WAN.

Configure NAT/PAT for URWB

Guides users through command-line procedures for configuring NAT and PAT in URWB environments, covering essential steps for implementation and verification.

Enable and configure NAT/PAT translation rules for URWB mobile units.

Use this task to allow communication between devices on an on-board local network and the WAN. This configuration is managed within the URWB network profile.

Before you begin

- Ensure the URWB mobility feature is enabled on the access point.
- Identify the IP address and netmask for the local network.

Follow these steps to configure NAT/PAT for URWB:

Procedure

1. Use the **configure terminal** command to enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Use the **wireless profile urwb** *urwb_profile_name* command to access the URWB profile configuration mode.

   Example

   ```
   Device(config)# wireless profile urwb URWB_PROFILE
   ```

3. Use the **nat enable** command to enable the NAT feature.

Example

```
Device(config-wireless-urwb-profile)# nat enable
```

4. Use the **nat address** *ip_address* **netmask** *netmask* command to define the inside IP address and netmask for the local network.

   Example

   ```
   Device(config-wireless-urwb-profile)# nat address 192.168.0.1 netmask
   255.255.255.0
   ```

5. Use the **nat rule** {**tcp**| **udp**} *outside-port* **inside** *inside-ip inside-port* command to add NAT port forwarding rules.

   Example

   ```
   Device(config-wireless-urwb-profile)# nat rule tcp 30022 inside 192.168.0.58
    22
   ```

   In this example, TCP traffic received on outside port 30022 is forwarded to 192.168.0.58 on port 22.

   The target host shall be located within the configured inside network address space.

6. (Optional) Use the **nat outside range min** *min* **max** *max* command to set the outside port range.

   Example

   ```
   Device(config-wireless-urwb-profile)# nat outside range min 25001 max 30000
   ```

7. Use the **end** command to exit to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-urwb-profile)# end
   ```

8. (Optional) Use the **show wireless profile urwb detailed** *urwb_profile_name* command to verify the configuration.

   Example

   ```
   Device# show wireless profile urwb detailed URWB_PROFILE
   ```

9. (Optional) Use the **show ip nat translations** command to display the current NAT translation mappings on the AP.

   Example

   ```
   Device# show ip nat translations
   ```

Digital I/O ports

Introduces digital I/O port functionality, explaining their features, operational characteristics, and integration within system architectures.

A digital I/O port is a software interface that:

- enables control of external hardware signals,
- facilitates the monitoring of input status, and
- allows for the management of status indicators such as LEDs.

The IW9165E supports mapping GPIO pins to character devices (e.g., `/dev/dio-x`). These devices are exposed to the IOx framework, enabling customer-developed applications to programmatically access and manage digital I/O interfaces for external hardware control.

Best practices for digital I/O control

To ensure reliable and consistent operation of digital I/O ports, follow these best practices:

- Use standardized paths: Always interact with the `/dev/dio-x` paths to ensure compatibility with the system's hardware abstraction layer.
- Verify status: Read the exposed Digital I/O status using the supported device interface or status file for your application.

Control digital I/O ports

Provides instructions for controlling digital I/O ports, including step-by-step procedures for configuring, monitoring, and operating ports within various deployment scenarios.

Manage external hardware signals and indicators through the command line interface.

You can use the character device interface to send high or low signals to hardware components or monitor the status of connected inputs.

Before you begin

Ensure the application environment is configured to access the `/dev/` directory.

Follow these steps to control digital I/O ports:

Procedure

1. Use the **echo** command to set the output state of a digital I/O port.
   - To set a port to high, use: echo 1 > /dev/dio-1
   - To set a port to low, use: echo 0 > /dev/dio-1

   Example

   ```
   Device# echo 1 > /dev/dio-1
   ```

   Example

   ```
   Device# echo 0 > /dev/dio-1
   ```

2. Monitor the input status of a port by using the corresponding device file.
   - To monitor the digital I/O output port status (high), use: echo 1 > /dev/dio-3
   - To monitor the digital I/O output port status (low), use: echo 0 > /dev/dio-4

Example

```
Device# echo 1 > /dev/dio-3
```

Example

```
Device# echo 0 > /dev/dio-4
```

3. Control an LED by writing to the associated device file.
   - To turn on the LED (1), use: echo 1 > /dev/dio-5
   - To turn off the LED (1), use: echo 0 > /dev/dio-5

Example

```
Device# echo 1 > /dev/dio-5
```

Example

```
Device# echo 0 > /dev/dio-5
```

Digital I/O device mapping

Details digital I/O device mapping, describing how devices are associated with I/O ports and offering reference information for hardware and logical mappings.

The table lists the software-exposed device nodes used by the application environment.

- Each entry specifies the device file corresponding to a digital I/O port configured for input or output and the device file that controls an LED indicator.
- Mapping device files is essential for managing and interacting with hardware interfaces through programs or command-line operations.

Table 171: Device file mapping for digital I/O ports and LEDs

GPIOs	Digital I/O ports	Function
GPIO29	/dev/dio-1	Digital I/O 1 output
GPIO31	/dev/dio-2	Digital I/O 2 output
GPIO38	/dev/dio-3	Digital I/O 1 output
GPIO39	/dev/dio-4	Digital I/O 2 output
GPIO51	/dev/dio-5	LED 1
GPIO48	/dev/dio-6	LED 2

> **Note**
>
> The GPIOs in the table are the logical device nodes exposed to IOx.

Power saving modes

Outlines power saving modes, explaining methods, criteria, and operational impact of reducing power consumption for digital I/O devices.

From release 26.2.1, Cisco IW9165E AP supports power saving modes through the exposed low-power control interface.

A power saving mode is a configuration state that

- reduces energy consumption for devices powered by external battery sources,
- maintains essential connectivity for remote management, and
- enables automated control of connected peripherals.

Sleep modes

The AP supports two types of power saving modes:

- Light sleep: In this mode, the device keeps the URWB backhaul radio on, and other radios and ethernet ports are shut down. The users can restore the device to active mode at any time via the backhaul radio. Enabling this mode reduces power consumption by 1.5 W to 2.4 W, depending on the traffic load.
- Deep sleep: This mode is not supported in release 26.2.1.

Features of power saving modes

- Automated control: Network administrators can use custom IOx applications to manage power states and control connected devices.
- Application integration: The Cisco App-hosting Framework (CAF) facilitates communication between control systems and IOx applications to execute power-saving logic.

> **Note**
>
> Power saving modes are not persistent. The device will default to active mode after a reload or power cycle.

Manage power saving modes

Provides instructions for managing power saving modes, including procedures for enabling and configuring energy-saving features on digital I/O devices.

Transition the device into a low-power state to conserve battery life during periods of inactivity.

Use this task when the device is deployed in mobile environments where power efficiency is required.

Before you begin

- Ensure the IOx application is deployed and configured to communicate with your control system.

- Verify that the necessary digital I/O interfaces are mapped to your IOx application. See Digital I/O device mapping section.

Follow these steps to manage power saving modes:

Procedure

1. Access your control system interface.

   Example

   ```
   Engineering: How to access the control system interface?
   ```

2. Send the sleep command to the IOx application.

   Example

   ```
   Device#: echo 1 > /mnt/host_cpu/low_power_mode
   ```

   The IOx application executes the power-saving logic to transition the device to light sleep mode.

3. Send the wake-up command from your control system to the IOx application to restore full functionality.

   Example

   ```
   Device# echo 0 > /mnt/host_cpu/low_power_mode
   ```

   The IOx application executes the power-saving logic to transition the device to active mode.

IOx frameworks

Introduces IOx frameworks, describing the core concepts, architecture, and operational principles that support IOx application deployment and management on network devices.

An IOx framework is an application hosting environment that

- provides a containerized platform for running applications directly on network devices,
- extends device functionality beyond standard networking tasks, and
- enables edge computing by processing data locally at the network edge.

From release 26.2.1, Cisco IW9165E AP supports IOx. IOx combines Cisco IOS-XE with Linux. This combination allows developers to build and run applications in a secure, isolated environment. IOx leverages container technology such as Docker. This enables the deployment of custom code, analytics, and third-party software on Cisco hardware. As a result, the network device becomes a compute node.

IOx resource specifications for IW9165E

Details IOx resource specifications for IW9165E, outlining hardware requirements, resource limits, and supported configurations necessary for effective deployment and operation of IOx applications on the device.

The IW9165E supports application hosting with the following resource specifications:

- The Max Memory and Max App Size are dependent on the actual memory and storage available at run time in IW9165E

Table 172: IOx resource specifications

CPU Architecture	CPU Allocated (MFLOPS)	Max Memory Allocated (RAM)	Application Type	Max Number of Apps	Max APP size
ARM 64-bit	4800	100 MB	Docker	2	20 MB

Enable IOx using controller

Provides instructions for enabling IOx through a network controller, covering the procedures necessary to activate and configure IOx services via centralized management interfaces.

Enable the IOx framework on the IW9165E access point to support application hosting.

IOx is disabled by default on the IW9165E. You can enable it either through the Wireless Controller or directly via the AP CLI.

Before you begin

Ensure you have administrative access to the Wireless Controller or the AP console.

Follow these steps to enable IOx via the Wireless Controller:

Procedure

1. Use the **configure terminal** command to enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Use the **ap profile** *ap-profile-name* command to configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile iox
   ```

3. Use the **apphost** command to enable the apphost framework on Cisco APs.

   Example

   ```
   Device(config-ap-profile)# apphost
   ```

   This step enables IOx allowing for application hosting.

Enable IOx using AP

Guides the process of enabling IOx directly on an AP, including step-by-step procedures to activate, configure, and verify IOx capability locally on the access point.

Enable the IOx framework on the IW9165E access point to support application hosting.

IOx is disabled by default on the IW9165E. You can enable it either through the Wireless Controller or directly via the AP CLI.

Before you begin

Ensure you have administrative access to the Wireless Controller or the AP console.

Follow these steps to enable IOx via the AP CLI:

Procedure

1. Access the AP console
2. Use the **configure iox{enable| disable}** command to enable or disable IOx.

   Example

   ```
   Device# configure iox enable
   ```

3. (Optional) Use the **show iox status** command to view the status of IOx.

   Example

   ```
   Device# show iox status
   ```

   The system displays the current enabled or disabled status of the IOx framework.

URWB topology awareness

Explains how URWB topology awareness works across multiple controllers in a mobility group to ensure unified network visibility and high availability.

The URWB topology awareness feature is a mobility group capability that

- provides all the controllers in a mobility group with visibility into URWB telemetry across the entire group,
- synchronizes URWB topology information between peer controllers to ensure consistent network state.

Additional reference information

The URWB topology awareness feature allows controllers to share telemetry data within the mobility infrastructure. A controller that receives telemetry forwards it to all other controllers in the mobility group. Each peer then updates its local URWB telemetry database. This synchronizes topology information across the group, enabling consistent monitoring and troubleshooting.

In URWB deployments, a controller learns the network topology from the coordinator AP. When a single URWB network spans multiple controllers, only the controller receiving the telemetry from the coordinator AP has full visibility of the network nodes and links. Without synchronization, other controllers lack the URWB topology information and cannot provide a unified network view.

Configure URWB topology awareness (CLI)

Configure the coordinator Access Point (AP) to assign a unique URWB network name and enable topology awareness for telemetry across multiple controllers.

Set up URWB topology awareness on the Coordinator Access Point (AP) to help identify the network and improve telemetry reporting across multiple controllers.

Configuring topology awareness ensures the Coordinator AP can be easily identified in the network.

Before you begin

- Ensure the AP is in coordinator mode.
- Verify the AP has an active connection to the controller.

Follow these steps to configure the URWB network name and enable telemetry reporting on the coordinator AP.

Procedure

Use the **ap name** *ap-name* **urwb mode coordinator nwname** *network-name* command to configure the AP in coordinator mode and assign a unique network name to the URWB network.

Example

```
Device# ap name coordinator urwb mode coordinator nwname PTP1
```

Note

For wired-only coordinator configurations, use the **ap name** *ap-name* **urwb mode coordinator wired-only fixed nwname** *network-name* or **ap name** *ap-name* **urwb mode coordinator wired-only mobility nwname** *network-name* commands.

Configure URWB topology awareness (GUI)

Configure the URWB network name for an access point using the wireless management GUI

Use the wireless device management interface to assign a unique URWB network name. Assigning a unique name is essential for proper network coordination and grouping access points in URWB environments.

Before you begin

Ensure you have access credentials for the wireless management interface.

Procedure

1. Navigate to **Configuration** > **Wireless** > **Access Points**.
2. Select the desired **Access Point** from the list to open the **Edit AP** window.
3. Click the **URWB tab** to access the URWB configuration settings.
4. Select the **Coordinator** check box to enable the coordinator role for the selected AP.
5. Enter the unique identifier in the **URWB Network Name** field.

   The **URWB Network Name** field is mandatory and must be between 1 and 32 characters in length. Only alphanumeric characters, hyphens (-), dots (.), and underscores (_) are supported. Ensure the network name adheres to these requirements to avoid configuration errors.

6. Click **Update and Apply to Device** to save the configuration and push the changes to the coordinator AP.

Verify URWB network information on the coordinator AP

Use the show urwb network command to view the URWB network name and detailed topology information.

This task allows you to quickly verify URWB network details, such as name, coordinator, status, and last update, from the controller. This helps you troubleshoot and monitor your access point efficiently.

Procedure

1. Use the **show ap name** *ap-name* command to view the URWB network name and topology identifier information.

   Example

```
Device# show ap name coordinator urwb network
Mesh ID                           : 5.190.114.16
URWB Status                       : Enabled
Coordinator Mode                  : Enabled
Gateway AP MAC Address            : aaa.bbb.ccc
Gateway ID                        : 5.190.114.16
URWB Network Name                 : newux
Number of Connected Devices       : 1
Connected Devices:
IP Address       VLAN ID     Last Update
------------------------------------------------------------
11.1.1.11       1           05/25/26 06:06:07

Number of Nodes                   : 2
Number of Coordinator routes      : 1
Number of Fixed Links             : 1
Number of Mobility Links          : 0

URWB Network Nodes summary:

Node ID         Node Name               Device Model      Role
IP Address       Last Update
————————————————————————————————————————————————————————————
5.190.114.16    APCC6E.2ABE.7210          CW9166D1        Coordinator
11.1.1.11       05/25/26 06:05:37
5.255.224.16    APC4D6.66FF.E010          CW9177D         Backbone
11.1.1.11       05/25/26 06:05:37
```

2. Use the **show ap name coordinator urwb info** command to display URWB information for the coordinator.

   Example

```
Device# show ap name coordinator urwb info
URWB Status                       : Enabled
Radio MAC                         : aaa.bbb.ccc
URWB Profile Inuse                : URWB_PAIR1
Mesh ID                           : 5.190.114.16

Config errors from AP             : None

Network-key hash                  :
7fd211ec1ed86daa0d9cf3b0cf245c5ee8a8added48d966a3b9c61d7d2237b06
```

```
Coordinator Mode                  : Enabled
Network Name in Use               : newux
Multicast                         : Disabled
```

3. Use the **show wireless urwb network summary** command to display all the URWB networks configured on the controller.

   Example

```
Device# show wireless urwb network summary
URWB Networks

Network Name         Network ID            Coordinator MAC       Coordinator
 Mesh ID
-------------------------------------------------------------------------------------------
coordinator         5957441956030247710       aaa.bbb.ccc
5.190.114.16
```

4. Use the **show wireless urwb network detail** *nw-name* command to display URWB network details.

   Example

```
Device# show wireless urwb network detail coordinator
URWB Network Detail

Network Name                    : coordinator
Network ID                      : 5957441956030247710
Coordinator MAC                 : aaa.bbb.ccc
Coordinator Mesh ID             : 5.190.114.16
```

Ethernet port authentication

Explains how 802.1X authentication secures Ethernet ports in vehicle ground access scenarios by validating hardware and managing access through industry-standard protocols.

802.1X authentication on Ethernet ports is a security mechanism designed for vehicle devices that

- Validates wired clients through a centralized RADIUS server.
- Ensures that only authenticated hardware gains access to network resources.
- Secures Ethernet LAN ports in industrial vehicle-to-ground deployment scenarios.

Key features

Authenticating 802.1X on an Ethernet port provides these benefits:

- Prevents unauthorized devices from accessing the network by requiring authentication before allowing data traffic on the Ethernet port.
- You can validate client credentials using industry-standard protocols such as PEAP, EAP-TLS, and EAP-FAST.
- Tailored for Vehicle Ground Access scenarios, including robotic or autonomous vehicle fleets, where physical port security is critical.

Table 173: Hardware and software compatibility

Attribute	Requirement
Software Release	26.2.1 or later
Operating Mode	Wireless Backhaul (URWB)
Hardware Models	Supported on all URWB-capable Cisco Catalyst and Industrial Wireless (IW) series PIDs
Configuration	Managed via Wireless Backhaul Network Profile

Guidelines and limitations

Outlines the key guidelines and limitations you should follow when configuring 802.1X on Ethernet ports.

Before you configure 802.1X on Ethernet ports, follow these guidelines:

- Only one wired client per LAN port is supported in this release.
- Multiple clients behind a single LAN port (e.g., via a switch) are not supported
- Authentication protocols supported are limited to PEAP, EAP-TLS, and EAP-FAST.
- You can configure a maximum of two RADIUS servers.
- The RADIUS server group can be configured only when the URWB mobility role is set to client.
- IPv6 is not supported for RADIUS server group configurations.
- Extended information parameters such as coa_port, retransmit, interim interval, and timeout are not supported.
- The accounting mechanism is also not supported in this context.

Configure a RADIUS server and group

Configure a RADIUS server and group to enable centralized authentication for users and devices.

Configure a RADIUS server and group (CLI)

Configure a RADIUS server and assign it to a server group to enable authentication services on network devices

Define a RADIUS server and assign it to a server group to enable authentication services.

This configuration is a prerequisite for enabling 802.1x authentication on Ethernet ports.

Before you begin

Ensure you have the RADIUS server IPv4 address, authentication port, and shared secret (key).

Perform these steps to configure a RADIUS server and group.

Procedure

1. Use the **configure terminal** command to enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Use the **RADIUS server** *radius_name* command to initiate the RADIUS server configuration. Define the server name, IPv4 address, and shared secret.

   Example

   ```
   Device(config)# radius server <radius-name>
   ```

3. Use the **address ipv4** {*ip-address* | *hostname*}[**auth-port** *port-number*] command to specify the server address and ports.

   Example

   ```
   Device(config)# address ipv4 xx.x.x.xx auth-port 1812
   ```

4. Use the **key** *shared_secret* command to set the shared secret.

   Example

   ```
   Device(config)# key Cisco123!
   ```

5. Use the **aaa group server** *group-name* command to create a server group and assign the defined RADIUS server to it.

   Example

   ```
   Device(config)# aaa group server radius <group-name>
   ```

6. Use the **server name** *radius-name* command to associate a RADIUS server with a RADIUS server group.

   Example

   ```
   Device(config-sg-radius)# server name <radius-name>
   ```

The RADIUS server is configured and assigned to a server group, enabling authentication services for network devices.

Configure a RADIUS server and group (GUI)

Configure a RADIUS server and server group to enable authentication management in your network device.

Establish centralized authentication by configuring a RADIUS server and creating server groups for robust network security

Use this task before enabling 802.1x authentication on Ethernet ports. A properly configured RADIUS setup ensures that network devices authenticate users and devices through a central server.

Before you begin

Configure a RADIUS server and group before enabling the 802.1x authentication on Ethernet ports using this procedure.

Procedure

1. Navigate to **Configuration** > **Security** > **AAA** > **Servers/Groups**
2. Under the **RADIUS** > **Servers** tab and click **Add**, define the Server name, IPv4 address, shared secret, and authentication port
3. Under the **RADIUS** > **Server Groups** tab and click **Add**, name the group and add the previously created RADIUS server to the assigned list.

The device is ready to use the configured RADIUS server and group for authentication processes across network interfaces.

Enable 802.1x authentication on Ethernet ports

Configure your Ethernet ports to require secure 802.1x authentication, ensuring only authorized devices access the network.

Enable 802.1X authentication on Ethernet ports (CLI)

Configure 802.1X authentication to secure the Ethernet LAN port on a vehicle device using CLI.

Enable 802.1X authentication for Ethernet LAN ports to enforce secure network access for wired clients.

This task enforces network access control for wired clients (such as robots or onboard controllers) by requiring authentication via a RADIUS server before permitting network access.

Before you begin

- Ensure the backhaul network is established.
- Have your RADIUS server IP, port, and shared secret ready.
- Verify the wired client supports PEAP, EAP-TLS, or EAP-FAST.

To enable 802.1X authentication on Ethernet ports, perform these steps

Procedure

1. Use the **configure terminal** and enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Use the **wireless profile urwb** *urwb-profile_name* command to access the wireless backhaul network profile.

   Example

   ```
   Device(config)# wireless profile urwb <urbw-profile_name>
   ```

3. Use the **local-auth radius-server-group** *radius-server-group_name* command to add the RADIUS server group to the profile.

   Example

   ```
   Device(config-wireless-urwb-profile)# local-auth radius-server-group
   <radius-group-name>
   ```

To remove the RADIUS server group association, use the **no local-auth radius-server-group** *<radius-server-group_name>* command.

4. Use the **end write memory** command to exit configuration mode and save your settings.

Example

```
Device(config-wireless-urwb-profile)# end
```

5. Use the **show ap name** *ap_name* **urwb info** command to confirm that the RADIUS server details and port settings are correctly pushed to the AP.

Example

```
Device# show ap name <ap_name> urwb info
```

Verify the status of the secondary Ethernet port if the client is connected to port 1.

802.1X authentication is enabled on the specified Ethernet port. Wired clients connecting to this port must successfully authenticate via the configured RADIUS server before gaining network access.

Enable 802.1X authentication on Ethernet ports (GUI)

Configure 802.1X authentication on Ethernet ports to secure wired network access using a RADIUS server.

Enable port-based authentication for wired clients. This ensures that only authorized devices can connect to the network.

802.1X authentication provides port-based network access control for wired clients. The system enforces authentication before granting network access. You must associate the Ethernet ports with a RADIUS server group to handle authentication requests.

Before you begin

- Ensure you have the RADIUS server IP address, port number, and shared secret.
- Verify that the wired client supports one of the EAP methods: PEAP, EAP-TLS, or EAP-FAST.
- Confirm that the RADIUS server is properly configured and reachable from the controller.

Procedure

---

1. Navigate to **Configuration** > **Tags and Profiles** > **URWB Network Profile**
2. Edit the **URWB Network Profile** and go to **Wired Clients** > **Local Authentication** > **RADIUS Server Group**
3. Choose your **RADIUS Server Group**
4. Click **Update and Apply to Device**

Wired clients must authenticate with the specified RADIUS server before accessing the network. Only authorized devices will be permitted, enhancing security for your wired infrastructure.

Troubleshoot RADIUS authentication with debug commands

Configure debug commands to identify errors, events, and packet-level issues during RADIUS authentication on Ethernet ports.

Enable and disable debug commands to gain visibility into the authentication process, helping you pinpoint causes of RADIUS authentication failures.

RADIUS authentication issues may occur during device access via Ethernet ports. Debug commands provide detailed logs to assist in troubleshooting. Use these steps when authentication errors or unexpected network events occur.

Before you begin

Follow these steps to troubleshoot RADIUS authentication on Ethernet ports using debug commands:

Procedure

1. Use the **debug urwb radius eth** command to enable specific debugging levels.

   Example

   ```
   Device# debug urwb radius eth {error | events | info | packet}
   ```

   - **error**: Displays RADIUS authentication error messages.
   - **events**: Logs authentication-related events.
   - **info**: Provides informative authentication details.
   - **packet**: Shows packet-level debugging output.

2. To disable the debugging, use the **no** form of the command.

   Example

   ```
   Device# no debug urwb radius eth {error | events | info | packet}
   ```

Debug logs are generated based on the selected level, helping you identify RADIUS authentication problems on Ethernet ports.

Verify RADIUS authentication status on an AP

Verify RADIUS authentication and view active authentication tables on an AP using show commands.

Ensure that wired client authentication on the AP is correctly configured and operating by checking RADIUS settings and active authentication tables.

Perform this task to maintain network security and troubleshoot authentication issues for wired clients by validating RADIUS configuration and monitoring active client authentication tables.

Before you begin

Ensure that RADIUS authentication is enabled for wired clients on the access point.

Follow these steps to verify RADIUS authentication and view wired client tables on the AP:

Procedure

1. Use the **show urwb wired radius config** command to display the current RADIUS configuration settings for the wired interface.

Example

```
Device# show urwb wired radius config
```

2. Use the **show urwb wired radius tables** command to view the active RADIUS authentication tables, which contain information about the wired clients currently being processed or authenticated.

   Example

   ```
   Device# show urwb wired radius tables
   ```

---

ou can review the RADIUS configuration and authentication tables, ensuring all settings are correct and wired clients are properly authenticated.

Part XVIII

VideoStream

Topics:

- VideoStream

196 VideoStream

Topics:

- Media stream

Media stream

Use the Media Stream feature to improve wireless multicast delivery. This feature converts multicast traffic to unicast and enables reliable packet acknowledgment by clients.

A Media Stream feature is a multicast delivery solution that

- converts IP multicast traffic into unicast frames for transmission over the air
- enables each wireless client to acknowledge receipt of video multicast data, and
- increases the reliability and quality of the media stream delivery in wireless networks.

The IEEE 802.11 wireless multicast delivery mechanism does not provide a reliable way to acknowledge lost or corrupted packets. As a result, if a multicast packet is lost during transmission, the system does not resend it, which may prevent you from viewing the IP multicast stream.

Cisco IOS XE Gibraltar 16.12.1 and later support IPv6 multicast addresses with the Media Stream feature. To configure media stream on IPv6 networks, specify IPv6 multicast addresses instead of IPv4 multicast addresses.

Prerequisites for media stream

This section lists the prerequisites that must be met before configuring media stream functionality, including network settings, controller mode, and client requirements.

These prerequisites must be satisfied before you configure media stream:

- Enable the multicast feature. Configure IP multicast on the controller in multicast-multicast mode.
- Check the IP address on the client machine. Verify that the client machine's IP address belongs to the appropriate VLAN.
- Verify that the access points have joined the controllers .

How to configure media stream

-

Configure the general parameters for a media stream (GUI)

Use this procedure to configure general parameters for a media stream in the wireless controller GUI.

Specify the parameters required to enable and manage multicast media streams for your network.

Procedure

1. Choose **Configuration** > **Wireless** > **Media Stream**.
2. In the **General** tab, check the **Multicast Direct Enable** check box.
3. In the **Session Message Config** section, check the **Session Announcement State** check box to enable the session announcement mechanism. The wireless controller applies the new general parameters for media streaming, and and notifies clients.
4. In the **Session Announcement URL** field, enter the URL where the client can find more information when an error occurs during the multicast media stream transmission.
5. In the **Session Announcement Email** field, enter the email address that users can contact for assistance.
6. In the **Session Announcement Phone** field, enter a phone number that users can call for assistance.
7. In the **Session Announcement Note** field, enter the reason why a client cannot be served with a multicast media.

8. Click **Apply**.

The wireless controller applies the new general parameters for media streaming, and clients are notified accordingly.

Configure multicast-direct globally for media stream (CLI)

Use this procedure to configure multicast-direct globally for media streams using CLI commands.

Enable and configure multicast-direct globally for all media streams on the wireless controller using CLI commands.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable multicast for wireless forwarding.

   Example

   ```
   Device(config)# wireless multicast
   ```

3. Enable IGMP snooping for each VLAN. If the global setting is disabled, all VLANs are considered disabled, regardless of their individual settings.

   Example

   ```
   Device(config)# ip igmp snooping
   ```

4. Enable a snooping querier on an interface when there is no multicast router in the VLAN to generate queries.

   Example

   ```
   Device(config)#ip igmp snooping querier
   ```

5. Configure the global multicast-direct on the controller.

   Example

   ```
   Device(config)# wireless media-stream multicast-direct
   ```

6. Configure the message configuration parameters, such as phone number, URL, email address, and notes.

   Example

   ```
   Device(config)# wireless media-stream message
   ```

   When a media stream is refused due to bandwidth constraints, a message can be sent to the affected user. These parameters define the IT support email address, explanatory notes to be displayed for the user, the redirect URL, and the phone number that users can contact regarding the refused stream.

7. Configure each media stream and its parameters, such as expected multicast destination addresses, stream bandwidth consumption, and stream priority parameters.

   Example

   ```
   Device(config)# wireless media-stream group grp-name startIp endIp
   ```

8. Return to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

The controller is now configured for multicast-direct for media streams as specified.

```
Device# configure terminal
Device(config)# wireless multicast
Device(config)# ip igmp snooping
Device(config)#ip igmp snooping querier
Device(config)# wireless media-stream multicast-direct
Device(config)# wireless media-stream message
 Email  Configure Session Announcement Email
 Notes  Configure Session Announcement notes
 URL    Configure Session Announcement URL
 phone  Configure Session Announcement Phone number
 <cr>
Device(config)# wireless media-stream group grp1 231.1.1.1 239.1.1.3
 avg-packet-size Configure average packet size
 default Set a command to its defaults
 exit Exit sub-mode
 max-bandwidth Configure maximum expected stream bandwidth in Kbps
 no Negate a command or set its defaults
 policy Configure media stream admission policy
 priority Configure media stream priority, <1:Lowest - 8:Highest>
 qos Configure over the air QoS class, <'video'> ONLY
 rrc-evaluation Configure RRC re-evaluation admission
 violation Configure stream violation policy on periodic re-evaluation
Device(config)# end
```

Configure media stream for 802.11 bands (CLI)

Learn how to configure media streams for 802.11 bands using CLI commands.

Enable and optimize media stream handling for 802.11 wireless bands by configuring relevant settings and parameters using CLI commands.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure whether media stream (multicast to unicast) is allowed for the 802.11 band.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} media-stream multicast-direct
```

You must disable the 802.11 network to enable media stream.

3. Optional: Configure the redirection of unicast video traffic to the best-effort queue.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} media-stream video-redirect
   ```

4. Configure the media stream to be sent through the best-effort queue if that media stream cannot be prioritized due to bandwidth-availability limitations.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} media-stream multicast-direct
   admission-besteffort
   ```

   Run the **no** form of the command to drop the stream if the media stream cannot be prioritized due to bandwidth-availability limitations.

5. Configure the maximum number of media streams allowed per client.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} media-stream multicast-direct
   client-max value
   ```

   The maximum is 15 and the default is 0. The value of 0 denotes unlimited streams.

6. Configure the maximum number of radio streams.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} media-stream multicast-direct
   radio-maximum value
   ```

   The valid range is from 1 to 20. The default is 0. The value of 0 denotes unlimited streams.

7. Configure maximum media (voice and video) bandwidth as a percentage.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} cac multimedia max-bandwidth
   bandwidth
   ```

   The range is from 5 percent to 85 percent.

8. Configure the minimum PHY rate needed for a client to send a media stream as unicast. If a client communicates below this rate, it will not receive the media stream as a unicast flow.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} cac media-stream multicast-direct
    min_client_rate dot11_rate
   ```

Typically, this PHY rate is equal to or higher than the rate at which multicast frames are sent.

9. Configure Call Admission Control (CAC) parameters for the media stream access category.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} cac media-stream
   ```

10. Configure CAC parameters for media access category used for voice and video.

    Example

    ```
    Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} cac multimedia
    ```

11. Configure CAC parameters for voice access category.

    Example

    ```
    Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} cac voice
    ```

12. Return to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

    Example

    ```
    Device(config)# end
    ```

The 802.11 bands are configured to handle media streams according to your specified parameters, enabling efficient multicast-to-unicast conversion and optimized bandwidth management for media traffic.

```
Device# configure terminal
Device(config)# ap dot11 24ghz media-stream multicast-direct
Device(config)# ap dot11 24ghz media-stream video-redirect
Device(config)# ap dot11 24ghz media-stream multicast-direct
admission-besteffort
Device(config)# ap dot11 24ghz media-stream multicast-direct client-max 15
Device(config)# ap dot11 24ghz media-stream multicast-direct radio-maximum 20
Device(config)# ap dot11 24ghz cac multimedia max-bandwidth 60
Device(config)# ap dot11 24ghz cac media-stream multicast-direct min_client_rate
Device(config)# ap dot11 5ghz cac media-stream
Device(config)# ap dot11 5ghz cac multimedia
Device(config)# ap dot11 5ghz cac voice
Device(config)# end
```

Configure a WLAN to stream video (GUI)

This task explains how to enable video streaming on a WLAN using the web GUI, so clients can access multicast video streams efficiently.

Procedure

---

1. Choose **Configuration** > **Wireless** > **WLANs** > **Wireless Networks**.
2. Select a **WLAN** to view the **Edit WLAN** window.

3. Click the **Advanced** tab.
4. Check the **Media Stream Multicast-Direct** check box to enable the feature.
5. Click **Update & Apply to Device**.

The selected WLAN now supports multicast-direct video streaming, allowing wireless clients to receive optimized video streams.

Configure WLAN to stream video (CLI)

Use this procedure to configure a WLAN for video streaming using command-line interface commands.

Enable video streaming functionality on a WLAN by configuring multicast-direct media streaming.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter WLAN configuration mode.

   Example

   ```
   Device(config)# wlan wlan_name
   ```

3. Disable WLAN for configuring its parameters.

   Example

   ```
   Device(config-wlan)# shutdown
   ```

4. Configure the multicast-direct media stream for the WLAN.

   Example

   ```
   Device(config)# media-stream multicast-direct
   ```

5. Enable the WLAN.

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

6. Return to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

   Example

   ```
   Device(config)# end
   ```

The WLAN is configured to support optimized video streaming via multicast-direct media streams.

```
Device# configure terminal
Device(config)# wlan wlan50
Device(config-wlan)# shutdown
Device(config)# media-stream multicast-direct
Device(config-wlan)# no shutdown
Device(config)# end
```

Enable a media stream per WLAN (GUI)

Use this procedure to enable a media stream for each WLAN using the GUI.

You can prioritize media streams over each WLAN to improve video and audio transmission quality.

Perform this task when you need to ensure media streaming is supported and optimized on specific WLANs.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. On the **WLANs** page, click the name of the **WLAN** or click **Add** to create a new one.
3. In the **Add/Edit WLAN** window that is displayed, click the **Advanced** tab.
4. Check the **Enabling a Media Stream for each WLAN** check box to enable the media stream on the WLAN.
5. Save the configuration.

You have enabled media streaming for the selected WLAN. Streaming traffic is now optimized.

Enable a media stream per WLAN (CLI)

Use this procedure to enable a media stream for a WLAN using CLI commands. This procedure allows you to provide multicast-direct media streaming on a per-WLAN basis.

Enable multicast media streaming functionality for a specified WLAN.

Perform this procedure when you need to configure multicast media streaming on individual WLANs in your wireless network environment.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter WLAN configuration mode.

   Example

   ```
   Device(config)# wlan wlan-name
   ```

3. Disable the WLAN for configuring its parameters.

Example

```
Device(config-wlan)# shutdown
```

4. Configure multicast-direct for the WLAN.

Example

```
Device(config-wlan)# media-stream multicast-direct
```

5. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

The WLAN is now enabled with multicast-direct media streaming.

```
Device# configure terminal
Device(config)# wlan wlan5
Device(config-wlan)# shutdown
Device(config-wlan)# media-stream multicast-direct
Device(config-wlan)# no shutdown
```

Configure the general parameters for a media stream (GUI)

Use this procedure to configure general parameters for a media stream in the wireless section. This ensures correct setup of session announcements and global multicast settings through the GUI.

Configure the general parameters for a media stream to enable wireless multicast functionality and session announcement options. This helps you ensure the media stream operates correctly in your wireless network environment.

Procedure

---

1. Choose **Configuration** > **Wireless** > **Media Stream**.
2. Check the **Multicast Direct Enable** check box to enable multicast direct globally on the local mode.
3. In the **Session Message Config** section, enter values for these parameters:
   - Session Announcement URL
   - Session Announcement Email
   - Session Announcement Phone
   - Session Announcement Note
4. Save the configuration.

The system saves and applies the general parameters for the media stream.

Configure the general parameters for a media stream (CLI)

Use this procedure to configure the general parameters for a media stream using CLI commands.

Configure phone, URL, email, and notes for a media stream and enable multicast direct globally in a local wireless environment.

Perform this procedure when you want to apply or change general configuration parameters for a media stream through the CLI.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure various message configuration parameters, such as phone, URL, email, and notes.

   Example

   ```
   Device(config)# wireless media-stream message {url url | email email-address
    | phone phone-number | notes notes}
   ```

3. Enable multicast direct globally for local mode.

   Example

   ```
   Device(config)# wireless media-stream multicast-direct
   ```

   > **Note**
   >
   > This configuration will not impact flex and fabric media-stream configurations.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# exit
   ```

The general parameters for the media stream are now configured, and multicast direct is enabled for local mode.

```
Device# configure terminal
Device(config)# wireless media-stream message url www.xyz.com
Device(config)# wireless media-stream multicast-direct
Device(config)# exit
```

Add media stream (CLI)

Use this task to add a media stream by entering the appropriate commands on the CLI.

Configure media streams to manage multicast traffic efficiently on the device.

Media streams help you improve multicast traffic performance for applications such as streaming video or audio over wireless networks. Use media streams to reserve resources and prioritize bandwidth.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure each media stream and set parameters such as multicast destination addresses, bandwidth consumption, and priority.

   Example

   ```
   Device(config)# wireless media-stream group groupName startIpAddr endIpAddr
   ```

3. Configure the average packet size.

   Example

   ```
   Device(media-stream)# avg-packet-size packetsize
   ```

4. Configure the maximum expected stream bandwidth, in Kbps.

   Example

   ```
   Device(media-stream)# max-bandwidth bandwidth
   ```

5. Configure the media stream admission policy.

   Example

   ```
   Device(media-stream)# policy {admit | deny}
   ```

6. Configure the over-the-air QoS class as "video".

   Example

   ```
   Device(media-stream)# qos video
   ```

7. Configure the violation mode.

   Example

   ```
   Device(media-stream)# violation {drop | fallback}
   ```

8. Configure Resource Reservation Control (RRC) re-evaluation admission to provide initial or periodic admission evaluation. This process occurs at intervals of 2, 4, 8, or more seconds.

Example

```
Device(media-stream)# rrc-evaluation {initial | periodic}
```

9. Set the priority value. The valid range is 1 to 8, where 1 is the lowest.

Example

```
Device(media-stream)# priority priority-value
```

The system configures the media stream using your parameters and activates it for use.

```
Device# configure terminal
Device(config)# wireless media-stream group group1 224.0.0.0 224.0.0.223
Device(media-stream)# avg-packet-size 100
Device(media-stream)# max-bandwidth 80
Device(media-stream)# policy admit
Device(media-stream)# qos video
Device(media-stream)# violation drop
Device(media-stream)# rrc-evaluation initial
Device(media-stream)# priority 6
```

Delete a media stream (GUI)

Use this procedure to delete a media stream in the user interface when it is no longer needed.

Remove an existing media stream to keep only necessary and relevant streams in the system.

Procedure

---

1. Choose **Configuration** > **Wireless** > **Media Stream**.
2. Click the **Streams** tab.
3. Check the check box next to the stream name you want to delete.

   To delete multiple streams, select the checkboxes next to each stream name.
4. Click **Delete**.
5. Click **Yes** in the confirmation window to delete the media stream.

The selected media stream is permanently deleted from the system. It is no longer available for use or configuration.

Delete a media stream (CLI)

Use this procedure to delete a configured media stream from your device using CLI commands when the media stream is no longer needed or must be reconfigured.

Remove a configured media stream from the device using CLI commands when the media stream is no longer required or needs to be reconfigured.

Before you begin

Before you delete a media stream, make sure it is enabled and configured.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Delete the media stream identified by the command.

   Example

   ```
   Device(config)# no wireless media-stream group media_stream_name
   ```

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The specified media stream is removed from the device configuration and is no longer active.

```
Device# configure terminal
Device(config)# no wireless media-stream group grp1
Device(config)# end
```

Configure multicast direct admission control (GUI)

Use the GUI to enable and configure multicast direct admission control settings, improving media stream management on wireless networks.

This task explains how to enable and configure multicast direct admission control using the GUI.

Procedure

1. Choose **Configuration** > **Wireless** > **Media Stream**.
2. Check the **Media Stream Admission Control (ACM)** check box to enable multicast direct admission control.
3. In the **Maximum Media Stream RF bandwidth (%)** field, enter the percentage of maximum bandwidth for media applications on this radio band. The valid range is 5 to 85. If your client reaches the specified value, new calls on this radio band are rejected.
4. In the **Maximum Media Bandwidth (%)** field, enter the bandwidth. The valid range is 5 percent to 85 percent.
5. From the **Client Minimum Phy Rate** drop-down list, select the minimum transmission data rate in kilobits per second that the client can use. If the transmission data rate is below the physical rate, video does not start or your client is classified as a bad client. The system then demotes the video for better effort quality of service (QoS) or denies it.
6. In the **Maximum Retry Percent (%)** field, enter the percentage of maximum retries that are allowed. The default value is 80. If the value exceeds 80, video does not start or your client might be classified as a bad client. The system can demote the video for better effort QoS or deny it.

7. Click **Apply**.

You have activated multicast direct admission control.

Configure multicast direct admission control (CLI)

Learn how to configure multicast direct admission control using commands.

Configure multicast direct admission control to:

- ensure that media streams are efficiently delivered to wireless clients
- maintain optimal bandwidth allocation for multicast video and audio traffic, and
- enforce admission control policies that prevent network congestion and protect media quality.

To configure multicast direct admission control, perform these steps:

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Disable the 802.11 network.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} shutdown
   ```

3. Configure the redirection of the unicast video traffic to the best effort queue.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} media-stream video-redirect
   ```

4. Enable admission control on the media-stream access category.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} cac media-stream acm
   ```

5. Configure the maximum media bandwidth, in percent.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} cac media-stream max-bandwidth
    bandwidth
   ```

   The range is from 5 percent to 85 percent.

6. Configure the maximum bandwidth allocated for media to Wi-Fi Multimedia (WMM) clients.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} cac multimedia max-bandwidth
bandwidth
```

The range is from 5 percent to 85 percent.

7. Configure the minimum PHY rate needed for a client to receive media stream as unicast.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} cac media-stream multicast-direct
 min-client-rate dot11Rate
```

If your client communicates below this rate, it will not receive the media stream as a unicast flow. Usually, this PHY rate is equal to or higher than the rate multicast frames are sent.

8. Configure the maximum retry percent CAC parameter for multicast-direct streams.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} cac media-stream multicast-direct
 max-retry-percent retryPercent
```

9. Configure the maximum number of radio streams.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} media-stream multicast-direct
radio-maximum value
```

The valid range is 1 to 20. Default is 0. Value 0 denotes unlimited streams.

10. Configure the maximum number of allowed media streams for each client.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} media-stream multicast-direct
client-maximum value
```

The maximum value is 15, and the default is 0. A value of 0 denotes unlimited streams.

11. Configure the media stream so that it is sent through the best effort queue if it cannot be prioritized due to limited bandwidth.

Example

```
Device(config)# ap dot11 {24ghz | 5ghz| 6ghz} media-stream multicast-direct
admission-besteffort
```

Add **no** in the command to drop the stream when the media stream cannot be prioritized because bandwidth is unavailable.

12. Enable the 802.11 network.

Example

```
Device(config)# no ap dot11 {24ghz | 5ghz| 6ghz} shutdown
```

You have successfully configured multicast direct admission control on the specified wireless network interface.

```
Device# configure terminal
Device(config)# ap dot11 24ghz shutdown
Device(config)# ap dot11 24ghz media-stream video-redirect
Device(config)# ap dot11 24ghz cac media-stream acm
Device(config)# ap dot11 24ghz cac media-stream max-bandwidth 65
Device(config)# ap dot11 24ghz cac multimedia max-bandwidth 65
Device(config)# ap dot11 24ghz cac media-stream multicast-direct min-client-rate
 800
Device(config)# ap dot11 24ghz cac media-stream multicast-direct
max-retry-percent 50
Device(config)# ap dot11 24ghz media-stream multicast-direct radio-maximum 10
Device(config)# ap dot11 24ghz media-stream multicast-direct client-maximum
12
Device(config)# ap dot11 24ghz media-stream multicast-direct
admission-besteffort
Device(config)# no ap dot11 24ghz shutdown
```

Create and attach policy-based QoS profile

Create and attach policy-based QoS profile.

The high-level steps to create and attach policy-based QoS profile are as follows:

1. Create a QoS profile
2. Create a service template
3. Map the service template to thepPolicy map
4. Map the policy map to the policy profile

Create a QoS profile (GUI)

Learn how to create a QoS profile using the web interface to prioritize network traffic according to business needs.

Define and apply policies that optimize network resource allocation for various types of traffic.

Procedure

---

1. Click **Configuration** > **Services** > **QoS**.
2. Click **Add** to create a new QoS Policy.
3. Enter a **Policy Name**.
4. Enter a **Description** for the policy.
5. In the **Class Default** section, choose a value in the **Mark** drop-down list.
6. Enter the **Police (kbps)** value.
7. Click **Apply to Device**.

The new QoS profile is deployed to the selected devices. It enforces your traffic priorities immediately.

Create a QoS profile (CLI)

Learn how to create a QoS profile using commands.

Use this procedure to configure a custom Quality of Service (QoS) profile on your device by using the CLI. This configuration lets you manage network traffic according to your needs.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a policy map.

   Example

   ```
   Device(config)# policy-map policy-map-name
   ```

3. Add a description to the policy map.

   Example

   ```
   Device(config-pmap)# description description
   ```

4. Create policy criteria.

   Example

   ```
   Device(config-pmap)# class class-map-name
   ```

5. Set the committed information rate for policing.

   Example

   ```
   Device(config-pmap-c)# police cir committed-information-rate
   ```

6. Configure what action occurs when the rate is less than the conform burst value.

   Example

   ```
   Device(config-pmap-c-police)# conform-action drop
   ```

7. Configure what action occurs when the rate is between the conform burst and the sum of conform burst and exceed burst values.

   Example

   ```
   Device(config-pmap-c-police)# exceed-action drop
   ```

8. Return to privileged EXEC mode.

Example

```
Device(config-pmap-c-police)# end
```

The device creates and stores the QoS profile. Display the policy map and class map with the appropriate CLI commands to verify your configuration.

```
Device# configure terminal
Device(config)# policy-map QoS_Drop_Youtube
Device(config-pmap)# description QoS_Drop_Youtube
Device(config-pmap)# class QoS_Drop_Youtube1_AVC_UI_CLASS
Device(config-pmap-c)# police cir 8000
Device(config-pmap-c-police)# conform-action drop
Device(config-pmap-c-police)# exceed-action drop
Device(config-pmap-c-police)# end
```

Create a service template (GUI)

Use this procedure to create a new service template using GUI, specifying VLAN, session timeout, access controls, and QoS options for network clients.

Create a reusable configuration profile that defines network service attributes–such as VLAN, session timeout, access controls, and Quality of Service (QoS) policies–for consistent and efficient assignment to one or more devices.

Procedure

1. Choose **Configuration** > **Security** > **Local Policy**.
2. On the **Local Policy** page, **Service Template** tab, click **Add**.
3. In the **Create Service Template** window, configure these parameters:
   - **Service Template Name**: Enter a name for the template.
   - **VLAN ID**: Enter the VLAN ID for the template. The valid range is 1 to 4,094.
   - **Session Timeout (secs)**: Set the timeout duration for the template. The valid range is 1 to 65,535.
   - **Access Control List**: Select the Access Control List from the drop-down list.
   - **Ingress QOS**: Select the input QoS policy for the client from the drop-down list.
   - **Egress QOS**: Select the output QoS policy for the client from the drop-down list.
4. Click **Apply to Device**.

The new service template is created and available for assignment to network devices.

Map the service template to the policy map (CLI)

Map the service template to the policy map using commands.

Map a service template to a policy map using CLI commands to enable attribute-driven service control.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Specify the parameter map type and name.

Example

```
Device(config)# parameter-map type subscriber attribute-to-service
parameter-map-name
```

3. Specify the parameter map attribute filter criteria. Multiple filters are used in the example provided here.

Example

```
Device(config-parameter-map-filter)# map-index map device-type eq filter-name
 user-role eq user-name
```

4. Specify the service template.

Example

```
Device(config-parameter-map-filter-submode)# map-index service-template
service-template-name precedence precedence-num
```

5. Return to privileged EXEC mode.

Example

```
Device(config-parameter-map-filter-submode)# end
```

6. Enter global configuration mode.

Example

```
Device# configure terminal
```

7. Specify the policy map type.

Example

```
Device(config)# policy-map type control subscriber policy-map-name
```

8. Specify the match criteria to the policy map.

Example

```
Device(config-event-control-policymap)# event identity-update match-all
```

9. Apply a class map using a service template.

Example

```
Device(config-event-control-policymap)# class-num class always do-until-failure
```

10. Apply the parameter map.

Example

```
Device(config-event-control-policymap)# action-index map attribute-to-service
 table parameter-map-name
```

The service template is successfully mapped to the specified policy map.

```
Device# configure terminal
Device(config)# parameter-map type subscriber attribute-to-service
QoS-Policy_Map-param
Device(config-parameter-map-filter)# 1 map device-type eq "Android" user-role
 eq "student"
Device(config-parameter-map-filter)# 1 service-template Qos_template
Device(config-parameter-map-filter-submode)# end
Device# configure terminal
Device(config)# policy-map type control subscriber QoS-Policy_Map
Device(config-event-control-policymap)# event identity-update match-all
Device(config-event-control-policymap)# 1 class always do-until-failure
Device(config-event-control-policymap)# 1 map attribute-to-service table
QoS-Policy_Map-param
```

Map service template to the policy map (GUI)

Use this procedure to map a service template to a policy map using GUI.

Map a service template to an existing policy map. This enables deployment of predefined configurations.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. On the **Policy Profile** page, select the **Policy Profile** to be mapped.
3. In the **Edit Policy Profile** window, click the **Access Policies** tab.
4. Use the **Local Subscriber Policy Name** drop-down list to select the policy name.
5. Click **Update & Apply to Device**.

The selected service template is now mapped to the policy map and applied to the device.

Map the service template to policy map (CLI)

Use this task to map a service template to a policy map by entering the required commands in the CLI. This enables the policy map to apply the service template's configurations to network traffic.

Map a service template to a policy map so the policy map automatically applies the desired service configurations to targeted network traffic.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the parameter map type and name.

   Example

   ```
   Device(config)# parameter-map type subscriber attribute-to-service
   parameter-map-name
   ```

3. Specify the parameter map attribute filter criteria. You can use multiple filters.

   Example

   ```
   Device(config-parameter-map-filter)# map-index map device-type eq filter-name
    user-role eq user-name
   ```

4. Specify the service template.

   Example

   ```
   Device(config-parameter-map-filter-submode)# map-index service-template
   service-template-name [precedence precedence-num]
   ```

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config-parameter-map-filter-submode)# end
   ```

6. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

7. Specify the type of policy map.

   Example

   ```
   Device(config)# policy-map type control subscriber policy-map-name
   ```

8. Specify the match criteria to the policy map.

   Example

   ```
   Device(config-event-control-policymap)# event identity-update match-all
   ```

9. Apply a class map with a service template.

   Example

```
Device(config-event-control-policymap)# class-num class always do-until-failure
```

10. Apply a parameter map.

    Example

```
Device(config-event-control-policymap)# action-index map attribute-to-service
 table parameter-map-name
```

The policy map is associated with the service template. It enforces the defined service configurations on specified network traffic."

```
Device# configure terminal
Device(config)# parameter-map type subscriber attribute-to-service
QoS-Policy_Map-param
Device(config-parameter-map-filter)# 1 map device-type eq "Android" user-role
 eq "student"
Device(config-parameter-map-filter-submode)# 1 service-template Qos_template
Device(config-parameter-map-filter-submode)# end
Device# configure terminal
Device(config)# policy-map type control subscriber QoS-Policy_Map
Device(config-event-control-policymap)# event identity-update match-all
Device(config-event-control-policymap)# 1 class always do-until-failure
Device(config-event-control-policymap)# 1 map attribute-to-service table
QoS-Policy_Map-param
```

Map the policy map (GUI)

Learn how to map a policy map using the GUI in order to apply service templates and criteria to device policies.

Map a new policy map so that you can enforce local security and service templates based on device and user attributes.

Procedure

---

1. Choose **Configuration** > **Security** > **Local Policy** > **Policy Map** tab.
2. Click **Add**.
3. Enter a name in the **Policy Map Name** text field.
4. Click **Add** to add the matching criteria information.
5. Select the service template from the **Service Template** drop-down list.
6. Select the filters from **Device Type**, **User Role**, **User Name**, **OUI**, and **MAC Address** drop-down lists.
7. Click **Add Criteria**.
8. Click **Apply to Device**.

The new policy map is created and applied according to the specified criteria.

Map the policy map (CLI)

Use this procedure to map a policy map to a wireless profile policy using CLI commands.

Map a policy map to a wireless policy profile using CLI commands to apply specific subscriber policies.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a wireless policy profile.

   Example

   ```
   Device(config)# wireless profile policy wlan-policy-profile-name
   ```

3. Add a description for the policy profile.

   Example

   ```
   Device(config-wireless-policy)# description profile-policy-description
   ```

4. Configure the subscriber policy name.

   Example

   ```
   Device(config-wireless-policy)# subscriber-policy-name policy-name
   ```

After you complete these steps, the policy map is mapped to the wireless policy profile, and the configuration becomes active on the device.

```
Device# configure terminal
Device(config)# wireless profile policy wireless-policy-profile
Device(config-wireless-policy)# description "wireless policy profile"
Device(config-wireless-policy)# subscriber-policy-name QoS-Policy_Map
```

Monitor media streams

This section lists the commands used to monitor media streams and provides their descriptions.

To provide reference information about the available commands used to monitor media streams on wireless devices, including a description of each command to assist users in understanding and using these monitoring tools effectively.

Table 174: Commands for monitoring media streams

Commands	Description
show wireless media-stream client detail *group name*	Displays media stream client details of the particular group.
show wireless media-stream client summary	Displays the media stream information of all the clients.
show wireless media-stream group detail *group name*	Displays the media stream configuration details of the particular group.

Commands	Description
show wireless media-stream group summary	Displays the media stream configuration details of all the groups.
show wireless media-stream message details	Displays the session announcement message details.
show wireless multicast	Displays the multicast-direct configuration state.
show ap dot11 {24ghz | 5ghz} media-stream rrc	Displays 802.11 media resource reservation control configurations.

Verify media stream information

View media stream information.

Use these **show** commands to view the media stream information.

To view general media stream information and status, use these commands:

```
Device# show wireless media-stream multicast-direct state

Multicast-direct State..................... : enabled
Allowed WLANs:
WLAN-Name                                          WLAN-ID
-------------------------------------------------------
zsetup_mc                                          1
vwlc-mc_mo                                         3
mcuc_test1                                         4
mcuc_test2                                         5
```

```
Device# show wireless media-stream group summary


Number of Groups: 4

Stream Name          Start IP                End IP            Status
------------------------------------------------------------------------------
new2                 231.2.2.3               231.2.4.4         Enabled
my234                234.0.0.0               234.10.10.10      Enabled
uttest2              235.1.1.20              235.1.1.25        Enabled
uttest3              235.1.1.40              235.1.1.200       Enabled
```

To view the details of a particular media stream, use the **show wireless media-stream client detail** *media_stream_name* command:

```
Device# show wireless media-stream group detail uttest2


Media Stream Name         : uttest2
Start IP Address          : 235.1.1.20
End IP Address            : 235.1.1.25
RRC Parameters:
 Avg Packet Size(Bytes)   : 1200
 Expected Bandwidth(Kbps) : 1000
 Policy                   : Admitted
```

```
 RRC re-evaluation        : Initial
 QoS                      : video
 Status                   : Multicast-direct
 Usage Priority           : 4
 Violation                : Drop
```

To view RRC information for a dot11 band, use the **show ap dot11 {24ghz | 5ghz | 6ghz} mediastream rrc** command:

```
Device# show  ap dot11 5ghz media-stream rrc

Multicast-direct                 : Enabled
Best Effort                      : Disabled
Video Re-Direct                  : Disabled
Max Allowed Streams Per Radio    : Auto
Max Allowed Streams Per Client   : 5
Max Media-Stream Bandwidth       : 5
Max Voice Bandwidth              : 50
Max Media Bandwidth              : 43
Min PHY Rate (Kbps)              : 6000
Max Retry Percentage             : 5
```

To view session announcement message details, use the **show wireless media-stream message details** command:

```
Device# show wireless media-stream message details
URL          :
Email        : abc@cisc
Phone        :
Note         :
State        : Disabled
```

To view the list of clients in the blocked list database, use the **show ip igmp snooping igmpv2-tracking** command:

```
Device# show ip igmp snooping igmpv2-tracking

Client to SGV mappings
----------------------
Client: 10.10.10.215 Port: Ca1
 Group: 239.255.255.250 Vlan: 10 Source: 0.0.0.0 blacklisted: no
 Group: 234.5.6.7 Vlan: 10 Source: 0.0.0.0 blacklisted: no
 Group: 234.5.6.8 Vlan: 10 Source: 0.0.0.0 blacklisted: no
 Group: 234.5.6.9 Vlan: 10 Source: 0.0.0.0 blacklisted: no

Client: 10.10.101.177 Port: Ca2
 Group: 235.1.1.14 Vlan: 10 Source: 0.0.0.0 blacklisted: no
 Group: 235.1.1.16 Vlan: 10 Source: 0.0.0.0 blacklisted: no
 Group: 235.1.1.18 Vlan: 10 Source: 0.0.0.0 blacklisted: no

SGV to Client mappings
----------------------
Group: 234.5.6.7 Source: 0.0.0.0 Vlan: 10
 Client: 10.10.10.215 Port: Ca1 Blacklisted: no
```

To view wireless client summary, use the **show wireless media-stream client summary** command:

```
Device# show wireless media-stream client summary
```

To view details of a specific wireless media stream, use the **show wireless media-stream client detail** command:

```
Device# show wireless media-stream client detail uttest2

Media Stream Name         : uttest2
Start IP Address          : 235.1.1.20
End IP Address            : 235.1.1.25
RRC Parameters:
 Avg Packet Size(Bytes)   : 1200
 Expected Bandwidth(Kbps) : 1000
 Policy                   : Admitted
 RRC re-evaluation        : Initial
 QoS                      : video
 Status                   : Multicast-direct
 Usage Priority           : 4
 Violation                : Drop
```

Part XIX

Software-Defined Access Wireless

Topics:

- Software-Defined Access Wireless
- Passive Client
- Fabric in a Box with External Fabric Edge

Introduces Software-Defined Access wireless architecture and provides guidance on configuring, monitoring, and managing wireless mesh networks with inter-fabric edge switch roaming protection.

197 Software-Defined Access Wireless

Topics:

- Software-defined access wireless
- SD-Access wireless mesh inter fabric edge switch roaming protection
- Configure SD-access wireless
- SD-Access wireless verification commands

Introduces Software-Defined Access wireless architecture and provides guidance on configuring, monitoring, and managing wireless mesh networks with inter-fabric edge switch roaming protection.

Software-defined access wireless

Explains software-defined access wireless network topology and its components.

Software-defined access wireless is a network topology that

- passes traffic through inter-connected switches while providing the abstraction of a single Layer 2 or Layer 3 device
- provides seamless connectivity with policy application and enforcement at the edge of the fabric
- uses IP overlay to make the network appear as a single virtual entity without clustering technologies, and
- moves the enterprise network from VLAN-centric architecture to user group-based enterprise architecture with flexible Layer 2 extensions.

Network components and architecture

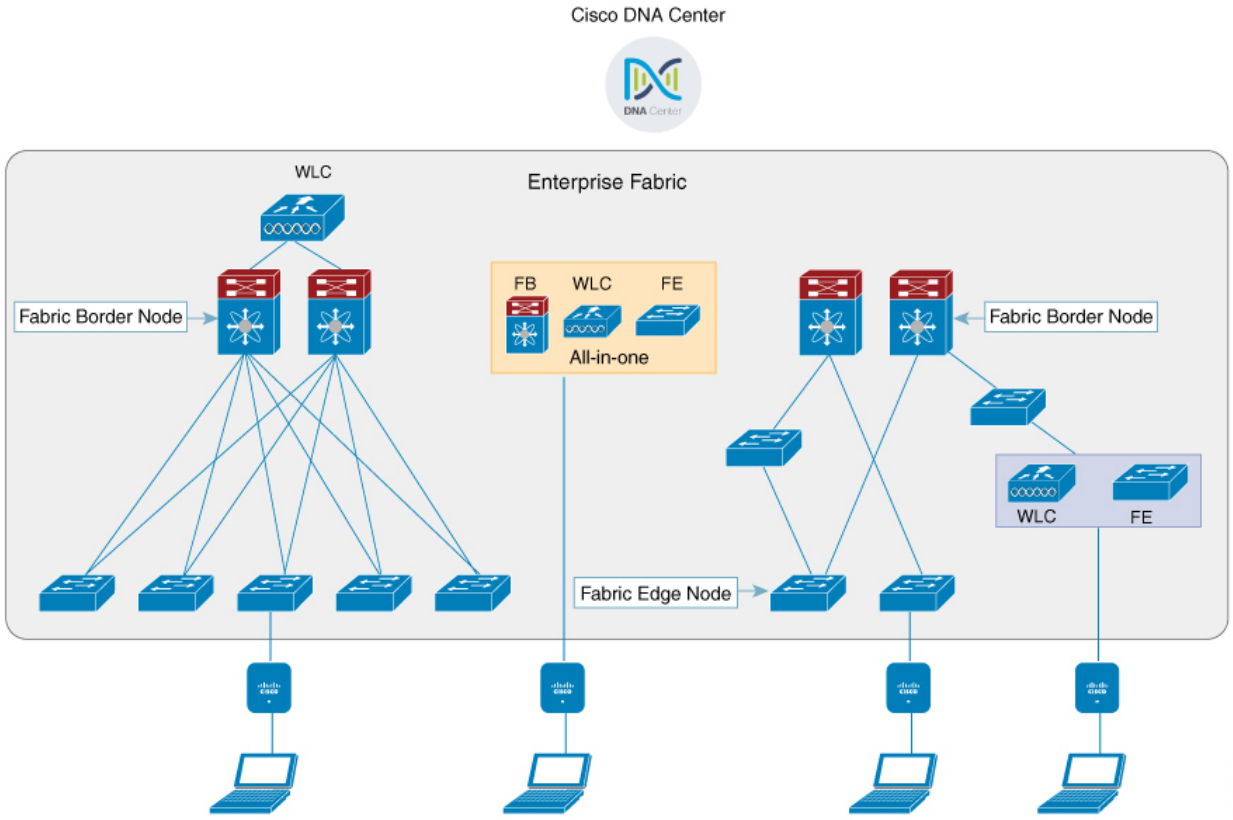
The Enterprise Fabric provides end-to-end enterprise-wide segmentation, flexible subnet addressing, and controller-based networking with uniform enterprise-wide policy and mobility. It moves the enterprise network from current VLAN-centric architecture to a user group-based enterprise architecture, with flexible Layer 2 extensions within and across sites.

The following definitions are used for fabric nodes:

- **Enterprise Fabric**: A network topology where traffic is passed through inter-connected switches, while providing the abstraction of a single Layer 2 or Layer 3 device.
- **Fabric Domain**: An independent operation part of the network. It is administered independent of other fabric domains.
- **End Points**: Hosts or devices that connect to the fabric edge node are known as end points (EPs). They directly connect to the fabric edge node or through a Layer 2 network.

This figure shows the components of a typical SD-Access Wireless. It consists of Fabric Border Nodes (BN), Fabric Edge Nodes (EN), Wireless Controller, Cisco Catalyst Center, and Host Tracking Database (HDB).

Figure 35: Software-Defined Access Wireless



The figure covers the following deployment topologies:

- All-IN-one Fabric–When we have all Fabric Edge, Fabric Border, Control-Plane and controller functionality enabled on a Cat 9K switch.

  This topology is depicted IN the mid part of the figure.

- Split topology–When we have Fabric Border, or Control Plane, or controller on a Cat 9K switch with separate Fabric Edge. This topology is depicted IN the left-most part of the figure.

- Co-located Fabric Edge and Controller–When we have Fabric Edge and controller on a Cat 9K switch. This topology is depicted IN the right-most part of the figure.

Cisco Catalyst Center: Is an open, software-driven architecture built on a set of design principles with the objective of configuring and managing Cisco Catalyst 9800 Series Wireless Controllers.

Control Plane: This database allows the network to determine the location of a device or user. When the EP ID of a host is learnt, other end points can query the database about the location of the host. The flexibility of tracking subnets helps IN summarization across domains and improves the scalability of the database.

Fabric Border Node (Proxy Egress Tunnel Router [PxTR or PITR/PETR] IN LISP): These nodes connect traditional Layer 3 networks or different fabric domains to the enterprise fabric domain. If there are multiple fabric domains, these nodes connect a fabric domain to one or more fabric domains, which could be of the same or different type. These nodes are responsible for translation of context from one fabric domain to another. When the encapsulation is the same across different fabric domains, the translation of fabric context is generally 1:1. The fabric control planes of two domains exchange reachability and policy information through this device.

Fabric Edge Nodes (Egress Tunnel Router [ETR] or Ingress Tunnel Router [ITR] IN LISP): These nodes are responsible for admitting, encapsulating or decapsulating, and forwarding of traffic from the EPs. They lie at the perimeter of the fabric and are the first points of attachment of the policy. EPs could be directly or indirectly attached to a fabric edge node using an intermediate Layer 2 network that lies outside the fabric domain. Traditional Layer 2 networks, wireless access points, or end hosts are connected to fabric edge nodes.

Wireless Controller: The controller provides AP image and configuration management, client session management and mobility. Additionally, it registers the MAC address of wireless clients IN the host tracking database at the time of client join, as well as updates the location at the time of client roam.

Access Points: AP applies all the wireless media specific features. For example, radio and SSID policies, webauth punt, peer-to-peer blocking, and so on. It establishes CAPWAP control and data tunnel to controller. It converts 802.11 data traffic from wireless clients to 802.3 and sends it to the access switch with VXLAN encapsulation.

The SDA allows to simplify:

- Addressing IN wireless networks
- Mobility IN wireless networks
- Guest access and move towards multi-tenancy
- Leverage Sub-net extension (stretched subnet) IN wireless network
- Provide consistent wireless policies

Note

Role co-location between wireless controller and fabric edge is supported.

Platform support information for software-defined access wireless includes compatibility with various Catalyst switches and wireless controllers.

Table 175: Supported platforms for software-defined access wireless

Platforms	Support
Catalyst 9300	Yes
Catalyst 9400	Yes
Catalyst 9500H	Yes
Cisco Catalyst 9800 Series Wireless Controller for Cloud	Yes
Cisco Catalyst 9800-40 Series Wireless Controller	Yes
Cisco Catalyst 9800-80 Series Wireless Controller	Yes

Table 176: Multi-instance support

Multi-instance	Support
Multiple LISP sessions	Yes
Emulated database support	Yes
Client roaming between WNCd instances	Yes

Table 177: Feature support

Feature	Support
Inter-WLC roam for IRCM	Only L2 mobility is supported as VLAN is stretched across the fabric.
DNS-IPv4-ACL	• ACLs are enforced at AP. • Controller needs to push the DNS-ACL information to AP.
IPv6 ACL for clients	Yes. Open, 802.11x, WebAuth, PSK WLANs, IPv6 address visibility are also supported.
Location tracking/Hyperlocation	Yes
Multicast Video-Stream (IPv4)	Yes
Smart Licensing	Yes

Table 178: Outdoor access points support

AP	Support
1542	Yes
1560	Yes

SD-Access wireless mesh inter fabric edge switch roaming protection

Describes SD-Access wireless mesh inter fabric edge switch roaming protection functionality.

SD-Access wireless mesh inter fabric edge switch roaming protection is a wireless mesh feature that

- prevents mesh access points from roaming to root access points or other mesh access points connected to different fabric edge switches under normal conditions
- allows roaming when the current mesh access point link quality degrades significantly below established thresholds, and
- maintains wireless client connectivity by restarting CAPWAP tunnels after roaming across fabric edge switches.

Roaming behavior and criteria

When a Mesh AP (MAP) finds a Root AP (RAP) or other MAP with better adjacency, it roams to that RAP or MAP. However, a MAP cannot roam to another MAP or RAP connected to a different fabric edge switch due to wireless client connectivity loss. The reason being the VXLAN tunnels of the wireless client cannot be moved to another fabric edge switch.

However, if the current MAP link is much worse than the link to RAP or MAP connected to a different fabric edge switch, then the MAP roams and restart its CAPWAP tunnel.

Thus, all the MAPs connected to the MAP restarts the CAPWAP tunnels as well. This allows the wireless clients to connect again and creates wireless client VXLAN tunnels in the fabric edge.

Note

The criteria for a MAP roaming to a RAP or MAP connected to a different fabric edge switch are the same as the mesh preferred parent. They are:

- If the current link SNR is worse than 12 dB and there is another better link. (Or)
- If the current link SNR is worse than 20 dB and there is another link with 20% or better SNR.

The mesh daisy chain roaming is not supported in RAP or MAP connected to different fabric edge switches.

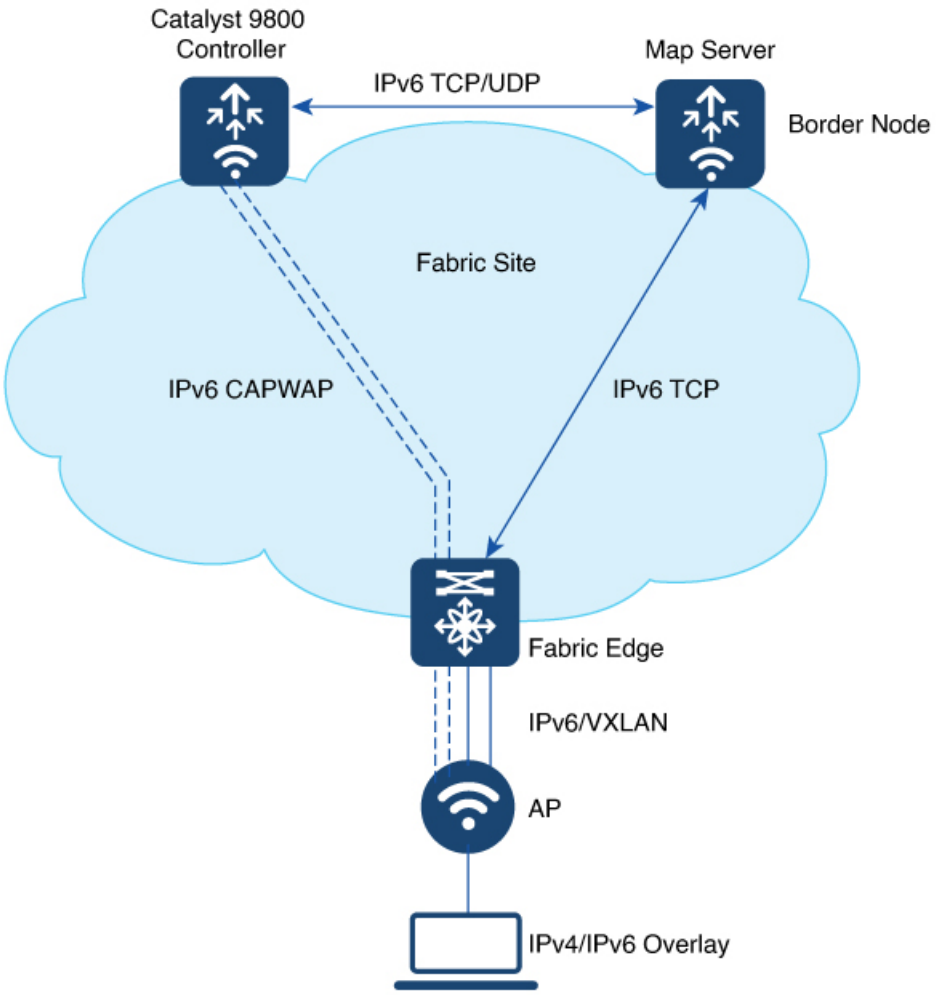
SDA IPv6 underlay support

This feature provides a wireless SDA IPv6 underlay support to enable IPv6-based communications in a fabric site. IPv6 is used to establish LISP connections between the controller and MAP server as well as between the MAP server and Fabric Edge. IPv6 underlay is also used to construct a VXLAN tunnel between the Fabric Edge and the AP.

The feature implementation is as follows:

- **Catalyst 9800 Controller**:
  - The controller manages IPv6-based LISP sessions to the MAP server
  - Encodes, decodes, and processes LISP messages through IPv6 TCP/UDP sockets.)
  - Communicates RLOC IPv6 address to the AP for VXLAN creation and client tunnel mapping
- **Fabric Edge**:
  - Processes MAP notification from MAP server
  - Creates IPv6 VXLAN tunnel
  - Maps client to IPv6 VXLAN tunnel
  - Encapsulates and decapsulates client traffic into and out of VXLAN tunnel
- **Access Points**:
  - Processes fabric TLV from the controller
  - Creates IPv6 VXLAN tunnel
  - Encapsulates and decapsulates client traffic into and out of VXLAN tunnel

The following figure shows the feature architecture:



Note

For AP to join controller using IPv6 address, ensure that you configure the preferred mode in AP profile as IPv6.

Configure SD-access wireless

Configure SD-access wireless.

Enable SD-Access wireless globally, you need to run the **wireless fabric** configuration command. During SD-Access Wireless provisioning, ensure that L2-VNID value is unique.

Configure default map server (GUI)

Configure the default map server to enable communication between fabric components using the GUI.

Set up the default map server. This action enables the wireless fabric network to properly associate controllers and devices with their respective mapping information.

Procedure

1. Click **Configuration** > **Wireless Plus** > **Fabric** > **Fabric Configuration**.
2. In the **Map Server** section, enter the IP address and preshared key details for Server 1.
3. Optional: Specify the IP address and preshared key details to configure Server 2.
4. Click **Apply**.

The default map server settings are saved, enabling fabric devices to communicate and register with the specified servers.

Configure default map server (CLI)

Use these configuration steps to configure the default map server on your wireless LAN controller and enable seamless device registration and operation in a Fabric network using commands.

Configure a default map server to ensure that wireless Fabric components in your Cisco network can register and operate as part of the control plane using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the default map server.

   Example

   ```
   Device(config)# wireless fabric control-plane map-server-name
   ```

   Here, *map-server-name* defines a pair of map servers.

3. Configure IP address for the default map server.

   Example

   ```
   Device(config-wireless-cp)# ip address ip-address key user-password
   reenter-password
   ```

4. Return to privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

   Example

   ```
   Device(config-wireless-cp)# end
   ```

The default map server is configured. Wireless fabric components can now register with the control-plane infrastructure and exchange location and policy information for network operations.

```
Device# configure terminal
Device(config)# wireless fabric control-plane map-server-name
Device(config-wireless-cp)# ip address 200.0.0.0 key user-password user-password
Device(config-wireless-cp)# end
```

Configure SD-Access Wireless Profile (GUI)

Configure SD-access wireless profile using GUI.

Configure an SD-Access wireless profile to define network parameters for wireless fabric integration.

SD-Access wireless profiles are used to configure wireless fabric settings including VNID and SGT tag parameters for network segmentation.

Procedure

1. Choose **Configuration** > **Wireless** > **Fabric**.
2. On the **Fabric** page, click the **Profiles** tab and click **Add**.
3. In the **Add New Profile** window that is displayed, specify the following parameters:
   - Profile name
   - Description
   - L2 VNID; valid range is between 0 and 16777215
   - SGT tag; valid range is between 2 and 65519
4. Click **Save & Apply to Device**.

The SD-Access wireless profile is configured and applied to the device with the specified parameters.

Configure SD-Access wireless profile (CLI)

Use this task to configure an SD-Access wireless profile using commands.

Configure a wireless profile for SD-Access fabric deployment using commands.

Follow the procedure given below to configure SD-Access wireless profile:

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the SD-Access wireless profile parameters.

Example

```
Device(config)# wireless profile fabric fabric-profile-name
```

3. Configure SGT tag.

   Example

   ```
   Device(config-wireless-fabric)# sgt-tag sgt
   ```

   Here, *sgt* refers to the sgt tag value. The valid range is from 2-65519. The default value is 0.

4. Configure client L2-VNID.

   Example

   ```
   Device(config-wireless-fabric)# client-l2-vnid client-l2-vnid
   ```

   Here, *client-l2-vnid* refers to the client L2-VNID value. The valid range is from 0-16777215.

5. Return to the privileged EXEC mode. Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

   Example

   ```
   Device(config-wireless-fabric)# end
   ```

Configuring Map Server in Site Tag (GUI)

Configure map server in site tag using GUI.

Assign a map server to a site tag using the graphical user interface, enabling the wireless fabric to properly reference the control plane for device management.

Use this task to associate a Fabric control plane with a site tag, which is required for wireless fabric configuration. This enables seamless device communication and site identification within the wireless network.

Before you begin

Ensure that you have configured a control plane at the time of configuring Wireless Fabric.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. On the **Manage Tags** page, click the **Site** tab.
3. Click the name of the site tag.
4. In the **Edit Site Tag** window, choose the Fabric control plane name from the **Control Plane Name** drop-down list.
5. Save the configuration.

The selected site tag is now associated with the specified Fabric control plane, enabling proper device management within the wireless fabric.

Configure map server in site tag (CLI)

Configure map server in site tag using commands.

Configure the fabric control plane map server name within a site tag to establish proper control plane connectivity.

Use this procedure to associate a map server with a site tag for fabric control plane operations. The map server provides centralized mapping services for the fabric infrastructure.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the site tag.

   Example

   ```
   Device(config)# wireless tag site site-tag
   ```

   Example

   ```
   Device(config)# wireless tag site default-site-tag
   ```

   Here, *site-tag* refers to the site tag name.

3. Configure fabric control plane details.

   Example

   ```
   Device(config-site-tag)# fabric control-plane map-server-name
   ```

   Here, *map-server-name* refers to the fabric control plane name associated with the site tag.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-site-tag)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The map server is now configured in the site tag and will provide fabric control plane services for the associated site.

Configure map server per L2-VNID using GUI

Configure map server per L2-VNID using GUI.

Configure the fabric VNID mapping to establish the map server per L2-VNID.

Use this procedure to set up fabric VNID mapping through the GUI interface when you need to establish map server configuration for specific L2-VNIDs.

Procedure

1. Choose **Configuration** > **Wireless** > **Fabric**.
2. On the **Fabric Configuration** page in the **Fabric VNID Mapping** section, click **Add**.
3. In the **Add Client and AP VNID** window, specify a name for the Fabric, L2 VNID value (valid range is from 0 to 4294967295), control plane name.
4. Save the configuration.

The map server per L2-VNID is configured and the fabric VNID mapping is saved.

Configure map server per L2-VNID (CLI)

Configure map server per L2-VNID using commands.

Configure the map server to the VNID map table for proper fabric network operation.

Follow the procedure given below to configure map server in site tag.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the map server to the VNID map table.

   Example

   ```
   Device(config)# wireless fabric name fabric-name l2-vnid l2-vnid-value l3-vnid
    l3-vnid-value ip network-ip subnet-mask control-plane-name control-plane-name
   ```

   Example

   ```
   Device(config)# wireless fabric name fabric_name l2-vnid 2 l3-vnid 2 ip
   122.220.234.0 255.255.0.0 control-plane-name sample-control-plane
   ```

   - *fabric-name* refers to the fabric name.
   - *l2-VNID-value* refers to the L2 VNID value. The valid range is from 0 to 16777215.
   - *l3-VNID-value* refers to the L3 VNID value. The valid range is from 0 to 16777215.
   - *control-plane-name* refers to the control plane name.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

The map server is now configured with the specified L2-VNID and L3-VNID values in the fabric network.

SD-Access wireless verification commands

Lists the commands for verifying SD-Access wireless configurations.

You can verify the SD-Access wireless configurations using these commands:

Table 179: Commands for Verifying SD-Access Wireless

Commands	Description
show wireless fabric summary	Displays the fabric status.
show wireless fabric VNID mapping	Displays all the VNID mapping details.
show wireless profile fabric detailed *fabric_profile_name*	Displays the details of a given fabric profile name.
show AP name *AP_name* **config general**	Displays the general details of the Cisco AP.
show wireless client MAC *MAC_addr* **detail**	Displays the detailed information for a client by MAC address.
show wireless tag site detailed *site_tag*	Displays the detailed parameters for a site tag.

198 Passive Client

Topics:

- Passive clients
- Enable passive client for FlexConnect mode (CLI)
- DHCP broadcast support for workgroup bridges
- Enable passive client on WLAN policy profile (GUI)
- Enable passive client on WLAN policy profile (CLI)
- Enable ARP broadcast on VLAN (GUI)
- Enable ARP broadcast on VLAN (CLI)
- Passive client in fabric deployment
- Verify passive client configuration

Passive clients

Passive Clients refers to wireless devices such as printers and devices configured using a static IP address.

A passive client is a wireless device that

- connects to the wireless network using a static IP address
- does not transmit IP information after associating with an AP, and
- requires additional mechanisms for controllers to learn its IP address (such as ARP requests or DHCP).

Feature history

Table 180: Feature history table for passive clients

Feature name	Release information	Feature description
Support for FlexConnect local switching	Cisco IOS XE 17.12.x	Passive client feature is now supported on FlexConnect local switching mode.
Passive clients	Cisco IOS XE 16.11.1	This feature enables special handling for wireless clients (such as printers and devices with static IPs) that do not transmit IP information after associating with an AP.

Passive clients are wireless devices, such as printers and devices configured using a static IP address. Such clients do no transmit any IP information after associating to an AP. That is why, the controller does not learn their IP address unless they perform the DHCP process.

In the controller, the clients just show up in the **Learn IP** state and get timed out because of the DHCP policy-timeout.

Changes in default behaviour

The Passive Client feature can be enabled on a per WLAN basis. Enabling this feature will change a few default behaviors in order to better accommodate passive clients . These changes include :

- No client will ever timeout in the IP_LEARN phase. The controller will keep on waiting to learn their IP address. Note that the idle timeout remains active and will delete the client entry after the timeout period expiry, if the client remains silent all along.
- ARP coming from the wired side is broadcasted to all the APs, if the controller does not know the client IP address, to ensure that it reaches the passive client. After this, the controller learns the client IP from the ARP response.
- In order to save air time, the controller transforms the ARP broadcast coming from the wired side or from other wireless clients and unicasts them to the wireless client it owns . This is only possible after the controller has learned the MAC-IP binding of its wireless client.
- When the controller enables ARP broadcast, the controller does not transform the ARP broadcasts into unicasts but only forwards the broadcast, thereby wasting air time for other clients (with a frame that is not acknowledgeable and therefore less reliable). This pushes the passive client to respond to the ARP request and therefore every other client benefits from learning the MAC-IP binding of the wireless client.
- When you enable IP MAC binding for clients with static IP address, the controller consumes the first ARP packet to perform IP MAC binding. This packet is not forwarded upstream.

Enable passive client for FlexConnect mode (CLI)

Enable the passive client feature when the device operates in FlexConnect mode to support client devices that do not initiate traffic themselves.

Enable the passive client feature when the device is operating in FlexConnect mode to support client devices that do not initiate traffic themselves (passive clients).

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy policy-profile
   ```

3. Enable passive client.

   Example

   ```
   Device(config-wireless-policy)# passive-client
   ```

   You can use the no form of the command to disable passive client.

4. Disable IP theft detection for static IP passive clients (local and FlexConnect configuration)

   Example

   ```
   Device(config-wireless-policy)# no ip mac-binding
   ```

   This is applicable for only IPv4 and has been designed for overlapping IP addresses.

5. Configure a VLAN and enter VLAN configuration mode.

   Example

   ```
   Device(config)# vlan configuration vlan-id
   ```

6. Broadcast ARP on VLAN, to move passive client to run state (local mode only).

   Example

   ```
   Device(config-vlan-config)# arp broadcast
   ```

7. Configure a FlexConnect profile and enter the FlexConnect profile configuration mode.

Example

```
Device(config)# wireless profile flex flex-profile
```

8. Disable ARP proxy for FlexConnect (FlexConnect mode only).

Example

```
Device(config-wireless-flex-profile)# no arp-caching
```

Passive client for FlexConnect mode is enabled.

```
Device# configure terminal
Device(config)# wireless profile policy policy-profile1
Device(config-wireless-policy)# passive-client
Device(config-wireless-policy)# no ip mac-binding
Device(config)# vlan configuration vlan-id
Device(config-vlan-config)# arp broadcast
Device(config)# wireless profile flex flex-profile1
Device(config-wireless-flex-profile)# no arp-caching
```

DHCP broadcast support for workgroup bridges

Explains how enabling the dhcp broadcast command under the FlexConnect profile ensures DHCP replies are broadcasted, addressing IP assignment issues for devices behind third-party WGBs that do not handle broadcast DHCP requests properly.

Devices placed behind a third-party WGB may fail to obtain an IP address if the WGB does not handle DHCP requests with the broadcast flag set.

- The **dhcp broadcast** command under the FlexConnect profile ensures that DHCP replies are sent as broadcasts instead of being converted to unicasts. This setting helps resolve compatibility issues, especially for some device types.
- In this scenario, you can enable this command to allow devices behind the WGB to receive IP addresses.

Enable passive client on WLAN policy profile (GUI)

Use this task to enable passive client support on a WLAN policy profile, allowing devices that do not transmit packets to remain connected on the WLAN.

Enable passive clients on a specific WLAN policy profile so devices that do not actively transmit packets can stay connected.

Use this task to enable support for passive clients on specific WLAN policy profiles.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Policy** page, click **Add** to open the **Add Policy Profile** page.
2. In the **General** tab, use the slider to enable **Passive Client**.
3. Click **Save & Apply to Device**.

The policy profile is updated to support passive clients on the WLAN.

Enable passive client on WLAN policy profile (CLI)

Use this task to enable the passive client feature on a selected WLAN policy profile using commands. This allows devices that do not actively communicate with the controller to be recognized and managed on the WLAN.

Enable passive client support for devices on a WLAN policy profile using commands.

Use the passive client feature to allow devices that do not actively communicate with the controller to be recognized and managed on the WLAN.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy policy-profile
   ```

3. Enable passive client.

   Example

   ```
   Device(config-wireless-policy)# [no] passive-client
   ```

4. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

The passive client feature is now enabled for the specified WLAN policy profile.

```
Device# configure terminal
Device(config)# wireless profile policy rr-xyz-policy-1
Device(config-wireless-policy)# [no] passive-client
Device(config-wireless-policy)# end
```

Enable ARP broadcast on VLAN (GUI)

Use this task to enable ARP broadcast functionality for VLANs through the GUI, allowing the VLAN to propagate ARP requests.

Enable ARP broadcast capability on a VLAN so ARP requests are properly propagated within the network using the GUI.

Procedure

1. Choose **Configuration** > **Layer2** > **VLAN** page, click **VLAN** tab.
2. Click **Add** to view the **Create VLAN** window.
3. Use the slider to enable **ARP Broadcast**.
4. Click **Save & Apply to Device**.

The selected VLAN now supports ARP broadcasts and propagates ARP requests.

Enable ARP broadcast on VLAN (CLI)

Configure ARP broadcast on a specified VLAN to allow devices within that VLAN to respond to ARP requests using commands.

Enable ARP broadcast on a VLAN so devices within that VLAN can communicate efficiently using ARP using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a VLAN or multiple VLANs to enter the VLAN configuration mode.

   Example

   ```
   Device(config)# vlan configuration vlan-id
   ```

3. Enable ARP broadcast on VLAN.

   Example

   ```
   Device(config-vlan)# [no] arp broadcast
   ```

4. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-vlan)# end
   ```

   You can also press **Ctrl-Z** to exit global configuration mode.

ARP broadcast is successfully enabled for the specified VLAN.

```
Device# configure terminal
Device(config)# vlan configuration 1
```

```
Device(config-vlan)# [no] arp broadcast
Device(config-vlan)# end
```

Passive client in fabric deployment

This topic outlines the characteristics and requirements for supporting passive clients within a fabric network deployment.

A passive client is a client device that

- does not send its own IP address information in packets (such as ARP requests)
- relies on the network infrastructure to discover and maintain its presence, and
- is supported in fabric deployments where special handling is required to ensure connectivity.

You need to enable these features:

- Enable ARP broadcast on VLANs to support passive client discovery.
- Enable LISP multicast to facilitate address learning. For information on LISP multicast, see https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_lisp/configuration/xe-3s/irl-xe-3s-book/irl-lisp-multicast.html.

For information on LISP (Locator ID Separation Protocol), see https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/iproute_lisp/configuration/xe-3s/irl-xe-3s-book/irl-cfg-lisp.html.

Enable broadcast underlay on VLAN

Use this procedure to enable broadcast underlay for a VLAN by configuring a LISP instance and ethernet service on the Fabric Edge Node using commands.

Configure LISP instance and IPv4 service (CLI)

Set up the LISP instance and enable IPv4 services to support mappings from endpoint identifier (EID) to routing locator (RLOC) in your network fabric using commands.

Perform these configuration tasks only on the Fabric Edge Node. Do not perform them from the controller.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   FabricEdge# configure terminal
   ```

2. Enter the LISP configuration mode.

   Example

   ```
   FabricEdge(config)# router lisp
   ```

3. Create a LISP EID instance to group multiple services. The configuration applied under this instance ID applies to all services in the group.

Example

```
FabricEdge(config-router-lisp)# instance-id instance
```

4. Enable Layer 3 network services for the IPv4 address family. Enter the service submode.

   Example

   ```
   FabricEdge(config-router-lisp-instance)# service ipv4
   ```

5. Configure EID to RLOC mapping relationship.

   Example

   ```
   FabricEdge(config-router-lisp-instance-dynamic-eid)# database-mapping eid
   locator-set rloc-name
   ```

6. Generate a static map request for the destination EID.

   Example

   ```
   FabricEdge(config-router-lisp-instance-service)# map-cache destination-eid
   map-request
   ```

7. Exit the service submode.

   Example

   ```
   FabricEdge(config-router-lisp-instance-service)# exit-service-ipv4
   ```

8. Exit the instance submode.

   Example

   ```
   FabricEdge(config-router-lisp-instance)# exit-instance-id
   ```

A system with a LISP instance and IPv4 service is now configured.

```
FabricEdge# configure terminal
FabricEdge(config)# router lisp
FabricEdge(config-router-lisp)# instance-id 3
FabricEdge(config-router-lisp-instance)# service ipv4
FabricEdge(config-router-lisp-instance-dynamic-eid)# database-mapping
66.66.66.64/32 locator-set rloc1
FabricEdge(config-router-lisp-instance-service)# map-cache 0.0.0.0/0 map-request
FabricEdge(config-router-lisp-instance-service)# exit-service-ipv4
FabricEdge(config-router-lisp-instance)# exit-instance-id
```

Configure Ethernet service and enable broadcast underlay (CLI)

Set up the LISP instance and enable IPv4 services to support endpoint identifier (EID) to routing locator (RLOC) mappings in the fabric.

Procedure

1. Create a LISP EID instance to group multiple services.

   Example

   ```
   FabricEdge(config-router-lisp)# instance-id instance
   ```

2. Enable Layer 2 network services and enter service submode.

   Example

   ```
   FabricEdge(config-router-lisp-instance)# service ethernet
   ```

3. Associate the LISP instance ID with a VLAN. This VLAN provides reachability to the endpoint identifier address space.

   Example

   ```
   FabricEdge(config-router-lisp-instance-service)# eid-table vlan vlan-number
   ```

4. Specify the multicast group that the underlay uses to carry broadcast traffic for the overlay Layer 2 network.

   Example

   ```
   FabricEdge(config-router-lisp-instance-service)# broadcast-underlay
   multicast-group
   ```

5. Exit the service submode.

   Example

   ```
   FabricEdge(config-router-lisp-instance-service)# exit-service-ethernet
   ```

6. Exit the instance submode.

   Example

   ```
   FabricEdge(config-router-lisp-instance)# exit-instance-id
   ```

The Ethernet service is enabled and the broadcast underlay support is configured for the specified VLAN and multicast group.

```
FabricEdge(config-router-lisp)# instance-id 101
FabricEdge(config-router-lisp-instance)# service ethernet
FabricEdge(config-router-lisp-instance-service)# eid-table vlan 101
FabricEdge(config-router-lisp-instance-service)# broadcast-underlay 239.0.0.1
FabricEdge(config-router-lisp-instance-service)# exit-service-ethernet
FabricEdge(config-router-lisp-instance)# exit-instance-id
```

Enable ARP flooding

Learn how to enable ARP flooding for IPv4 and Ethernet services on the Fabric Edge Node using commands.

Enable ARP flooding for IPv4 service (CLI)

Enable ARP flooding on the Fabric Edge Node for IPv4.

Perform these configuration tasks from Fabric Edge Node. You cannot perform them from your controller.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   FabricEdge# configure terminal
   ```

2. Enter the LISP configuration mode.

   Example

   ```
   FabricEdge(config)# router lisp
   ```

3. Create a LISP EID instance to group multiple services. This configuration applies to all services grouped under the instance-id.

   Example

   ```
   FabricEdge(config-router-lisp)# instance-id instance
   ```

4. Enable Layer 3 network services for the IPv4 address family and enter service submode.

   Example

   ```
   FabricEdge(config-router-lisp-instance)# service ipv4
   ```

5. Configure EID to RLOC mapping relationship.

   Example

   ```
   FabricEdge(config-router-lisp-instance-dynamic-eid)# database-mapping eid
   locator-set rloc1
   ```

6. Generate a static map request for the destination EID.

   Example

   ```
   FabricEdge(config-router-lisp-instance-service)# map-cache destination-eid
   map-request
   ```

7. Exit the service submode.

   Example

   ```
   FabricEdge(config-router-lisp-instance-service)# exit-service-ipv4
   ```

8. Exit the instance submode.

Example

```
FabricEdge(config-router-lisp-instance)# exit-instance-id
```

You have enabled ARP flooding on the Fabric Edge Node for IPv4.

```
FabricEdge# configure terminal
FabricEdge(config)# router lisp
FabricEdge(config-router-lisp)# instance-id instance
FabricEdge(config-router-lisp-instance)# service ipv4
FabricEdge(config-router-lisp-instance-dynamic-eid)# database-mapping
66.66.66.64/32 locator-set rloc1
FabricEdge(config-router-lisp-instance-service)# map-cache 0.0.0.0/0 map-request
FabricEdge(config-router-lisp-instance-service)# exit-service-ipv4
FabricEdge(config-router-lisp-instance)# exit-instance-id
```

What to do next

Enable ARP flooding for Ethernet service.

Enable ARP flooding for Ethernet service (CLI)

Enable ARP flooding for the Ethernet service on the Fabric Edge Node.

Procedure

---

1. Create a LISP EID instance, which groups multiple services.

   Example

   ```
   FabricEdge(config-router-lisp)# instance-id instance
   ```

2. Enable Layer 2 network services and enter service submode.

   Example

   ```
   FabricEdge(config-router-lisp-instance)# service ethernet
   ```

3. Associate the LISP instance ID you configured earlier with a VLAN. The VLAN provides access to the endpoint identifier address space.

   Example

   ```
   FabricEdge(config-router-lisp-instance-service)# eid-table vlan vlan-number
   ```

4. Enable ARP flooding.

   Example

   ```
   FabricEdge(config-router-lisp-instance-service)# flood arp-nd
   ```

5. Configure EID to RLOC mapping relationship.

Example

```
FabricEdge(config-router-lisp-instance-service)# database-mapping mac
locator-set rloc1
```

6. Exit the service submode.

Example

```
FabricEdge(config-router-lisp-instance-service)# exit-service-ethernet
```

7. Exit the instance submode.

Example

```
FabricEdge(config-router-lisp-instance)# exit-instance-id
```

You have enabled ARP flooding on the Fabric Edge Node for Ethernet service.

```
FabricEdge(config-router-lisp)# instance-id 101
FabricEdge(config-router-lisp-instance)# service ethernet
FabricEdge(config-router-lisp-instance-service)# eid-table vlan 101
FabricEdge(config-router-lisp-instance-service)# flood arp-nd
FabricEdge(config-router-lisp-instance-service)# database-mapping mac
locator-set rloc1
FabricEdge(config-router-lisp-instance-service)# exit-service-ethernet
FabricEdge(config-router-lisp-instance)# exit-instance-id
```

Verify passive client configuration

Verify passive client configuration.

To verify the status of the passive client, use this command:

```
Device# show wireless profile policy detailed sample-profile-policy
Policy Profile Name           : sample-profile-policy
Description                   : sample-policy
Status                        : ENABLED
VLAN                          : 20
Client count                  : 0
Passive Client                : ENABLED    <--------------------
WLAN Switching Policy
Central Switching           : ENABLED
Central Authentication      : ENABLED
Central DHCP                : DISABLED
Override DNS                : DISABLED
Override NAT PAT            : DISABLED
Central Assoc               : DISABLED
.
.
.
```

To verify VLANs that have ARP broadcast enabled, use this command:

```
Device# show platform software arp broadcast

Arp broadcast is enabled on vlans:
20
```

199 Fabric in a Box with External Fabric Edge

Topics:

- Fabric in a Box with external Fabric Edge

Describes the role of external fabric edge nodes in integrated wireless fabric operation, including their relationship with fabric topology, policy, site, WLAN, AP profiles, and LISP/map-server integration.

Fabric in a Box with external Fabric Edge

Explains the characteristics of Fabric in a Box topologies that incorporate external Fabric Edge Nodes for enhanced network scalability and wireless reach.

A Fabric in a Box topology is a network architecture that

- combines border node, control plane, Fabric Edge, and wireless controller in a single device
- supports external Fabric Edge Nodes to extend network capabilities, and
- enables seamless integration and roaming of APs and clients across both internal and external nodes.

Beginning with Cisco IOS XE Amsterdam 17.2.1, the Fabric in a Box (FiaB) topology supports external Fabric Edge Nodes. External Fabric Edge Nodes increase port density and extend wireless reach by adding more APs.

Configure a Fabric profile (CLI)

Configure a Fabric profile using commands to define wireless Fabric settings and client parameters.

Set up a wireless Fabric profile, assign a client L2-VNID, and add a description for organizational clarity using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the wireless Fabric profile parameters.

   Example

   ```
   Device(config)# wireless profile fabric fabric-profile-name
   ```

3. Configure client L2-VNID.

   Example

   ```
   Device(config-wireless-fabric)# client-l2-vnid client-l2-vnid
   ```

   Here, *client-l2-vnid* refers to the client L2-VNID value. The valid range is from 0 to 16777215.

4. Add a description for the Fabric profile.

   Example

   ```
   Device(config-wireless-fabric)# description description
   ```

5. Return to the privileged EXEC mode.

Example

```
Device(config-wireless-fabric)# end
```

Configure a policy profile (CLI)

Configure a wireless policy profile using commands to enable fabric deployment features and local switching.

Set up a wireless policy profile via commands to define AAA policy overrides, DHCP mode, local switching, and apply Fabric profiles for wireless networks.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure wireless policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy profile-policy
   ```

   > **Note**
   >
   > In Fabric deployments, local mode, local authentication, and local association are not supported.

3. Configure local DHCP mode, where the DHCP is performed in an AP.

   Example

   ```
   Device(config-wireless-policy)# no central dhcp
   ```

4. Configure a WLAN for local switching.

   Example

   ```
   Device(config-wireless-policy)# no central switching
   ```

5. Apply the Fabric profile.

   Example

   ```
   Device(config-wireless-fabric)# fabric fabric-name
   ```

6. Enable the policy profile.

Example

```
Device(config-wireless-fabric)# no shutdown
```

7. Return to the privileged EXEC mode.

Example

```
Device(config-wireless-fabric)# end
```

Configure a site tag (CLI)

Use this procedure to configure a site tag for wireless APs using commands.

Set up a site tag and assign an AP profile to manage wireless APs using commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure site tag and enter the site tag configuration mode.

Example

```
Device(config)# wireless tag site site-tag
```

3. Assign an AP profile to the wireless site.

Example

```
Device(config-site-tag)# ap-profile ap-profile-name
```

4. Add a description to the AP profile.

Example

```
Device(config-site-tag)# description description
```

5. Return to the privileged EXEC mode.

Example

```
Device(config-site-tag)# end
```

Configure a WLAN (CLI)

Configure a WLAN using commands.

Configure a WLAN on your device using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a WLAN and enter the WLAN configuration submode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id SSID-name
   ```

3. Enable the WLAN.

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

Configure a policy tag (CLI)

Configure policy tags on a device using the commands to map policy profiles to WLAN profiles.

Set up a policy tag, and map a policy profile to a WLAN profile using the device commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure policy tag and enter the policy tag configuration mode.

   Example

   ```
   Device(config)# wireless tag policy policy-tag-name
   ```

3. Map a policy profile to a WLAN profile.

Example

```
Device(config-policy-tag)# wlan wlan-name policy profile-policy-name
```

4. Return to the privileged EXEC mode.

Example

```
Device(config-site-tag)# end
```

Configure an AP profile (CLI)

Configure an AP profile on your device using command-line interface commands.

Set up an AP profile for managing APs with specific configuration parameters using commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure an AP profile and enter the AP profile configuration mode.

Example

```
Device(config)# ap profile ap-profile-name
```

3. Enter the AP configuration mode.

Example

```
Device(config-ap-profile)# ap ap-ether-mac
```

4. Specify the policy tag that is to be attached to the AP.

Example

```
Device(config-ap-profile)# policy-tag policy-tag
```

5. Return to the privileged EXEC mode.

Example

```
Device(config-ap-profile)# end
```

Configure map server and AP subnet (CLI)

Configure the map server and associated AP subnet using commands.

Enable SD-Access wireless features by configuring the map server and setting up AP subnet Layer 2 and Layer 3 VNIDs using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable SD-Access wireless globally.

   Example

   ```
   Device(config)# wireless fabric
   ```

3. Configure AP subnet Layer 2 and Layer 3 VNIDs.

   Example

   ```
   Device(config)# wireless fabric name name l2-vnid l2-vnid-value l3-vnid
   l3-vnid-value ip network-ip subnet-mask
   ```

4. Define client Layer 2 VNID AAA override.

   Example

   ```
   Device(config)# wireless fabric name name l2-vnid l2-vnid-value
   ```

5. Configure the control plane name.

   Example

   ```
   Device(config)# wireless fabric control-plane name
   ```

6. Configure the map server IP address and authentication key shared with the map server.

   Example

   ```
   Device((config-wireless-cp)# ip address ip-address key shared-key
   ```

7. Return to the privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

Alternatively, you can also press *Ctrl-Z* to exit global configuration mode.

Configure Fabric on FiaB node (CLI)

Configure the LISP Fabric and related networking features on a FiaB node using commands.

Enable Layer 2 and Layer 3 network services on a FiaB node, configure address resolution, and facilitate secure locator discovery using commands.

Procedure

1. Enter the global configuration mode. Enter the LISP configuration mode. Associate a default Virtual Routing and Forwarding (VRF) table through which the routing locator address space is reachable to a router Locator ID Separation Protocol (LISP) instantiation. Specify a named locator set and enter the LISP locator-set configuration mode. Specify an IP address of loopback or other egress tunnel router (ETR) interface. Exit the LISP locator-set configuration mode. Specify an existing locator set and enter the LISP locator-set configuration mode. Configure a locator address by creating a locator entry.

   Example

   ```
   FiaB# configure terminal
   ```

   ```
   FiaB(config)# router lisp
   ```

   ```
   FiaB(config-router-lisp)# locator-table default
   ```

   ```
   FiaB(config-router-lisp)# locator-set locator-set-name
   ```

   ```
   FiaB(config-router-lisp-locator-set)# ip-address
   ```

   ```
   FiaB(config-router-lisp-locator-set)# exit-locator-set
   ```

   ```
   FiaB(config-router-lisp)# locator-set rloc_loopback
   ```

2. Configure the ETR to auto discover the locators registered by other xTRs. (Ingress tunnel router (ITR) and an ETR are known as an xTR.). Exit LISP locator-set configuration mode. Enable Layer 3 network services for the IPv4 address family and enter the service submode. Configure VXLAN as encapsulation type for data packets. Configure map resolver address for sending map requests. Configure the map server for ETR registration. Configure a LISP ETR.

Example

```
FiaB(config-router-lisp-locator-set)# IPv4-interface interface
```

```
FiaB(config-router-lisp-locator-set)# auto-discover-rlocs
```

```
FiaB(config-router-lisp-locator-set)# exit-locator-set
```

```
FiaB(config-router-lisp)# service ipv4
```

```
FiaB(config-lisp-srv-ipv4)# encapsulation vxlan
```

```
FiaB(config-lisp-srv-ipv4)# itr map-resolver map-resolver-address
```

```
FiaB(config-lisp-srv-ipv4)# etr map-server map-server-address key key-type
authentication-key
```

3. Enable security group tag propagation in LISP-encapsulated traffic. Remove the address family-specific map cache configuration. Enable the Proxy Ingress Tunnel Router (PITR) functionality and specify the address to use when LISP encapsulating packets to LISP sites. Configure a LISP map server. Configure a LISP map resolver. Export table entries into the map cache, with the action set to send-map-request. Export LISP site registrations to the routing information base (RIB).

   Example

   ```
   FiaB(config-lisp-srv-ipv4)# etr
   ```

   ```
   FiaB(config-lisp-srv-ipv4)# sgt
   ```

   ```
   FiaB(config-lisp-srv-ipv4)# no map-cache away-eids send-map-request
   ```

   ```
   FiaB(config-lisp-srv-ipv4)# proxy-itr ip-address
   ```

   ```
   FiaB(config-lisp-srv-ipv4)# map-server
   ```

   ```
   FiaB(config-lisp-srv-ipv4)# map-resolver
   ```

   ```
   FiaB(config-lisp-srv-ipv4)# map-cache away-eids send-map-request
   ```

4. Configure LISP installed routes of type site registrations. Install the map cache to a map request for site registrations. Exit LISP service-ipv4 configuration mode. Select service type as Ethernet and enter the service submode. Configure the maximum number of dynamic local endpoint identifier (EID) prefix database entries. Configure the map-resolver address for sending map requests.

Example

```
FiaB(config-lisp-srv-ipv4)# route-export site-registrations
```

```
FiaB(config-lisp-srv-ipv4)# distance site-registrations number
```

```
FiaB(config-lisp-srv-ipv4)# map-cache site-registration
```

```
FiaB(config-lisp-srv-ipv4)# exit-service-ipv4
```

```
FiaB(config-router-lisp)# service ethernet
```

```
FiaB(config-lisp-srv-eth)# database-mapping limit dynamic limit
```

```
FiaB(config-lisp-srv-eth)# itr map-resolver map-resolver-address
```

5. Enable the LISP ITR functionality. Configure a map server for ETR registration. Enable the LISP ETR functionality. Enable the LISP map server functionality. Enable the LISP map resolver functionality. Exit LISP service-ethernet configuration mode. Create a LISP EID instance to group multiple services.

Example

```
FiaB(config-lisp-srv-eth)# itr
```

```
FiaB(config-lisp-srv-eth)# etr map-server map-server-address key key-type
authentication-key
```

```
FiaB(config-lisp-srv-eth)# etr
```

```
FiaB(config-lisp-srv-eth)# map-server
```

```
FiaB(config-lisp-srv-eth)# map-resolver
```

```
FiaB(config-lisp-srv-eth)# exit-service-ethernet
```

```
FiaB(config-router-lisp)# instance-id instance
```

6. Configure the parameters for probing of remote routing locators (RLOCs). Configure a dynamic EID and enter the dynamic EID configuration mode. Configure EID prefix and locator-set for dynamic EID. Exit LISP dynamic-eid configuration mode. Exit LISP instance-id configuration mode. Create a LISP EID instance to group multiple services. Configure parameters for probing remote RLOCs.

Example

```
FiaB(config-lisp-inst)# remote-rloc-probe on-route-change
```

```
FiaB(config-lisp-inst)# dynamic-eid dynamic-eid-name
```

```
FiaB(config-router-lisp-dynamic-eid)# database-mapping eid locator-set
rloc_loopback
```

```
FiaB(config-router-lisp-dynamic-eid)# exit-dynamic-eid
```

```
FiaB(config-router-lisp-instance)# exit-instance-id
```

```
FiaB(config-router-lisp)# instance-id instance
```

```
FiaB(config-lisp-inst)# remote-rloc-probe on-route-change
```

7. Enable Layer 2 network services and enter the service submode. Bind an EID table to VLAN. Configure an address family-specific local EID prefixes database. Exit LISP service-ethernet configuration mode. Exit LISP instance-id configuration mode. Configure a map server with open passive TCP sockets to listen for incoming connections.

   Example

   ```
   FiaB(config-lisp-inst)# service ethernet
   ```

   ```
   FiaB(config-lisp-inst-srv-eth)# eid-table vlan vlan-number
   ```

   ```
   FiaB(config-lisp-inst-srv-eth)# database-mapping mac locator-set rloc_loopbac
   ```

   ```
   FiaB(config-lisp-inst-srv-eth)# exit-service-ethernet
   ```

   ```
   FiaB(config-lisp-inst)# exit-instance-id
   ```

   ```
   FiaB(config-router-lisp)# map-server session passive-open server
   ```

8. Configure a LISP site on a map server. Specify a description text for the LISP site. Configure the authentication key used by the LISP site.

   Example

   ```
   FiaB(config-router-lisp)# site site-name
   ```

   ```
   FiaB(config-router-lisp-site)# description map-server-description
   ```

   ```
   FiaB(config-router-lisp-site)# authentication-key key
   ```

9. Accept registrations, if any, for Layer 2 EID records. Exit LISP site configuration mode. Configure the IPv4 locator address of the LISP. Configure the IPv4 source locator address of the interface. Exit LISP router-lisp configuration mode.

Example

```
FiaB(config-router-lisp-site)# eid-record instance-id instance-id address
accept-more-specifics
```

```
FiaB(config-router-lisp-site)# eid-record instance-id instance-id any-mac
```

```
FiaB(config-router-lisp-site)# exit-site
```

```
FiaB(config-router-lisp)# ipv4 locator reachability exclude-default
```

```
FiaB(config-router-lisp)# ipv4 source-locator interface-name
```

```
FiaB(config-router-lisp)# exit-router-lisp
```

Configure a Fabric Edge Node (CLI)

Configure a Fabric Edge Node using commands to enable LISP, interface settings, and network services necessary for its operation within a fabric network.

Set up a Fabric Edge Node with essential configurations for LISP operation, routing, and interface management using commands.

> **Note**
>
> You can perform these configuration tasks only from Fabric Edge Node, and not from your controller.

Procedure

1. Enter the global configuration mode. Enter the LISP configuration mode. Associate a default VRF table through which the routing locator address space is reachable to a router LISP instantiation. Specify a named locator set and enters LISP locator-set configuration mode. Configure the IPv4 address of the interface as locator. Exit the LISP locator-set configuration mode. Exit the LISP router-lisp configuration mode.

Example

```
FabricEdge# configure terminal
```

```
FabricEdge(config)# router lisp
```

```
FabricEdge(config-router-lisp)# locator-table default
```

```
FabricEdge(config-router-lisp)# locator-set rloc_loopback
```

```
FabricEdge(config-router-lisp-locator-set)# IPv4-interface interface-num
priority priority weight weight
```

```
FabricEdge(config-router-lisp-locator-set)# exit-locator-set
```

```
FabricEdge(config-router-lisp-)# exit-router-lisp
```

2. Configure an interface. Specify a description text for the interface. Set an interface MAC address manually. Configure an IP address for the interface. Specify a destination address for UDP broadcasts. Disable sending of ICMP redirect messages. Enable fast-switching cache for outgoing packets on the same interface.

Example

```
FabricEdge(config)# interface Vlan interface-num
```

```
FabricEdge(config-if)# description description
```

```
FabricEdge(config-if)# mac-address mac-address
```

```
FabricEdge(config-if)# ip address ip-address mask
```

```
FabricEdge(config-if)# ip helper-address ip-address
```

```
FabricEdge(config-if)# no ip redirects
```

```
FabricEdge(config-if)# ip route-cache same-interface
```

3. Remove liveness test on dynamic EID discovered on this interface. Allow EID mobility on the interface. Exit from the interface configuration mode. Enter the LISP configuration mode. Specify a locator set and enter the LISP locator-set configuration mode. Exit the LISP locator-set configuration mode. Enable Layer 3 network services for the IPv4 address family and enter the service submode.

Example

```
FabricEdge(config-if)# no lisp mobility liveness test
```

```
FabricEdge(config-if)# lisp mobility dynamic-eid-name
```

```
FabricEdge(config-if)# exit
```

```
FabricEdge(config)# router lisp
```

```
FabricEdge(config-router-lisp)# locator-set locator-set-name
```

```
FabricEdge(config-router-lisp-locator-set)# exit-locator-set
```

```
FabricEdge(config-router-lisp)# service ipv4
```

4. Configure the loopback IP address of the Proxy Egress Tunnel Router (PETR). Select the encapsulation type as VXLAN for data packets. Configure the map resolver address for sending map requests. Configure the map server for ETR registration. Configure the locator address of the LISP map server and the authentication key that this router, acting as a LISP ETR, will use to register with the LISP mapping system. Configure a LISP Egress Tunnel Router (ETR).

Example

```
FabricEdge(config-lisp-srv-ipv4)# use-petr ip-address
```

```
FabricEdge(config-lisp-srv-ipv4)# encapsulation vxlan
```

```
FabricEdge(config-lisp-srv-ipv4)# itr map-resolver map-resolver-address
```

```
FabricEdge(config-lisp-srv-ipv4)# etr map-servermap-server-address key key-type
 authentication-key
```

```
FabricEdge(config-lisp-srv-ipv4)# etr map-servermap-server-address proxy-reply
 authentication-key
```

```
FabricEdge(config-lisp-srv-ipv4)# etr
```

5. Enable security group tag propagation in LISP encapsulated traffic. Remove the address family-specific map cache configuration. Enable the Proxy Ingress Tunnel Router (PITR) functionality and specify the address to use when LISP encapsulating packets to LISP sites. Exit LISP service-ipv4 configuration mode. Select the service type as Ethernet. Configure the map-resolver address for sending map requests.

Example

```
FabricEdge(config-lisp-srv-ipv4)# sgt
```

```
FabricEdge(config-lisp-srv-ipv4)# no map-cache away-eids send-map-request
```

```
FabricEdge(config-lisp-srv-ipv4)# proxy-itr ip-address
```

```
FabricEdge(config-lisp-srv-ipv4)# exit-service-ipv4
```

```
FabricEdge(config-router-lisp)# service ethernet
```

```
FabricEdge(config-lisp-srv-eth)# itr map-resolver map-resolver-address
```

6. Enable the LISP ITR functionality. Configure the map server for ETR registration. Enable the LISP ETR functionality. Exit the LISP service-ethernet configuration mode. Create a LISP EID instance to group multiple services. Configure the parameters for probing remote Routing locators (RLOCs).

   Example

```
FabricEdge(config-lisp-srv-eth)# itr
```

```
FabricEdge(config-lisp-srv-eth)# etr map-server map-server-address key key-type
 authentication-key
```

```
FabricEdge(config-lisp-srv-eth)# etr
```

```
FabricEdge(config-lisp-srv-eth)# exit-service-ethernet
```

```
FabricEdge(config-router-lisp)# instance-id instance
```

```
FabricEdge(config-lisp-inst)# remote-rloc-probe on-route-change
```

7. Configure a dynamic EID and enter the dynamic EID configuration mode. Configure the EID prefix and locator set for the dynamic EID. Exit dynamic instance submode. Select service type as IPv4. Bind an EID table. Exit the LISP service-ipv4 configuration mode. Exit the LISP instance-id configuration mode.

Example

```
FabricEdge(config-lisp-inst)# dynamic-eid dynamic-eid-name
```

```
FabricEdge(config-router-lisp-dynamic-eid)# database-mapping eid locator-set
 rloc_loopback
```

```
FabricEdge(config-router-lisp-dynamic-eid)# exit-instance-id
```

```
FabricEdge(config-lisp-inst)# service ipv4
```

```
FabricEdge(config-lisp-inst-srv-ipv4)# eid-table default
```

```
FabricEdge(config-lisp-inst-srv-ipv4)# exit-service-ipv4
```

```
FabricEdge(config-lisp-inst)# exit-instance-id
```

8. Select service type as IPv4. Export away table entries into the map cache, with the action set to send-map-request. Exit LISP service-ipv4 configuration mode. Create a LISP EID instance to group multiple services. Configure parameters for probing remote RLOCs. Enable Layer 2 network services and enter the service submode. Bind an EID table to VLAN.

Example

```
FabricEdge(config-router-lisp)# service ipv4
```

```
FabricEdge(config-lisp-srv-ipv4)# map-cache away-eids map-request
```

```
FabricEdge(config-lisp-srv-ipv4)# exit-service-ipv4
```

```
FabricEdge(config-router-lisp)# instance-id instance
```

```
FabricEdge(config-lisp-inst)# remote-rloc-probe on-route-change
```

```
FabricEdge(config-lisp-inst)# service ethernet
```

```
FabricEdge(config-lisp-inst-srv-eth)# eid-table vlan vlan-number
```

9. Configure address family-specific local EID prefixes database. Exit LISP service-ethernet configuration mode. Exit from LISP instance-id configuration mode. Configure the IPv4 locator address of the LISP. Configure the IPv4 source locator address of the interface. Exit the LISP router-lisp configuration mode.

Example

```
FabricEdge(config-lisp-inst-srv-eth)# database-mapping mac locator-set
rloc_loopbac
```

```
FabricEdge(config-lisp-inst-srv-eth)# exit-service-ethernet
```

```
FabricEdge(config-lisp-inst)# exit-instance-id
```

```
FabricEdge(config-router-lisp)# ipv4 locator reachability minimum-mask-length
 length
```

```
FabricEdge(config-router-lisp)# ipv4 source-locator interface-name
```

```
FabricEdge(config-router-lisp)# exit-router-lisp
```

Verify Fabric configuration

Lists commands and sample outputs to verify LISP configuration, operational status, and site registration in Cisco Fabric environments.

Use these commands to verify the Fabric configuration.

To verify the LISP configuration on a device, use this command:

```
FabricEdge# show running-config | section router lisp

router lisp
locator-table default
locator-set default
exit-locator-set
 !
locator-set rloc_loopback
IPv4-interface Loopback0 priority 10 weight 10
exit-locator-set
 !
locator default-set rloc_loopback
service ipv4
encapsulation vxlan
itr map-resolver 21.21.21.21
itr
etr map-server 21.21.21.21 key tasman
etr map-server 21.21.21.21 proxy-reply
etr
use-petr 21.21.21.21 priority 1 weight 100
exit-service-ipv4
 !
service ethernet
itr map-resolver 5.5.5.5
itr map-resolver 21.21.21.21
itr
etr map-server 21.21.21.21 key tasman
etr map-server 21.21.21.21 proxy-reply
etr
exit-service-ethernet
 !
```

```
instance-id 0
loc-reach-algorithm lsb-reports ignore
dynamic-eid eid_10_56_25
database-mapping 10.56.25.0/24 locator-set rloc_loopback
exit-dynamic-eid
  !
service ipv4
eid-table default
database-mapping 26.26.26.26/32 locator-set rloc_loopback
exit-service-ipv4
  !
exit-instance-id
 !
instance-id 1
service ethernet
eid-table vlan 25
flood arp-nd
database-mapping mac locator-set rloc_loopback
exit-service-ethernet
  !
exit-instance-id
 !
instance-id 101
service ipv4
exit-service-ipv4
  !
exit-instance-id
 !
instance-id 8188
  exit-instance-id
 !
loc-reach-algorithm lsb-reports ignore
exit-router-lisp
```

To verify the operational status of LISP as configured on a device, use this command:

```
FabricEdge# show ip lisp

Information applicable to all EID instances:
Router-lisp ID:                             0
Locator table:                              default
Ingress Tunnel Router (ITR):                enabled
Egress Tunnel Router (ETR):                 enabled
Proxy-ITR Router (PITR):                    disabled
Proxy-ETR Router (PETR):                    disabled
NAT-traversal Router (NAT-RTR):             disabled
Mobility First-Hop Router:                  disabled
Map Server (MS):                            disabled
Map Resolver (MR):                          disabled
Mr-use-petr:                                disabled
Delegated Database Tree (DDT):              disabled
Publication-Subscription:                   enabled
Publisher(s):                              *** NOT FOUND ***
ITR Map-Resolver(s):                        21.21.21.21
ETR Map-Server(s):                          21.21.21.21
xTR-ID:
0xD89893A6-0x98749B2C-0x89810431-0x92F33C9C
site-ID:                                    unspecified
ITR local RLOC (last resort):               *** NOT FOUND ***
ITR use proxy ETR RLOC(Encap IID):          21.21.21.21
ITR Solicit Map Request (SMR):              accept and process
Max SMRs per map-cache entry:              8 more specifics
```

```
Multiple SMR suppression time:          20 secs
ETR accept mapping data:                  disabled, verify disabled
ETR map-cache TTL:                        1d00h
Locator Status Algorithms:
RLOC-probe algorithm:                   disabled
RLOC-probe on route change:             N/A (periodic probing disabled)
RLOC-probe on member change:            disabled
LSB reports:                            ignore
IPv4 RLOC minimum mask length:          /0
IPv6 RLOC minimum mask length:          /0
Map-cache:
Map-cache limit:                        32768
Map-cache activity check period:        60 secs
Persistent map-cache:                   disabled
Source locator configuration:
GigabitEthernet1/0/1: 24.24.24.24 (Loopback0)
Vlan25: 24.24.24.24 (Loopback0)
Database:
Dynamic database mapping limit:         25000
```

To verify the operational status of the map cache on a device configured as an ITR or PITR, use this command:

```
FabricEdge# show lisp instance-id iid ipv4 map-cache

LISP IPv4 Mapping Cache for EID-table default (IID 0), 5 entries
0.0.0.0/0, uptime: 2w5d, expires: never, via static-send-map-request
  Encapsulating to proxy ETR
10.56.25.0/24, uptime: 2w0d, expires: never, via dynamic-EID, send-map-request

  Encapsulating to proxy ETR
10.56.25.25/32, uptime: 2w5d, expires: 23:10:06, via map-reply, complete
  Locator      Uptime    State  Pri/Wgt     Encap-IID
  21.21.21.21  2w5d      up       0/0         -
22.0.0.0/8, uptime: 2w5d, expires: 00:04:54, via map-reply, forward-native
  Encapsulating to proxy ETR
26.26.26.26/32, uptime: 09:48:33, expires: 14:11:26, via map-reply, self,
complete
  Locator      Uptime    State     Pri/Wgt     Encap-IID
  24.24.24.24  09:48:33  up, self   50/50        -
```

To verify the operational status of the database mapping on a device configured as an ETR, use this command:

```
FabricEdge# show lisp instance-id iid ipv4 database

LISP ETR IPv4 Mapping Database for EID-table default (IID 0), LSBs: 0x1
Entries total 3, no-route 0, inactive 0
10.56.25.27/32, dynamic-eid eid_10_56_25, skip reg, inherited from default
locator-set rloc_loopback
  Uptime: 00:25:11, Last-change: 00:25:11
  Domain-ID: unset
  Locator      Pri/Wgt  Source     State
  24.24.24.24   10/10   cfg-intf   site-self, reachable
10.56.25.67/32, dynamic-eid eid_10_56_25, inherited from default locator-set
rloc_loopback
  Uptime: 00:24:47, Last-change: 00:24:47
  Domain-ID: unset
  Locator      Pri/Wgt  Source     State
  24.24.24.24   10/10   cfg-intf   site-self, reachable
26.26.26.26/32, locator-set rloc_loopback
  Uptime: 2w5d, Last-change: 00:50:36
  Domain-ID: unset
```

```
  Locator      Pri/Wgt  Source     State
  24.24.24.24   10/10   cfg-intf   site-self, reachable
```

To verify the configured LISP sites on a LISP map server, use this command:

```
FabricEdge# show lisp instance-id iid ipv4 server

LISP Site Registration Information
* = Some locators are down or unreachable
# = Some registrations are sourced by reliable transport
Site Name      Last      Up     Who Last             Inst     EID Prefix
               Register         Registered           ID
eca            never     no     --                   0        10.56.25.0/24
               04:52:53  yes#   21.21.21.21:40875    0        10.56.25.25/32
               04:07:09  yes#   27.27.27.27:24949    0        10.56.25.64/32
               03:21:16  yes#   24.24.24.24:23672    0        10.56.25.67/32
               04:52:53  yes#   21.21.21.21:40875    0        23.23.23.23/32
               03:47:04  yes#   24.24.24.24:23672    0        26.26.26.26/32
               2w0d      yes#   27.27.27.27:24949    0        29.29.29.29/32
site_uci       never     no     --                   4097     0.0.0.0/0
```

To verify the operational status of LISP sites, use this command in FiaB node:

```
FabricEdge# show lisp instance-id 1 ethernet server

==================================================
Output for router lisp 0 instance-id 1
==================================================
LISP Site Registration Information
==================================================
Output for router lisp 0 instance-id 1
==================================================
LISP Site Registration Information
* = Some locators are down or unreachable
# = Some registrations are sourced by reliable transport
Site Name      Last      Up     Who Last             Inst     EID Prefix
               Register         Registered           ID
eca            never     no     --                   1        any-mac
              04:10:37  yes#   27.27.27.27:24949    1       00b0.e19c.2578/48

              04:09:20  yes#   22.22.22.22:64083    1       00b0.e19c.fc40/48

              03:24:52  yes#   24.24.24.24:23672    1       dcce.c130.0b70/48

              03:23:39  yes#   22.22.22.22:64083    1       dcce.c130.9820/48
```

To verify the operational status of LISP sites, use this command in FiaB node:

```
FabricEdge# show lisp instance-id 0 ipv4 server

LISP Site Registration Information
* = Some locators are down or unreachable
# = Some registrations are sourced by reliable transport
Site Name      Last      Up     Who Last             Inst     EID Prefix
               Register         Registered           ID
eca            never     no     --                   0        10.56.25.0/24
               6d18h     yes#   21.21.21.21:40875    0        10.56.25.25/32
               01:23:56  yes#   27.27.27.27:24949    0        10.56.25.64/32
               00:24:40  yes#   24.24.24.24:23672    0        10.56.25.72/32
               6d18h     yes#   21.21.21.21:40875    0        23.23.23.23/32
```

```
                6d17h     yes#   24.24.24.24:23672    0        26.26.26.26/32
                3w0d      yes#   27.27.27.27:24949    0        29.29.29.29/32
```

To verify the operational status of LISP sites on IPv4 database, use this command in Fabric Edge Node:

```
FabricEdge# show lisp instance-id 0 ipv4 database

LISP ETR IPv4 Mapping Database for EID-table default (IID 0), LSBs: 0x1
Entries total 3, no-route 0, inactive 0
10.56.25.27/32, dynamic-eid eid_10_56_25, skip reg, inherited from default
locator-set rloc_loopback
  Uptime: 00:25:54, Last-change: 00:25:54
  Domain-ID: unset
  Locator      Pri/Wgt  Source     State
  24.24.24.24   10/10   cfg-intf   site-self, reachable
10.56.25.72/32, dynamic-eid eid_10_56_25, inherited from default locator-set
rloc_loopback
 Uptime: 00:25:25, Last-change: 00:25:25
  Domain-ID: unset
  Locator      Pri/Wgt  Source     State
  24.24.24.24   10/10   cfg-intf   site-self, reachable
26.26.26.26/32, locator-set rloc_loopback
  Uptime: 3w5d, Last-change: 6d17h
  Domain-ID: unset
  Locator      Pri/Wgt  Source     State
  24.24.24.24   10/10   cfg-intf   site-self, reachable
```

To verify the operational status of LISP sites on mac mapping database, use this command on the FE node:

```
FabricEdge# show lisp instance-id 1 ethernet database

LISP ETR MAC Mapping Database for EID-table Vlan 25 (IID 1), LSBs: 0x1
Entries total 2, no-route 0, inactive 0
cc98.911b.73f1/48, dynamic-eid Auto-L2-group-1, skip reg, inherited from default
 locator-set rloc_loopback
  Uptime: 00:00:49, Last-change: 00:00:49
  Domain-ID: unset
  Locator      Pri/Wgt  Source     State
  24.24.24.24   10/10   cfg-intf   site-self, reachable
dcce.c130.0b70/48, dynamic-eid Auto-L2-group-1, inherited from default
locator-set rloc_loopback
  Uptime: 00:00:50, Last-change: 00:00:50
  Domain-ID: unset
  Locator      Pri/Wgt  Source     State
  24.24.24.24   10/10   cfg-intf   site-self, reachable
```

Part XX

VLAN

Topics:

- VLANs
- VLAN Groups

Configuring VLANs chapter

200 VLANs

Topics:

- Virtual Local Area Network
- Configuring normal-range VLANs
- Extended-range VLANs
- Monitor VLANs

Virtual Local Area Network

Introduces VLANs as logical networks that segment switched networks by function or application, explains their isolation and packet forwarding behavior, and outlines how VLANs relate to IP subnets and interface-based membership.

A VLAN (Virtual Local Area Network) is a type of logical network that

- segments switched networks by function, team, or application without regard to the physical locations of the users
- forwards packets (unicast, broadcast, multicast) only among end stations in the same VLAN, and
- maintains its own bridge Management Information Base (MIB) for isolation and control.

VLANs have the same attributes as physical LANs. However, you can group end stations even if they are not physically located on the same LAN segment. Each VLAN is considered a logical network. To send packets to stations outside the VLAN, use a router or a controller that supports fallback bridging.

VLANs are often associated with IP subnets. For example, all the end stations in a particular IP subnet belong to the same VLAN. Assign controller interfaces to VLANs manually to use interface-based (static) VLAN membership.

Supported VLANs

The topic describes the controller supporting VLANs.

The controller supports VLANs in VTP client, server, and transparent modes. VLANs are identified by a number from 1 to 4094.

- VLAN 1 is the default VLAN and is created during system initialization.
- All of the VLANs except 1002 to 1005 are available for user configuration.

VLAN port membership modes

This section explains VLAN port membership modes.

A VLAN port membership mode is a configuration method that

- determines how a switch port associates with one or more VLANs
- specifies the type of VLAN traffic the port can carry, and
- controls whether VLAN membership is assigned manually or dynamically.

To assign a port to a VLAN, set the membership mode. The membership mode determines which type of traffic the port carries and how many VLANs the port can join. When a port belongs to a VLAN, the controller manages the port addresses for each VLAN.

Table 181: Port Membership Modes and Characteristics

Membership Mode	VLAN Membership Characteristics	VTP Characteristics
Static-access	A static-access port can belong to one VLAN and is manually assigned to that VLAN.	VTP is not required. Set the VTP mode to transparent if you do not want VTP to propagate information globally. Participation in VTP requires at least one trunk port on the controller connected to a trunk port on a second controller.

Membership Mode	VLAN Membership Characteristics	VTP Characteristics
Trunk IEEE 802.1Q) : • IEEE 802.1Q– Industry-standard trunking encapsulation.	A trunk port is a member of all VLANs by default, including extended-range VLANs, but membership can be limited by configuring the allowed-VLAN list.	VTP maintains VLAN configuration consistency by managing the addition, deletion, and renaming of VLANs across the network. VTP exchanges VLAN configuration messages with other controllers over trunk links.

Note

If a client VLAN has both a primary and a secondary subnet, you cannot assign a static IP address on the secondary subnet.

Consider this SVI configuration example:

interface VlanX

ip address a.b.c.254 255.255.255.0 secondary

ip address a.d.e.254 255.255.255.0

In this scenario, you can't allocate the secondary subnet for clients with static IP addresses.

VLAN configuration files

This topic describes configurations for VLAN IDs 1 to 1005 are written to the vlan.dat file.

When you configure VLAN IDs 1 to 1005, your settings are saved to the vlan.dat file (VLAN database). To display these configurations, enter the show vlan privileged EXEC command. The vlan.dat file is stored in flash memory. If the VTP mode is transparent, your configurations are also saved in the controller running configuration file.

Use interface configuration mode to define the port membership mode and to add or remove ports from VLANs. The results of your commands are saved to the running-configuration file. To display the running configuration, enter the show running-config privileged EXEC command. When you save VLAN and VTP information (including extended-range VLAN configuration information) in the startup configuration file and reboot the controller, the system chooses its configuration based on several criteria.

- If the VTP mode is transparent in the startup configuration and the VLAN database and domain name match those in the startup configuration file, the device ignores the VLAN database. In this situation, the controller uses the VTP and VLAN configurations from the startup configuration file. The VLAN database revision number does not change.
- If the VTP mode or domain name in the startup configuration does not match the VLAN database, the device uses the domain name, VTP mode, and VLAN IDs 1 to 1005 configuration from the VLAN database.
- In VTP versions 1 and 2, if you set the VTP mode to server, the device uses the domain name and VLAN configuration for VLAN IDs 1 to 1005 from the VLAN database. VTP version 3 supports VLANs 1006 to 4094.

> **Note**
>
> Delete the vlan.dat file and the configuration files before you reset the switch configuration with the write erase command. This step ensures the switch reboots correctly when you reset it.

Normal-range VLAN configuration guidelines

This topic describes the guidelines to create and modify normal-range VLAN configuration.

These guidelines help you create or change normal-range VLANs in your network.

- Normal-range VLANs use numbers from 1 to 1001.
- VLAN configurations for VLANs 1 to 1005 are always saved in the VLAN database. In transparent VTP mode, both VTP and VLAN configurations are also saved in the running configuration file.
- If your controller works in VTP server mode or VTP transparent mode, you can add, change, or remove VLANs 2 to 1001 in the VLAN database. VLAN ID 1, as well as VLANs 1002 to 1005, are automatically created. You cannot remove these VLANs.
- The system does not save extended-range VLANs that you create in VTP transparent mode, and these VLANs are not shared across the network. VTP version 3, when in server mode, allows you to share extended-range VLANs (VLANs 1006 to 4094) in the database.

If clients cannot connect to the controller because of a VLAN failure, you have these options:

- Configure **ip4 dhcp required** in the policy profile to make the client start a DHCP request.
- Set up the RADIUS server to send VLAN group information, including the client's static IP VLAN, so the client can use a static IP address.
- Configure 'aaa-override vlan fallback' in the policy profile to make the controller look for the client's static IP VLAN in other VLAN groups. The client can join the network if its static IP VLAN is listed in the VLAN group set under the policy profile.

Extended-range VLAN configuration guidelines

This topic describes the guidelines to create extended-range VLAN configuration.

Extended-range VLANs are VLANs with IDs from 1006 to 4094.

Follow these guidelines when creating extended-range VLANs:

- Extended-range VLAN IDs are not saved in the VLAN database. VTP does not recognize these VLANs unless your device uses VTP version three.
- Do not include extended-range VLANs in the pruning eligible range.
- For VTP version 1 or 2, set the VTP mode to transparent in global configuration mode. Save this configuration to the startup configuration so that your device boots in VTP transparent mode. Save the configuration to avoid losing the extended-range VLAN configuration when the device resets.

Prerequisites for VLANs

The topic describes the prerequisites for VLANs.

These are the prerequisites and considerations for configuring VLANs:

- To configure VLAN through the GUI, change the number of available Virtual Terminal (VTY) sessions to 50. The Web UI processes HTTP requests using VTY lines. If several connections are open, the device's default limit of 15 VTY lines can be exceeded. Increase the number of VTY lines to 50 before using the Web UI.

  Note

  To increase the number of VTY lines on your device, enter these configuration commands.

  ```
  Device# configure terminal
  Device(config)# service tcp-keepalives in
  Device(config)# service tcp-keepalives out

  Device# configure terminal
  Device(config)# line vty 16-50
  ```

  Note

  The maximum number of SSH VTY sessions supported on the standby controller is eight.

- Decide if you will use VLAN Trunking Protocol (VTP) to manage global VLAN configuration for your network before creating VLANs.
- Create the VLAN on the device before adding it to a VLAN group.

Restrictions for VLANs

This section describes the restrictions for VLANs.

These are the restrictions for VLANs:

- You cannot delete a wireless management interface if its associated VLAN interface has already been deleted. To prevent this issue, delete the wireless management interface before deleting the VLAN interface.
- The device supports IEEE 802.1Q trunking methods for sending VLAN traffic over Ethernet ports.
- When no client VLAN is configured for a policy profile, the AP native VLAN is used.
- The behavior of VLAN 1 changes depending on the AP mode. The device handles VLAN scenarios as described here:
  - Local mode AP : If you use *vlan-name* , clients are assigned to VLAN 1. If you use *vlan-id* 1, clients are assigned to the wireless management interface.
  - FlexConnect mode AP : If you use *vlan-name* , clients are assigned to VLAN 1. If you use *vlan-id* 1, clients are assigned to the native VLAN defined in the flex profile.

  By default, the policy profile assigns vlan-id 1 so that clients can use the wireless management VLAN.
- You cannot use the same VLAN on the same SSID for both local switching and central switching.

Configuring normal-range VLANs

Guides administrators through configuring normal-range VLANs, including creating, modifying, and deleting VLANs, setting key parameters, validating settings, and maintaining VLAN database integrity to ensure proper Layer 2 network segmentation and reliable switching operation.

Summary

Configuring normal-range VLANs lets administrators create or modify VLANs within the standard allowed range. It also lets them assign key properties and ensures proper Layer 2 network segmentation. The process requires setting necessary parameters and using recommended practices to maintain VLAN database integrity. The process includes several key components.

- The network administrator manages VLAN creation, modification, and deletion.
- The switch VLAN database stores and applies VLAN configurations.
- VLAN parameters include attributes such as VLAN ID, name, type, state, and associations which define each VLAN.

Workflow

Normal-range VLANs are VLANs with IDs from 1 to 1005. Proper configuration is essential for network segmentation and security. It also helps maintain a reliable switching environment. The process consists of several stages:

1. Access the VLAN database: The administrator uses the switch's VLAN configuration mode to begin changes.
2. Identify or create the VLAN: Select an existing VLAN or specify a new VLAN ID within the normal range (1-1005).
3. Set VLAN parameters: Assign a VLAN name, choose a VLAN type (Ethernet, Token Ring Bridge Relay Function [TrBRF], or Token Ring Concentrator Relay Function [TrCRF]), and set the VLAN state (active or suspended).
4. Validate the VLAN settings: Confirm that the VLAN was created or modified as expected, using appropriate verification commands.
5. Maintain VLAN database integrity: Avoid deleting the `vlan.dat` file manually because this can cause inconsistencies. Always use standard switch commands to modify or remove VLAN configurations.

Create or modify an Ethernet VLAN (CLI)

Use this procedure to create or modify an Ethernet VLAN.

Configure a new Ethernet VLAN or update an existing one using commands on the controller.

Before you begin

With VTP version 1 and 2, the controller in VTP transparent mode allows you to assign VLAN IDs greater than 1006. However, these VLANs are not added to the VLAN database.

The controller supports only Ethernet interfaces.

Procedure

1. Enter a VLAN ID to access VLAN configuration mode. Enter a new VLAN ID to create a VLAN, or enter an existing VLAN ID to modify that VLAN.

   Example

   ```
   Device(config)# vlan 20
   ```

> **Note**
>
> The valid VLAN ID range for this command is between 1 to 4094.

2. Enter a name for the VLAN.

   Example

   ```
   Device(config-vlan)# name test20
   ```

   If you do not enter a name for the VLAN, the system appends the *vlan-id* value with leading zeros to the word VLAN.

3. Configure the VLAN media type.

   Example

   ```
   Device(config-vlan)# media {ethernet | fd-net | trn-net}
   ```

4. Verify your entries.

   Example

   ```
   Device# show vlan {name test20 | id 20}
   ```

Assign static-access ports to a VLAN (GUI)

Use this procedure to assign static-access ports to a VLAN using GUI.

Assign static-access ports to a specific VLAN from the user interface using the GUI.

Procedure

1. Choose **Configuration** > **Layer2** > **VLAN** > **VLAN**.
2. Click the **VLAN** tab.
3. To assign **Port Members**, select the desired interfaces in the **Available** list. Then, use the arrow to move them to the **Associated** list.
4. Click **Update & Apply to Device**.

Assign static-access ports to a VLAN (CLI)

Assign static-access ports to a VLAN using commands.

Assign Layer 2 switch ports to a specific VLAN for network segmentation using commands.

You can assign a static-access port to a VLAN without having VTP globally propagate VLAN configuration information by disabling VTP (VTP transparent mode). For more information on static-access ports, see *VLAN Port Membership Modes*.

If you assign an interface to a VLAN that does not exist, the new VLAN is created.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the interface to be added to the VLAN.

   Example

   ```
   Device(config)# interface gigabitethernet2/0/1
   ```

3. Define the VLAN membership mode for the port (Layer 2 access port).

   Example

   ```
   Device(config-if)# switchport mode access
   ```

4. Assign the port to a VLAN.

   Example

   ```
   Device(config-if)# switchport access vlan 2
   ```

   The valid VLAN IDs range is between 1 to 4094.

5. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-if)# end
   ```

6. Verify the VLAN membership mode of the interface.

   Example

   ```
   Device# show running-config interface interface-id
   ```

7. Verify your entries in the *Administrative Mode* and the *Access Mode VLAN* fields of the display.

   Example

   ```
   Device# show interfaces gigabitethernet2/0/1 switchport
   ```

Extended-range VLANs

Introduces extended-range VLANs, outlining their purpose, VLAN ID support, and differences in configuration storage based on VTP versions and modes.

Extended-range VLANs enable service providers to extend their infrastructure to a greater number of customers. The extended-range VLAN IDs are permitted in any **switchport** command that accepts VLAN IDs.

- With VTP version 1 or 2, extended-range VLAN configurations are not stored in the VLAN database. However, if VTP mode is set to transparent, the configurations are stored in the controller running configuration file, and you can also save them in the startup configuration file.
- For VTP version 3, extended-range VLANs are stored in the VLAN database.

Create an extended-range VLAN (GUI)

Use this procedure to create an extended-range VLAN using GUI.

Add an extended-range VLAN to your network to support larger scaling or specialized network segmentation using the GUI.

Procedure

1. Choose **Configuration** > **Layer2** > **VLAN**.
2. In the VLAN page, click **ADD**.
3. Enter the extended range VLAN ID in the **VLAN ID** field.

   The extended range is between range is 1006 and 4094.
4. Enter a VLAN name in the **Name** field.
5. Save the configuration.

Create an extended-range VLAN (CLI)

Use this procedure to create an extended-range VLAN using commands.

Create an extended-range VLAN on a device using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Enter an extended-range VLAN ID and enter the VLAN configuration mode.

   Example

   ```
   Device(config)# vlan 2000
   ```

   The range is between 1006 to 4094.
3. Verify that the VLAN has been created.

Example

```
Device# show vlan id 2000
```

Monitor VLANs

This section provides the details about monitoring VLANs.

Table 182: Privileged EXEC show commands

Command	Purpose
show interfaces [**vlan** *vlan-id*]	Displays characteristics for all interfaces or for the specified VLAN configured on the controller.
show vlan [**access-map** *name* | **brief** | **group** | **id** *vlan-id* | **ifindex** | **mtu** | **name** *name* | **summary**]	Displays parameters for all VLANs or the specified VLAN on the controller. These command options are available: • ***brief*** –Displays VTP VLAN status in brief. • ***group*** –Displays the VLAN group with its name and the connected VLANs that are available. • ***id*** –Displays VTP VLAN status by identification number. • ***ifindex*** –Displays SNMP ifIndex. • ***mtu*** –Displays VLAN MTU information. • ***name*** –Displays the VTP VLAN information by specified name. • ***summary*** –Displays a summary of VLAN information.

201 VLAN Groups

Topics:

- VLAN groups
- VLAN group support for DHCP and static IP clients

VLAN groups

Outlines the configuration and operation of VLAN groups for wireless networks, including dynamic VLAN assignment, policy profile mapping, and management of VLAN integrity using Dirty flag mechanisms based on client DHCP failures and MAC address hashing.

A VLAN group is a wireless network configuration feature that

- enables assignment of multiple VLANs to a single policy profile
- dynamically allocates a wireless client to a VLAN within the group based on the client's MAC address, and
- manages VLAN availability and integrity by marking VLANs as *Dirty* when the clients are unable to receive IP addresses using DHCP.

Whenever a client connects to a wireless network (WLAN), the client is placed in a VLAN associated with the policy profile mapped to the WLAN. In a large venues, such as an auditorium, stadiums, or conference rooms with numerous wireless clients, managing connections using only a single WLAN can be challenging.

The clients can be assigned to one of the configured VLANs. This feature maps a policy profile to a single VLAN or to multiple VLANs using VLAN groups. A VLAN is assigned to the client, and the client receives the IP address from the assigned VLAN.

The system might not clear the *Dirty* flag from the VLAN, even after 30 minutes, for a VLAN group. After 30 minutes, when the VLAN is marked as non-dirty, new clients in the IP Learn state can be assigned IP addresses from the VLAN if available IPs exist in the pool and the DHCP scope is configured correctly. This is expected because the timestamp of each interface must be checked to confirm it exceeds 30 minutes, which results in a 5-minute lag before the global timer expires.

Note

The Controller marks the VLAN interface as *Dirty* when three or more clients fail to receive IP addresses through DHCP. The VLAN interface is marked *Dirty* if the VLAN *Dirty* counter increases as a result of an ***IP Learn Failure***, which occurs when a client sends three DHCP discoveries without receiving a response. This occurs before the client is deleted due to an ***IP Learn Timeout***.

Each client receives a unique hash value derived from its MAC address. This approach ensures that clients from the same vendor, which may differ only by a few bits, do not cause incorrect *Dirty* VLAN markings.

Prerequisites for VLAN groups

This section describe a prerequisite for VLAN groups.

Review the required condition for VLAN group.

- A VLAN should be present in the device for it to be added to the VLAN group.
- IP MAC binding should be configured on a policy profile that uses a VLAN group.

Restrictions for VLAN groups

This section describes the restrictions for VLAN groups.

Follow these restrictions when configuring VLAN groups:

- If the number of VLANs in a VLAN group exceeds 32, the mobility functionality might not work as expected and Layer 2 multicast might break for some VLANs. Therefore, it is the responsibility of network administrators to configure a feasible number of VLANs in a VLAN group.
- The VLAN Groups feature works for APs in local mode.
- The VLAN Groups feature operates only in central switching mode; it cannot be used in FlexConnect local switching mode.
- The ARP Broadcast feature is not supported on VLAN groups.
- Multicast with VLAN group is only supported for access points in local mode. A multicast VLAN is required when a VLAN group is configured and multicast traffic is used.
- When configuring VLAN groups with multiple VLANs, and each VLAN is used by a different subnet, clients with static IP addresses might be assigned to an incorrect VLAN if SVIs are not present on the controller. For every VLAN in the VLAN group, configure an SVI interface with a valid IP address.

Create a VLAN Group (GUI)

Use this procedure to create a VLAN group using GUI.

Create a VLAN group to assign multiple VLAN IDs efficiently in the network using the GUI.

Procedure

1. Choose **Configuration** > **Layer2** > **VLAN Group**.
2. Click **Add**.
3. Set the **VLAN GROUP** name and the list of **VLANS**.

   This can be a range, or a set of comma-separated values.
4. Enter the VLAN name in the **Name** field.

   Configure the other parameters if required.
5. Click **Update & Apply to Device**.

Create a VLAN Group (CLI)

Use this procedure to create a VLAN group using commands.

Create a VLAN group for network segmentation and traffic management using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Create a VLAN group with the given group name and add all the VLANs listed in the command.

Example

```
Device(config)# vlan group vlangrp1 vlan-list 91-95.
```

The VLAN list ranges from 1 to 4096 and the maximum number of VLANs supported in a group is 64.

3. Exit the global configuration mode and return to the privileged EXEC mode.

Example

```
Device(config)# end
```

Alternatively, press *CTRL-Z* to exit the global configuration mode.

Add a VLAN group to policy profile (GUI)

Use this procedure to add a VLAN group to policy profile using GUI.

Assign a VLAN group to an existing policy profile, enabling client traffic association with the specified VLANs using commands.

The policy profile consists mainly of network and switching policies. The policy profile is a reusable entity that can be applied to multiple tags. When you apply a client policy to the AP or controller, the policy profile includes it. Examples are VLAN, ACL, QoS, session timeout, idle timeout, AVC profile, Bonjour profile, local profiling, device classification, and BSSID QoS. However, all security attributes and features related to wireless on the WLAN are grouped under the WLAN profile.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. On the **Policy Profile** page, click on a policy profile name.
3. Click on the **Access Policies** tab.
4. Under **VLAN** section, select a a VLAN or VLAN Group from the **VLAN/VLAN Group** drop-down list.
5. Click **Update & Apply to Device**.

Add a VLAN group to a policy profile (CLI)

Use this procedure to add a VLAN group to policy profile using commands.

Map a specific VLAN group to a WLAN policy profile using commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the WLAN policy profile.

Example

```
Device(config)# wireless profile policy my-wlan-policy
```

3. Map the VLAN group to the WLAN by entering the group name.

Example

```
Device(config-wireless-policy)# vlan myvlan-group
```

4. Exit the global configuration mode and return to the privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

View the VLANs in a VLAN group

This section discusses how you can view the VLANs in a VLAN group.

These commands let you view VLAN group information and VLAN assignments:

Command	Description
show vlan group	Displays the list of VLAN groups with name and the VLANs that are configured.
show vlan group group-name *group_name*	Displays the specified VLAN group details.
show wireless client mac-address *client-mac-addr* **detail**	Displays the VLAN group assigned to the client.
show wireless vlan details	Displays VLAN details.

VLAN group support for DHCP and static IP clients

Outlines VLAN group support for assigning DHCP and static IP clients to VLANs, configuring MAC and IP bindings, enforcing network access policies, and details prerequisites and feature history for IPv4 client management.

A VLAN group feature is a network function that

- assigns clients to VLANs based on DHCP or static IP information
- restricts internet access for static IP clients whose addresses do not match the VLAN's IP list, and
- applies only to IPv4 clients with automatic enablement unless explicitly disabled by configuration.
- The VLAN Group to support DHCP and Static IP clients feature manages network access for DHCP and static IP clients.
- Ensure that the **ipv4 dhcp required** command is not configured on the wireless policy profile. If this command is present, the feature is disabled, and the client remains in the IP learn state. Ensure that the **ip mac-binding** command is configured on the wireless policy profile. This command associates VLAN information with the MAC address and IP address assigned to the client, whether assigned statically or through DHCP.

Feature history

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 183: Feature history for VLAN group support for DHCP and static IP clients

Feature Name	Release Information	Feature Description
VLAN group support for DHCP and static IP clients	Cisco IOS XE 17.9.1	From this release VLAN group support for DHCP and static IP clients enhances network segmentation by allowing controllers to assign both DHCP and IPv4 static IP clients to VLANs using computed logic.

Prerequisites and restrictions of VLAN Group support for DHCP and static IP clients

Prerequisites and restrictions of VLAN Group support for DHCP and static IP clients.

Prerequisites of VLAN Group support for DHCP and static IP clients

- Ensure that a switch VLAN interface (SVI) is configured with the IP address.
- Ensure that IP MAC-binding is configured on the target Policy Profile(s).

Restrictions of VLAN Group support for DHCP and static IP clients

- FlexConnect local switching and FlexConnect local authentication are not supported. Only local mode, FlexConnect central switching, and FlexConnect central authentication are supported.
- IPv6 is not supported.
- The peer controller cannot have a VLAN group in the policy profile, because a VLAN group with static IP mobility is not supported. To support VLAN group with an anchor peer controller, **static-ip-mobility** must be configured and **DHCP required** must be disabled.

Supported features

This section provides details about supported features.

This table provides details about supported features for VLAN group support for DHCP and static IP clients.

- Configure SVI with an IP address in the same subnet as the client's IP address.
- Clients are excluded if there is no matching SVI.

Feature	Support
Guest Anchor	Yes
Mobility	Yes
RLAN	Yes

SVI	Yes Ensure that you configure SVI with an IP address in the same subnet as that of the client's IP address.
IRCM support: Guest AireOS as anchor and Cisco Catalyst 9800 controller as foreign	Yes
IRCM support : Guest AireOS as foreign and Cisco Catalyst 9800 controller as anchor	Yes The client is excluded if there is no match for the SVI.

Part XXI

WLAN

Topics:

- WLANs
- WLAN Security
- Remote LANs
- RLAN External Module
- 802.11ax Per WLAN
- BSS Coloring
- DHCP for WLANs
- Aironet Extensions IE (CCX IE)
- Device Analytics
- Device Classifier Dynamic XML Support
- BSSID Counters
- Fastlane+
- Workgroup Bridges
- Peer-to-Peer Client Support
- Deny Wireless Client Session Establishment Using Calendar Profiles
- Ethernet over GRE
- Wireless Guest Access
- Wired Guest Access
- Data Rate Selection at SSID Level
- User Defined Network
- Hotspot 2.0
- Client Roaming Across Policy Profile
- Assisted Roaming
- 802.11r BSS Fast Transition
- 802.11v
- Virtual Routing and Forwarding
- Automated Frequency Coordination

202 WLANs

Topics:

- WLANs
- How to configure WLANs

WLANs

Explains wireless local area networks that enable control of lightweight access points with configurable SSIDs and profiles.

A WLAN is a wireless network feature that

- enables control of wireless local area networks for lightweight access points
- supports up to 16 advertised WLANs per access point with up to 4096 total configurable WLANs, and
- allows selective advertisement using profiles and tags for better manageability.

An SSID identifies the specific wireless network that you want the device to access. You can configure WLANs with different SSIDs or with the same SSID.

WLAN configuration details

Each WLAN has a separate WLAN ID, a separate profile name, and a WLAN SSID.

Note

The **wireless client max-user-login concurrent** command will work as intended even if the **no configure max-user-identity response** command is configured.

Note

We recommend that you configure the **password encryption aes** and the **key config-key password-encrypt** *key* commands to encrypt your password.

Note

For C9105, C9115, and C9120 APs, when a new WLAN is pushed from the controller and if the existing WLAN functional parameters are changed, the other WLAN clients will disconnect and reconnect.

Wi-Fi multimedia (WMM) policy

Wi-Fi Multimedia (WMM) is used to prioritize different types of traffic. The WMM policy options include:

- **Disabled**: Disables WMM on the WLAN.
- **Required**: Requires client devices to use WMM. Devices that do not support WMM cannot join the WLAN.
- **Allowed**: Devices that do not support WMM can join the WLAN but will not benefit from the 802.11n rates.

> **Note**
>
> For 802.11ax APs and higher (Cisco Catalyst 9117, 9124, 913x, 916x, 917x APs), WMM can not be **Disabled**. Only **Required** or **Allowed** options are supported.

Prerequisites for WLANs

Outlines the prerequisites for configuring WLANs.

You can associate up to 16 WLANs with each access point group and assign specific access points to each group. Each access point advertises only the enabled WLANs that belong to its access point group. The access point (AP) does not advertise disabled WLANs in its access point group or WLANs that belong to another group.

We recommend that you assign one set of VLANs for WLANs and a different set of VLANs for management interfaces to ensure that devices properly route VLAN traffic.

Restrictions for WLANs

Lists restrictions for WLANs.

This reference provides restrictions and limitations that apply when configuring and managing WLANs on wireless controllers.

- Do not configure PSK as this configuration is not supported and impacts client join flow.
- Ensure that TKIP or AES ciphers are enabled with WPA1 configuration, else ISSU may break during upgrade process.
- When you change the WLAN profile name, then FlexConnect APs (using AP-specific VLAN mapping) will become WLAN-specific. If FlexConnect Groups are configured, the VLAN mapping will become Group-specific.
- Do not enable IEEE 802.1X Fast Transition on Flex Local Authentication enabled WLAN, as client association is not supported with Fast Transition 802.1X key management.
- Peer-to-peer blocking does not apply to multicast traffic.
- In FlexConnect, peer-to-peer blocking configuration cannot be applied only to a particular FlexConnect AP or a subset of APs. It is applied to all the FlexConnect APs that broadcast the SSID.
- The WLAN name and SSID can have up to 32 characters.
- WLAN and SSID names support only the following ASCII characters:
  - Numerals: 48 through 57 hex (0 to 9)
  - Alphabets (uppercase): 65 through 90 hex (A to Z)
  - Alphabets (lowercase): 97 through 122 hex (a to z)
  - ASCII space: 20 hex
  - Printable special characters: 21 through 2F, 3A through 40, and 5B through 60 hex, that is: ! " # $ % & ' ( ) * + , - . / : ; < = > ? @ [\ ] ^ _ ` { | } ~
- WLAN name cannot be a keyword; for example, if you try to create a WLAN with the name as 's' by entering the **wlan s** command, it results in shutting down all WLANs because 's' is used as a keyword for shutdown.
- You cannot map a WLAN to VLAN 0. Similarly, you cannot map a WLAN to VLANs 1002 to 1006.
- Dual stack clients with a static-IPv4 address is not supported.

- In a dual-stack with IPv4 and IPv6 configured in the Cisco 9800 controller, if an AP tries to join controller with IPv6 tunnel before its IPv4 tunnel gets cleaned, you would see a traceback and AP join will fail.
- When creating a WLAN with the same SSID, you must create a unique profile name for each WLAN.
- When multiple WLANs with the same SSID get assigned to the same AP radio, you must have a unique Layer 2 security policy so that clients can safely select between them.
- The SSID that is sent as part of the user profile will work only if **aaa override** command is configured.
- RADIUS server overwrite is not configured on a per WLAN basis, but rather on a per AAA server group basis.
- Downloadable ACL (DACL) is supported only on the central switching mode. It is not supported for Flex Local switching or on the Cisco Embedded Wireless Controller.
- You cannot mix open configuration models with CLI-based, GUI-based, or Catalyst Center-based configurations. However, if you decide to use multiple model types, they must remain independent of each other. For example, in open configuration models, you can only manage configurations that have been created using an open configuration model, not a CLI-based or GUI-based model. Configurations that are created using open configuration models cannot be modified using a GUI-based model, or CLI-based model, or any other model.
- When you are configuring **dot11bg 11g** command and **radio dot11bg** or **radio dot11g** command, the clients can still connect in 5GHz radio. In this scenario, the client association needs to be blocked. This option is only available on a 2.4-GHz radio.

> ⚠ **Caution**
>
> Some clients might not be able to connect to WLANs properly if they detect the same SSID with multiple security policies. Use this WLAN feature with care.

Band selection

Describes how band select facilitates the movement of dual-band client radios from congested frequency bands to less congested 5-GHz access points for improved network performance.

A band selection is a wireless network feature that

- facilitates the movement of dual-band client radios from congested frequency bands
- enables clients to connect to less congested 5-GHz access points for improved network performance, and
- reduces interference from other devices by optimizing which frequency band clients use.

The 2.4-GHz band is often congested. Clients using this band may experience interference from Bluetooth devices, microwave ovens, cordless phones, and co-channel interference from other access points due to the 802.11b/g channel limit of 3 nonoverlapping channels. By configuring band selection on devices, administrators can steer dual-band clients toward the 5-GHz band, improving overall network performance.

Off-channel scanning deferral

Describes a wireless network feature that temporarily postpones RRM off-channel scanning activities.

Off-channel scanning deferral is a wireless network feature that

- temporarily postpones RRM off-channel scanning activities when important data transmission occurs
- prevents performance impact on critical traffic by avoiding the normal 70-millisecond off-channel periods, and

- can be configured on a per-WLAN basis with specific WMM UP class and time threshold parameters.

Off-channel scanning operations

A lightweight access point, in normal operational conditions, periodically goes off-channel and scans another channel. This is in order to perform RRM operations such as the following:

- Transmitting and receiving Neighbor Discovery Protocol (NDP) packets with other APs.
- Detecting rogue APs and clients.
- Measuring noise and interference.

During the off-channel period, which normally is about 70 milliseconds, the AP is unable to transmit or receive data on its serving channel. Therefore, there is a slight impact on its performance and some client transmissions might be dropped.

While the AP is sending and receiving important data, it is possible to configure off-channel scanning deferral so that the AP does not go off-channel and its normal operation is not impacted. You can configure off-channel scanning deferral on a per-WLAN basis, per WMM UP class basis, with a specified time threshold in milliseconds. If the AP sends or receives, on a particular WLAN, a data frame marked with the given UP class within the specified threshold, the AP defers its next RRM off-channel scan. For example, by default, off-channel scanning deferral is enabled for UP classes 4, 5, and 6, with a time threshold of 100 millseconds. Therefore, when RRM is about to perform an off-channel scan, a data frame marked with UP 4, 5, or 6 is received within the last 100 milliseconds, RRM defers going off-channel. The AP radio does not go off-channel when a voice call sending and receiving audio samples is marked as UP class 6 for every active 20 milliseconds.

Off-channel scanning deferral does come with a tradeoff. Off-channel scanning can impact throughput by 2 percent or more, depending on the configuration, traffic patterns, and so on. Throughput can be slightly improved if you enable off-channel scanning deferral for all traffic classes and increase the time threshold. However, by not going off-channel, RRM can fail to identify AP neighbors and rogues, resulting in negative impact to security, DCA, TPC, and 802.11k messages.

DTIM period

Explains how DTIM period allows power-saving clients to wake up at the appropriate time.

A Delivery Traffic Indication Map (DTIM) period is a 802.11 network timing mechanism that

- allows power-saving clients to wake up at the appropriate time if they are expecting broadcast or multicast data
- determines when access points transmit buffered broadcast and multicast frames after beacon broadcasts, and
- coincides with the DTIM broadcast interval.

DTIM period characteristics

In the 802.11 networks, lightweight access points broadcast a beacon at regular intervals, which coincides with the Delivery Traffic Indication Map (DTIM). After the access point broadcasts the beacon, it transmits any buffered broadcast and multicast frames based on the value set for the DTIM period.

Typical DTIM values and their transmission patterns:

- DTIM value 1: Transmits broadcast and multicast frames after every beacon
- DTIM value 2: Transmits broadcast and multicast frames after every other beacon

For instance, if the beacon period of the 802.11 network is 100 ms and the DTIM value is set to 1, the access point transmits buffered broadcast and multicast frames for 10 times every second. If the beacon period is 100 ms and the DTIM value is set to 2, the access point transmits buffered broadcast and multicast frames for 5 times every second. Either of these settings are suitable for applications, including Voice Over IP (VoIP), that expect frequent broadcast and multicast frames.

However, the DTIM value can be set as high as 255 (to transmit broadcast and multicast frames after every 255th beacon). The only recommended DTIM values are 1 and 2; higher DTIM values will likely cause communications problems.

> **Note**
>
> A beacon period, which is specified in milliseconds on the device, is converted internally by the software to 802.11 Time Units (TUs), where 1 TU = 1.024 milliseconds. Depending on the AP model, the actual beacon period may vary slightly; for example, a beacon period of 100 ms may in practice equate to 104.448 ms.

WLAN radio policies

Explains the WLAN Radio Policy feature that allows broadcasting WLANs on corresponding slots.

A WLAN radio policy is a wireless network configuration feature that

- allows broadcasting the WLAN on the corresponding slot
- provides more granular control compared to existing WLAN features that broadcast on all applicable slots, and
- is supported only on 5 GHz band.

WLAN radio policy functionality

The existing WLAN feature allows you to broadcast WLAN on a specified radio on all the applicable slots. With the WLAN Radio Policy feature, you can broadcast the WLAN on the corresponding slot.

Restrictions for WLAN radio policy

Lists restrictions for WLAN radio policy.

This topic lists the restrictions that apply to WLAN radio policy configuration.

- WLAN is pushed to all the radios only if these configurations are used:
  - WPA3 + AES cipher + 802.1x-SHA256 AKM
  - WPA3 + AES cipher + OWE AKM
  - WPA3 + AES cipher + SAE AKM
  - WPA3 + CCMP256 cipher + SUITEB192-1X AKM
  - WPA3 + GCMP256 cipher + SUITEB-1X AKM
  - WPA3 + GCMP128 cipher + SUITEB192-1X AKM

Prerequisites for configuring Cisco client extensions

Outlines prerequisites for configuring Cisco client extensions.

The software supports CCX versions 1 through 5, which enables devices and their access points to communicate wirelessly with third-party client devices that support CCX. CCX support is enabled automatically for every WLAN on the device and cannot be disabled. However, you can configure Aironet information elements (IEs).

If Aironet IE support is enabled, the access point sends an Aironet IE 0x85 (which contains the access point name, load, number of associated clients, and so on) in the beacon and probe responses of this WLAN, and the device sends Aironet

IEs 0x85 and 0x95 (which contains the management IP address of the device and the IP address of the access point) in the reassociation response if it receives Aironet IE 0x85 in the reassociation request.

Peer-to-peer blocking

Explains a WLAN security feature that prevents direct communication between wireless devices within the same network.

Peer-to-peer blocking is a WLAN security feature that

- is applied to individual WLANs, where each client inherits the peer-to-peer blocking setting of the WLAN to which it is associated
- enables you to have more control over how traffic is directed by allowing you to choose to have traffic bridged locally within the device, dropped by the device, or forwarded to the upstream VLAN, and
- is supported for clients that are associated with local and central switching WLANs.

> **Note**
>
> Peer-to-peer blocking feature is VLAN-based. WLANs using the same VLAN has an impact, if Peer-to-peer blocking feature is enabled.

Diagnostic channels

Describes diagnostic channel selection for troubleshooting client WLAN communication issues.

A diagnostic channel is a troubleshooting feature that

- enables testing of client and AP communication to identify network difficulties
- allows corrective measures to be taken to make the client operational on the network, and
- provides GUI or CLI configuration options for enabling diagnostic tests.

Configuration options

You can use the device GUI or CLI to enable the diagnostic channel, and you can use the device **diag-channel** CLI to run the diagnostic tests.

> **Note**
>
> We recommend that you enable the diagnostic channel feature only for non-anchored SSIDs that use the management interface. CCX Diagnostic feature has been tested only with clients having Cisco ADU card

AP names in beacon

AP names in beacon is a feature that

- extends the maximum length for AP name from 16 to 32 characters
- prevents AP name truncation in beacons, and
- uses a vendor specific (VS) Information Element (IE) instead of CCX IEs.

The extended length matches the maximum length supported in the controller. This enhancement resolves issues where AP names were truncated due to a 16-character limit imposed by CCX Information Elements (IEs).

Feature history

Table 184: Feature history for AP names in beacons

Feature name	Release information	Feature description
AP names in beacons	Cisco IOS XE 17.18.2	AP names in beacon is a feature that allows AP names to be extended from 16 to 32 characters, prevents truncation of AP names in beacon frames, and uses a vendor-specific Information Element (IE) instead of the older CCX IEs.

How to configure WLANs

Configure WLANs to create wireless network connectivity. Define essential parameters, apply security policies, and provision access points to meet your organization's coverage and performance needs.

WLAN wizard

Explains a feature that simplifies the process of setting up a Wireless Local Area Network (WLAN).

A WLAN wizard is a simplified workflow that

- helps you quickly create a WLAN and setup some primary configurations for your specific deployment
- supports wireless deployment modes including Local mode, FlexConnect mode, and Guest CWA mode, and
- provides different authentication methods for each deployment mode.

Wireless deployment modes

The wizard supports the following wireless deployment modes:

- Local mode: In Local mode, the WLAN is broadcast in the campus locally.
- Flex Connect mode: In FlexConnect mode, the WLAN is broadcast remotely across the WAN in a branch.
- Guest CWA mode: In Guest CWA mode, the WLAN is created for guest access with Central Web Authentication (CWA).

Important

These steps help configure WLANs, otherwise an error might occur.

To configure a WLAN for your preferred wireless deployment mode using the WLAN wizard on the WebUI, go to **Configuration > Wireless Setup > WLAN Wizard**.

You can also navigate to the WLAN Wizard by these paths:

- On the Toolbar, click on the **Wireless Setup** icon and select **WLAN Wizard** from the drop-down list.

- On the left navigation pane, go to **Configuration > Tags & Profiles > WLANs** and click on **WLAN Wizard** on the top-right corner.

On the **WLAN Wizard** page, select a wireless deployment mode for the WLAN to initiate steps for setting up the WLAN with profiles, authentication methods, tags, and APs and other configurations.

Local mode

Describes the WLAN deployment configuration used in office setups with no branch offices.

Local mode is a WLAN deployment configuration that

- is used when the WLAN is present in an office setup with no branch offices
- creates two CAPWAP tunnels from an AP to the controller, one for management and the other for data traffic, and
- implements centrally switched behavior where data traffic is tunneled from the AP to the controller for routing.

Local mode traffic handling

In local mode, data traffic is tunneled (bridged) from the AP to the controller where it is then routed by some routing device. This behavior is known as "centrally switched" because the data traffic is processed at the controller level. Locally switched means the traffic is terminated at the local switch adjacent to the access point.



Authentication methods

Explains the authentication methods available for configuring a WLAN in local mode.

An authentication method is a WLAN security configuration that

- sets the method by which a client can access the WLAN
- decides the level of security on the WLAN, and
- is selected from the left panel when configuring a WLAN for local mode.

Authentication method options

The authentication method options are:

- PSK: A Pre-Shared Key (PSK) is a unique key created for individuals or groups of users on the same SSID. A client will have to enter the PSK to be authenticated and allowed to access the WLAN.
- Dot1x: The client must go through relevant EAP authentication model to start exchanging traffic in the WLAN.
- Local Web Authentication: The controller intercepts htttp(s) traffic and redirects the client to the internal web page for authentication.
- External Web Authentication: The controller intercepts htttp(s) traffic and redirects the client to the login page hosted on the external web server for authentication.
- Central Web Authentication: The controller redirects all web traffic from the client to the ISE login page for authentication.

Configure WLAN profile and policy

Configure WLAN profile and policy details including network name, SSID, and associated policy parameters.

Configure the WLAN profile and policy to define wireless network properties and network policies for client switching and AP management.

After selecting the Authentication method, click on **WLAN** on the left panel to enter the WLAN profile and policy details.

The WLAN profile defines the properties of a WLAN such as Profile Name, Status, WLAN ID, L2 and L3 Security parameters, AAA Server associated with this SSID and other parameters that are specific to a particular WLAN. The policy profile defines the network policies and the switching policies for a client (with the exception of QoS), which constitute the AP policies as well.

Procedure

1. In the **Network Name** section, enter a **WLAN profile name**, which is a unique name for your wireless network. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.
2. Enter a valid **SSID** for the WLAN. A valid SSID can be up to 32 characters and can contain spaces. A valid SSID can be ASCII characters from 0 to 31, with leading and trailing spaces. This is the broadcast name for your WLAN.
3. Enter the **WLAN ID**.
4. In the **WLAN Policy** section, enter the **Policy Profile name**. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.
5. Select the **VLAN** to be associated with the Policy Profile from the drop-down list.
6. To select an existing Policy Profile for the WLAN, click on **Select Existing** and choose a **Policy Profile** from the drop-down list.

The WLAN profile and policy are configured with the specified network name, SSID, WLAN ID, and associated policy parameters.

Configure authentication for WLAN

Configure authentication configurations and filters for the WLAN depending on the selected authentication method.

Set up authentication configurations to secure your WLAN and control client access based on your chosen authentication method.

Authentication configurations include keys, filters, ACLs, and parameter maps that correspond to your selected authentication method. Each authentication method requires specific configuration steps to establish secure client connections.

Follow these steps to configure authentication for your WLAN:

Procedure

1. If you have selected **PSK** as the authentication method, configure these parameters:
   a) In the **WLAN > Pre-Shared Key (PSK)** section, select the PSK format. Choose between ASCII and Hexadecimal formats.
   b) From the **PSK type** drop-down list, choose if you want the key to be unencrypted or AES encrypted.
   c) In the **Pre-Shared Key** field, enter the pass key for the WLAN.
2. If you have selected **Dot1x** as the authentication method, configure the following:
   a) In the **WLAN > AAA** tab, configure the AAA server list for the WLAN.
   b) Select any of the available AAA servers to add to the WLAN.
   c) To add a new AAA server to the list, click on Add New Server and enter the IP address and server-key.

d) To use an already configured AAA server list, click on **Use Existing** and select the appropriate list from the drop-down.

3. If you have selected **Local Web Authentication** as the authentication method, configure these parameters:
   a) In the **WLAN > Parameter Map** tab, configure the parameter map for the WLAN. A parameter map sets parameters that can be applied to subscriber sessions during authentication.
      1. In the **Global Configuration** section, configure the global parameter map.
      2. Enter an IPv4 or IPv6 address to configure a virtual IP address for redirecting the clients to the login page of the controller.
      3. From the Trustpoint drop-down list, select the trustpoint for HTTPS login page. The trustpoint corresponds to the device certificate the controller will use in conjunction with the virtual IP and hostname.
      4. In the **WLAN Specific Configuration** section, either create a new parameter map for the WLAN, or select an existing parameter map from the drop-down list.

   b) In the **WLAN > Local Users** tab, enter the username in the local database to establish a username-based authentication system.
      1. Enter the user name to be saved.
      2. From the **Password Encryption** drop-down list, choose if you want the password to be unencrypted or encrypted.
      3. In the **Password** field, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters and can contain embedded spaces.
      4. Click on the **+** sign to add the credentials to the database. Add as many user credentials as required.

4. If you have selected **External Web Authentication** as the authentication method, configure these parameters:
   a) In the **WLAN > Parameter Map** tab, configure the parameter map for the WLAN.
      1. In the **Global Configuration** section, configure the global parameter map.
      2. Enter an IPv4 or IPv6 address to configure the virtual IP address of the external web authentication login page to which the guest users are redirected.
      3. From the **Trustpoint** drop-down list, select the trustpoint for HTTPS login page. The trustpoint corresponds to the device certificate the controller will use in conjunction with the virtual IP and hostname.
      4. In the **WLAN Specific Configuration** section, either create a new parameter map for the WLAN, or select an existing parameter map from the drop-down list.
      5. To create a new parameter map, enter the parameter-map name.
      6. In the **Redirect URL for login** field, enter the URL of the external server that will host the authentication page for login.
      7. In the **Portal IPV4 Address** field, enter the IPv4 address of the external server to send redirects. If the external server uses an IPv6 address, in the **Portal IPV6 Address** field, enter the IPv6 address of the portal to send redirects.

   b) In the **WLAN > ACL / URL Filter** tab, configure the ACL rules and the URL filter list.
      1. In the **Pre Auth ACL** section, enter the name of the ACL.
      2. In the **IP address** field, enter the source IP address and the destination IP address. This will configure the ACL to permit packet transfer from and to the specified IP address. You can add as many IP addresses as required.
      3. In the **URL Filter** section, enter a name for the URL Filter list that you are creating.

4. Use the slider to set the list action to **Permit** or **Deny** the URLs.
5. Specify the URLs in the **URLs** box. Enter every URL on a new line.

5. If you have selected Central Web Authentication as the authentication method, configure the following:
   a) In the **WLAN > AAA/ACL** tab, configure the AAA server list and ACL for the WLAN.
   b) In the **AAA Configuration** section, select any of the available AAA servers to add to the WLAN. This will be the server where the clients will get authenticated.
   c) To add a new AAA server to the list, click on **Add New Server** and enter the IP address and server-key.
   d) To use an already configured AAA server list, click on **Use Existing** and select the appropriate list from the drop-down.
   e) In the **ACL List** section, enter the name of the ACL. This ACL will contain the rules regarding URLs that can be accessed by the client and should match the name configured on the RADIUS server.

Configure tags on WLAN

Configure tags on WLAN to define properties through associated policies that are inherited by clients and access points.

Configure tags on WLAN to define properties through associated policies that are inherited by clients and access points.

A Tag's property is defined by the policies associated to it. This property is in turn inherited by an associated client/AP. There are various type of tags, each associated to different profiles.

Procedure

1. In the **Site Configuration** section, either enter a site tag to be added, or select an existing site tag from the drop-down list. You can add as many tags as required. In the local mode, the site tag contains the AP join profile only.
2. In the **Policy Tag** section, either enter a policy tag to be added, or select an existing policy tag from the drop-down list. You can add as many tags as required. The policy tag constitutes mapping of the WLAN profile to the policy profile. The WLAN profile defines the wireless characteristics of the WLAN. The policy profile defines the network policies and the switching policies for the client.
3. In the **RF Tag** section, either enter an RF tag to be added, or select an existing RF tag from the drop-down list. You can add as many tags as required. The RF tag contains the 2.4 GHz and 5 GHz RF profiles.

The tags are successfully configured on the WLAN. The properties defined by the policies associated with these tags will be inherited by the associated clients and access points.

AP Provision

Configure access point provisioning to associate APs with WLAN tags and locations.

This task allows you to add access points to the local site and associate them with appropriate tags for WLAN configuration.

Once the Wireless network and RF characteristics are set up, access points can be added to the local site either using static AP MAC address assignment or by assigning already joined APs to a specific location.

To add tags and associate APs to the WLAN, click **AP Provisioning** from the left panel.

Procedure

1. The APs already discovered by the controller are listed in the **Provision Joined APs** tab. You can select the APs to be associated to the WLAN from this table.
2. To add tags to the selected APs, select the appropriate Policy Tag, Site Tag, and RF Tag from the respective drop-down lists. Click on **Add** to apply the tags.

3. To add APs manually, click on the **Pre-provision APs** tab. You can either add individual MAC addresses of the APs or upload a CSV file with the AP MAC addresses listed. The added APs will be listed in the table below.
4. Select the APs to be associated to the WLAN from this table.
5. To add tags to the selected APs, select the appropriate Policy Tag, Site Tag, and RF Tag from the respective drop-down lists. Click on **Add** to apply the tags.

   A table of all the APs and the tags added to them is displayed in the **Selected APs** tab.

6. Click **Apply**.

   This will create a WLAN in local mode with the authentication method, authentication filters, tags, and APs configured on it.

The access points are successfully provisioned and associated with the configured tags, creating a WLAN in local mode with the specified authentication and configuration settings.

FlexConnect mode

Describes a wireless solution for branch office and remote office deployments.

FlexConnect mode is a wireless solution that

- enables configuration and control of access points (AP) in a branch or remote office from the corporate office through a wide area network (WAN) link without deploying a controller in each office
- allows FlexConnect APs to switch client data traffic and perform client authentication locally when their connection to the controller is lost, and
- offers network survivability in the event of a loss of connection to the centralized wireless controller.

FlexConnect mode deployment

This figure shows a FlexConnect mode deployment:



Authentication methods

Describes the methods by which clients can access WLANs in FlexConnect mode and the security levels they provide.

An authentication method is a security mechanism that

- sets the method by which a client can access the WLAN
- decides the level of security on the WLAN, and
- determines how traffic is intercepted and redirected for client authentication.

Authentication method options

To configure a WLAN for FlexConnect mode, select the preferred authentication method from the left panel. The authentication method options are:

- Local Web Authentication: The controller intercepts htttp(s) traffic and redirects the client to the internal web page for authentication.

- External Web Authentication: The controller intercepts htttp(s) traffic and redirects the client to the login page hosted on the external web server for authentication.
- Central Web Authentication: The controller redirects all web traffic from the client to the ISE login page for authentication.

Configure WLAN profile and policy

Configure WLAN profile and policy settings to define network properties and switching policies for wireless networks.

Configure WLAN profile and policy settings to establish wireless network connectivity with appropriate security parameters and client policies.

After selecting the Authentication method, click on **WLAN** on the left panel to enter the WLAN profile and policy details.

The WLAN profile defines the properties of a WLAN such as Profile Name, Status, WLAN ID, L2 and L3 Security parameters, AAA Server associated with this SSID and other parameters that are specific to a particular WLAN. The policy profile defines the network policies and the switching policies for a client (with the exception of QoS), which constitute the AP policies as well.

Procedure

1. In the **Network Name** section, enter a **WLAN profile name**, which is a unique name for your wireless network.

   The name can be ASCII characters from 32 to 126, without leading and trailing spaces.

2. Enter a valid **SSID** for the WLAN.

   A valid SSID can be up to 32 characters and can contain spaces. A valid SSID can be ASCII characters from 0 to 31, with leading and trailing spaces. This is the broadcast name for your WLAN.

3. Enter the **WLAN ID**.
4. In the **WLAN Policy** section, enter the **Policy Profile name**.

   The name can be ASCII characters from 32 to 126, without leading and trailing spaces.

5. Select the **VLAN** to be associated with the Policy Profile from the drop-down list.
6. To select an existing Policy Profile for the WLAN, click on **Select Existing** and choose a **Policy Profile** from the drop-down list.

The WLAN profile and policy are configured with the specified network name, SSID, WLAN ID, and associated VLAN settings for wireless network connectivity.

Configure authentication configurations

Configure authentication settings and filters for the WLAN depending on your selected authentication method.

Configure authentication settings, keys, filters, ACLs, and parameter maps for your selected WLAN authentication method to enable secure wireless access for users.

Set up the authentication configurations and filters for the WLAN depending on the method you have chosen. These include the keys, filters, ACLs, and parameter maps as applicable to the selected authentication method.

Follow these steps to configure authentication configurations:

Procedure

1. If you have selected **Local Web Authentication** as the authentication method, configure the following:

a) In the **WLAN > Parameter Map** tab, configure the parameter map for the WLAN. A parameter map sets parameters that can be applied to subscriber sessions during authentication.

1. In the **Global Configuration** section, configure the global parameter map.
2. Enter an IPv4 or IPv6 address to configure a virtual IP address for redirecting the clients to the login page of the controller.
3. From the **Trustpoint** drop-down list, select the trustpoint for HTTPS login page. The trustpoint corresponds to the device certificate the controller will use in conjunction with the virtual IP and hostname.
4. In the **WLAN Specific Configuration** section, either create a new parameter map for the WLAN, or select an existing parameter map from the drop-down list.

b) In the WLAN > Local Users / Flex tab, configure a Flex profile and enter the username in the local database to establish a username-based authentication system.

1. In the **Flex Profile** section, enter the name of the new flex profile and the native VLAN ID.
2. To use an already existing Flex profile, click on **Select Existing** to choose a profile from the drop-down list and enter the native VLAN ID.
3. In the **Local Users** section, enter the user name to be saved.
4. From the **Password Encryption** drop-down list, choose if you want the password to be unencrypted or encrypted.
5. In the **Password** field, specify the password the user must enter to gain access to the switch. The password must be from 1 to 25 characters and can contain embedded spaces.
6. Click on the **+** sign to add the credentials to the database. Add as many user credentials as required.

2. If you have selected **External Web Authentication** as the authentication method, configure the following:

a) In the **WLAN > Parameter Map** tab, configure the parameter map for the WLAN.

1. In the **Global Configuration** section, configure the global parameter map.
2. Enter an IPv4 or IPv6 address to configure the virtual IP address of the external web authentication login page to which the guest users are redirected.
3. From the **Trustpoint** drop-down list, select the trustpoint for HTTPS login page. The trustpoint corresponds to the device certificate the controller will use in conjunction with the virtual IP and hostname.
4. In the **WLAN Specific Configuration** section, either create a new parameter map for the WLAN, or select an existing parameter map from the drop-down list.
5. To create a new parameter map, enter the parameter-map name.
6. In the **Redirect URL for login** field, enter the URL of the external server that will host the authentication page for login.
7. In the **Portal IPV4 Address** field, enter the IPv4 address of the external server to send redirects. If the external server uses an IPv6 address, in the **Portal IPV6 Address** field, enter the IPv6 address of the portal to send redirects.

b) In the **WLAN > ACL / URL Filter** tab, configure the ACL rules and the URL filter list.

1. In the **Flex Profile** section, enter the name of the new flex profile and the native VLAN ID.
2. To use an already existing Flex profile, click on **Select Existing** to choose a profile from the drop-down list and enter the native VLAN ID.
3. In the **Pre Auth ACL** section, enter the name of the ACL.

4. In the **IP address** field, enter the source IP address and the destination IP address. This will configure the ACL to permit packet transfer from and to the specified IP address. You can add as many IP addresses as required.
5. In the **URL Filter** section, enter a name for the URL Filter list that you are creating.
6. Click on **Add** to add the URLs.
7. Specify the URL to be added to the list and its preference.
8. Use the slider to set the list action to **Permit** or **Deny** the URLs.
9. Click **Save**.

   You can add as many URLs to the list as required.

c) To add a new AAA server to the list, click on Add New Server and enter the IP address and server-key.
d) To use an already configured AAA server list, click on **Use Existing** and select the appropriate list from the drop-down.

3. If you have selected **Central Web Authentication** as the authentication method, configure the following:
   a) In the **WLAN > AAA/ACL** tab, configure the AAA server list and ACL for the WLAN.
   b) In the **AAA Configuration** section, select any of the available AAA servers to add to the WLAN. This will be the server where the clients will get authenticated.
   c) To add a new AAA server to the list, click on **Add New Server** and enter the IP address and server-key.
   d) To use an already configured AAA server list, click on **Use Existing** and select the appropriate list from the drop-down.
   e) In the **Flex Profile** section, enter the name of the new flex profile and the native VLAN ID.
   f) To use an already existing Flex profile, click on **Select Existing** to choose a profile from the drop-down list and enter the native VLAN ID.
   g) In the **ACL List** section, enter the name of the ACL. This ACL will contain the rules regarding URLs that can be accessed by the client and should match the name configured on the RADIUS server.

Configure tags on WLAN

Configure tags on WLAN to define properties through associated policies that are inherited by clients and access points.

Configure tags on WLAN to define properties through associated policies that are inherited by clients and access points.

A Tag's property is defined by the policies associated to it. This property is in turn inherited by an associated client/AP. There are various type of tags, each associated to different profiles.

Follow these steps to configure tags on WLAN:

Procedure

---

1. In the **Site Configuration** section, either enter a site tag to be added, or select an existing site tag from the drop-down list. You can add as many tags as required. In FlexConnect mode, the site tag contains the AP join profile and the Flex profile.
2. In the **Policy Tag** section, either enter a policy tag to be added, or select an existing policy tag from the drop-down list. You can add as many tags as required. The policy tag constitutes mapping of the WLAN profile to the policy profile. The WLAN profile defines the wireless characteristics of the WLAN. The policy profile defines the network policies and the switching policies for the client.
3. In the **RF Tag** section, either enter an RF tag to be added, or select an existing RF tag from the drop-down list. You can add as many tags as required. The RF tag contains the 2.4 GHz and 5 GHz RF profiles.

The tags are configured on the WLAN with their associated policies and profiles.

AP Provision

Configure access point provisioning to associate APs with WLAN tags and locations.

Access point provisioning allows you to associate discovered or manually added APs with specific WLAN configurations by applying policy, site, and RF tags.

Once the Wireless network and RF characteristics are set up, access points can be added to the local site either using static AP MAC address assignment or by assigning already joined APs to a specific location.

Follow these steps to provision access points and apply tags to associate them with the WLAN:

Procedure

1. To add tags and associate APs to the WLAN, click on **AP Provisioning** from the left panel.
2. The APs already discovered by the controller are listed in the **Provision Joined APs** tab. You can select the APs to be associated to the WLAN from this table.
3. To add tags to the selected APs, select the appropriate Policy Tag, Site Tag, and RF Tag from the respective drop-down lists. Click on **Add** to apply the tags.
4. To add APs manually, click on the **Pre-provision APs** tab. You can either add individual MAC addresses of the APs or upload a CSV file with the AP MAC addresses listed. The added APs will be listed in the table below.
5. Select the APs to be associated to the WLAN from this table.
6. To add tags to the selected APs, select the appropriate Policy Tag, Site Tag, and RF Tag from the respective drop-down lists. Click on **Add** to apply the tags.

   A table of all the APs and the tags added to them is displayed in the **Selected APs** tab.
7. Click **Apply**.

   This will create a WLAN in FlexConnect mode with the authentication method, authentication filters, tags, and APs configured on it.

The access points are successfully provisioned and associated with the WLAN configuration. A WLAN in FlexConnect mode is created with the specified authentication method, authentication filters, tags, and APs.

Guest CWA mode

Explains a secure wireless guest network implementation using Central Web Authentication for accountable internet access to visitors.

Guest CWA mode is a wireless network security implementation that

- provides secure and accountable internet access to visitors using Central Web Authentication
- uses the enterprise's existing wireless and wired infrastructure to the maximum extent, and
- comprises two controllers - a Guest Foreign and a Guest Anchor.

Guest CWA mode implementation

The Guest mode addresses the need to provide internet access to guests in a secure and accountable manner with Central Web Authentication as the security method.



Controller types

Describes the types of controllers available for WLAN Guest CWA mode configuration.

A controller type is a WLAN configuration category that

- determines how controllers manage guest client connections and traffic
- defines the role each controller plays in the guest access network, and
- enables proper traffic flow and security implementation for guest WLAN services.

Controller type options

To configure a WLAN for Guest CWA mode, select the type of controller configuration you want to set up on the device from the left panel.

The controller type options are:

- Foreign: A Foreign is a controller in the WLAN that exists in the enterprise. A client sends a connection request to a Foreign controller to join the WLAN. It is a dedicated guest WLAN or SSID and is implemented throughout the campus wireless network wherever guest access is required. The Foreign controller manages the anchor controllers.
- Anchor: An Anchor is a controller or group of controllers in a WLAN that manage traffic within the network for a guest client. It provides internal security by forwarding the traffic from a guest client to a Cisco Wireless Controller in the demilitarized zone (DMZ) network.

Configure WLAN profile and policy

Configure WLAN profile and policy settings to define wireless network properties and network policies for client switching.

Configure WLAN profile and policy settings to establish wireless network connectivity and define network switching policies for clients.

After selecting the Authentication method, click on **WLAN** on the left panel to enter the WLAN profile and policy details.

The WLAN profile defines the properties of a WLAN such as Profile Name, Status, WLAN ID, L2 and L3 Security parameters, AAA Server associated with this SSID and other parameters that are specific to a particular WLAN. The policy profile defines the network policies and the switching policies for a client (with the exception of QoS), which constitute the AP policies as well.

Procedure

1. In the **Network Name** section, enter a **WLAN profile name**, which is a unique name for your wireless network. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.
2. Enter a valid **SSID** for the WLAN. A valid SSID can be up to 32 characters and can contain spaces. A valid SSID can be ASCII characters from 0 to 31, with leading and trailing spaces. This is the broadcast name for your WLAN.
3. Enter the **WLAN ID**.
4. In the **WLAN Policy** section, enter the **Policy Profile name**. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.
5. Select the **VLAN** to be associated with the Policy Profile from the drop-down list..

6. To select an existing Policy Profile for the WLAN, click on **Select Existing** and choose a **Policy Profile** from the drop-down list..
7. If you have selected **Foreign**, in the **Mobility Anchors** section, select the IP address of an available controller to assign it as the mobility anchor for the WLAN. This will extend the configurations on the Foreign controller onto the anchor controllers as well.

The WLAN profile and policy are configured with the specified network name, SSID, WLAN ID, and policy settings. The wireless network is ready for client connections with the defined switching policies.

Configure authentication for guest access

Configure authentication settings for guest access mode using Central Web Authentication.

Configure AAA server settings and access control lists to enable guest authentication through Central Web Authentication for WLAN access.

For the Guest access mode, the authentication method is Central Web Authentication. This configuration allows guest users to authenticate through a centralized web portal before gaining network access.

Procedure

---

1. In the **WLAN > AAA/ACL** tab, configure the AAA server list and ACL for the WLAN.
2. In the **AAA Configuration** section, select any of the available AAA servers to add to the WLAN.

   This will be the server where the clients will get authenticated.
3. To add a new AAA server to the list, click on **Add New Server** and enter the IP address and server-key.
4. To use an already configured AAA server list, click on **Use Existing** and select the appropriate list from the drop-down.
5. In the **ACL List** section, enter the name of the ACL.

   This ACL will contain the rules regarding URLs that can be accessed by the client and should match the name configured on the RADIUS server.

The WLAN is now configured with AAA authentication settings for guest access, allowing clients to authenticate through the specified AAA servers and apply the configured access control rules.

Configure tags on WLAN

Configure tags on WLAN to define properties inherited by associated clients and access points.

Configure tags to define properties that are inherited by associated clients and access points through various policy profiles.

A Tag's property is defined by the policies associated to it. This property is in turn inherited by an associated client/AP. There are various type of tags, each associated to different profiles.

Procedure

---

1. In the **Site Configuration** section, either enter a site tag to be added, or select an existing site tag from the drop-down list. You can add as many tags as required.
2. In the **Policy Tag** section, either enter a policy tag to be added, or select an existing policy tag from the drop-down list. You can add as many tags as required. The policy tag constitutes mapping of the WLAN profile to the policy profile. The WLAN profile defines the wireless characteristics of the WLAN. The policy profile defines the network policies and the switching policies for the client.

3. In the **RF Tag** section, either enter an RF tag to be added, or select an existing RF tag from the drop-down list. You can add as many tags as required. The RF tag contains the 2.4 GHz and 5 GHz RF profiles.

Tags are configured on the WLAN with site configuration, policy, and RF tags that define the properties inherited by associated clients and access points.

AP provision

Configure access point provisioning for wireless networks by adding tags and associating APs to the WLAN.

This task enables you to add access points to the local site and associate them with the wireless LAN configuration by applying appropriate tags.

Once the Wireless network and RF characteristics are set up, access points can be added to the local site either using static AP MAC address assignment or by assigning already joined APs to a specific location.

If you have selected **Foreign**, click on **AP Provisioning** from the left panel to add tags and associate APs to the WLAN.

Procedure

---

1. The APs already discovered by the controller are listed in the **Provision Joined APs** tab. You can select the APs to be associated to the WLAN from this table.
2. To add tags to the selected APs, select the appropriate Policy Tag, Site Tag, and RF Tag from the respective drop-down lists. Click on **Add** to apply the tags.
3. To add APs manually, click on the **Pre-provision APs** tab. You can either add individual MAC addresses of the APs or upload a CSV file with the AP MAC addresses listed. The added APs will be listed in the table below.
4. Select the APs to be associated to the WLAN from this table.
5. To add tags to the selected APs, select the appropriate Policy Tag, Site Tag, and RF Tag from the respective drop-down lists. Click on **Add** to apply the tags.

   A table of all the APs and the tags added to them is displayed in the **Selected APs** tab.

6. Click **Apply**.

   This will create a WLAN in Guest CWA mode with the authentication method, mobility anchors, authentication filters, tags, and APs configured on it.

The access points are successfully provisioned and associated with the WLAN configuration. A WLAN in Guest CWA mode is created with the specified authentication method, mobility anchors, authentication filters, tags, and APs.

Create WLANs (GUI)

Configure WLANs using the GUI interface.

Create wireless local area networks (WLANs) to enable network connectivity and manage wireless access configurations through the GUI interface.

Use the GUI interface to create and configure WLANs when you need to establish wireless network connectivity. This method provides a visual interface for WLAN configuration and management.

Procedure

---

1. In the **Configuration** > **Tags & Profiles** > **WLANs** page, click **Add**.

   The **Add WLAN** window is displayed.

2. Under the **General** tab and **Profile Name** field, enter the name of the WLAN. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.
3. Click **Save & Apply to Device**.

The WLAN is created and applied to the device. The new WLAN appears in the WLANs list and is ready for use.

Create WLANs (CLI)

Create WLANs using commands.

Create wireless local area networks (WLANs) to provide wireless network access for clients.

You can create SSID using GUI or CLI. However, we recommend that you use CLI to create SSID. By default, the WLAN is disabled after creation.

Follow these steps to create WLANs using CLI commands:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the WLAN name and ID.

   Example

   ```
   Device(config)# wlan profile-name wlan-id [ssid]
   ```

   Example

   ```
   Device(config)# wlan profile-name wlan-id [ssid] mywlan 34 mywlan-SSID
   ```

   - For the *profile-name*, enter the profile name. The range is from 1 to 32 alphanumeric characters.
   - For the *wlan-id*, enter the WLAN ID. The range is from 1 to 512.
   - For the *ssid*, enter the Service Set Identifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID.

   Note

   - You can create SSID using GUI or CLI. However, we recommend that you use CLI to create SSID.
   - By default, the WLAN is disabled.

3. Return to privileged EXEC mode.

Example

```
Device# end
```

Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The WLAN is created with the specified profile name, WLAN ID, and SSID. The WLAN is disabled by default and must be enabled separately for use.

Delete WLANs (GUI)

Configure WLAN deletion using the GUI interface.

Remove unwanted or obsolete WLANs from your network configuration to maintain optimal performance and security.

Use this procedure when you need to remove WLANs that are no longer required or need to be replaced with new configurations. This task can be performed on single or multiple WLANs simultaneously.

Procedure

---

1. In the **Configuration** > **Tags & Profiles** > **WLANs** page, check the check box adjacent to the WLAN you want to delete.

   To delete multiple WLANs, select multiple WLANs check boxes.
2. Click **Delete**.
3. Click **Yes** on the confirmation window to delete the WLAN.

The selected WLAN(s) are permanently removed from the system and will no longer appear in the WLANs list.

Delete WLANs (CLI)

Configure WLAN deletion using CLI commands to remove WLAN profiles from the wireless network configuration.

Delete WLAN profiles that are no longer needed or require reconfiguration in the wireless network environment.

Use this procedure when you need to remove WLAN profiles from the wireless controller configuration. Deleting a WLAN removes the profile and associated settings from the system.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Delete the WLAN.

Example

```
Device(config)# no wlan wlan-profile-name wlan-id  [ssid]
```

```
Device(config)# no wlan test2
```

The arguments are:

- The *wlan-name* is the WLAN profile name.
- The *wlan-id* is the WLAN ID.
- The *ssid* is the WLAN SSID name configured for the WLAN.

> **Note**
>
> If you delete a WLAN that is part of an AP group, the WLAN is removed from the AP group and from the AP's radio.

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

The WLAN profile is deleted from the wireless controller configuration. If the WLAN was associated with an AP group, it is automatically removed from the group and the AP's radio interface.

WLAN search methods (CLI)

Lists commands for searching WLANs using the command-line interface.

To verify the list of all WLANs configured on the controller, use this command:

```
Device# show wlan summary
Number of WLANs: 4

WLAN Profile Name                     SSID                           VLAN
Status
-------------------------------------------------------------------------------
1    test1                            test1-SSID                     137  UP
3    test2                            test2-SSID                     136  UP
2    test3                            test3-SSID                     1    UP
45   test4                            test4-SSID                     1
DOWN
```

To use wild cards and search for WLANs, use this **show** command:

```
Device# show wlan summary | include test-wlan-ssid
1    test-wlan                        test-wlan-ssid                  137
 UP
```

Enable WLANs (GUI)

Enable WLANs using the graphical user interface.

Enable wireless local area networks (WLANs) to allow wireless client connections and network access through the graphical user interface.

Use this task when you need to activate WLANs that have been configured but are currently disabled, allowing wireless devices to connect to your network.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. On the **WLANs** page, click the WLAN name.
3. In the **Edit WLAN** window, toggle the **Status** button to **ENABLED**.
4. Click **Update & Apply to Device**.

The WLAN is ENABLED and wireless clients can now connect to the network through this WLAN.

Enable WLANs (CLI)

Configure WLANs using commands.

Enable previously configured WLANs to make them operational and available for client connections.

WLANs must be explicitly enabled after configuration to allow wireless clients to connect. This procedure activates a specific WLAN profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. Enter WLAN configuration submode.

   Example

   ```
   Device(config)# WLAN profile-name
   ```

   Example

   ```
   Device(config)# WLAN test4
   ```

   The *profile-name* is the profile name of the configured WLAN.

3. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

4. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

The WLAN is now enabled and operational, allowing wireless clients to connect to the network.

Disable WLANs (GUI)

Configure WLAN settings to disable wireless local area networks using the GUI interface.

Disable WLANs to prevent wireless network access when maintenance is required or to temporarily suspend wireless services.

Use this procedure when you need to temporarily disable wireless networks without deleting the WLAN configuration. The WLAN settings remain intact and can be re-enabled later.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. In the **WLANs** window, click the WLAN name.
3. In the **Edit WLAN** window, set the **Status** toggle button as **DISABLED**.
4. Click **Update & Apply to Device**.

The WLAN is DISABLED and wireless clients can no longer connect to this network. The WLAN configuration is preserved and can be re-enabled when needed.

Disable WLANs (CLI)

Disable WLANs using commands.

Disable specific WLANs to prevent wireless client access when needed for maintenance or security purposes.

You may need to disable WLANs temporarily for maintenance, troubleshooting, or security reasons. This procedure shows how to disable a WLAN using the CLI and verify the change.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enter WLAN configuration submode.

Example

```
Device(config)# wlan profile-name
```

Example

```
Device(config)# wlan test4
```

The *profile-name* is the profile name of the configured WLAN.

3. Disable the WLAN.

Example

```
Device(config-wlan)# shutdown
```

4. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

5. Display the list of all WLANs configured on the device.

Example

```
Device# show wlan summary
```

You can search for the WLAN in the output to verify it has been disabled.

The specified WLAN is now disabled and will not accept wireless client connections until re-enabled.

Configure general WLAN properties (CLI)

Configure general WLAN properties using commands.

Configure media stream, broadcast SSID, and radio properties for a WLAN.

You can configure these properties: Media stream, Broadcast SSID, and Radio settings for optimal WLAN performance and visibility.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enter WLAN configuration submode.

Example

```
Device(config)# wlan profile-name
```

Example

```
Device(config)# wlan test4
```

The *profile-name* is the profile name of the configured WLAN.

3. Disable the WLAN.

Example

```
Device(config-wlan)# shutdown
```

4. Broadcast the SSID for this WLAN.

Example

```
Device(config-wlan)# broadcast-SSID
```

5. Configure the WLAN radio policy for dot11 radios.

Example

```
Device(config-wlan)# dot11bg 11g
```

Also see the section: Configuring a WLAN Radio Policy.

6. Enable multicast VLANs on this WLAN.

Example

```
Device(config-WLAN)# media-stream multicast-direct
```

7. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

8. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

The WLAN is now configured with the specified general properties including media stream, broadcast SSID, and radio settings.

Configure advanced WLAN properties (CLI)

Configure advanced WLAN properties using commands.

Configure advanced WLAN properties to customize wireless network behavior and security settings.

Advanced WLAN properties provide additional configuration options for wireless networks, including coverage hole detection, client association limits, access control, and peer-to-peer blocking settings. These configurations help optimize wireless performance and security based on specific network requirements.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter WLAN configuration submode.

   Example

   ```
   Device(config)# wlan profile-name
   ```

   Example

   ```
   Device(config)# wlan test4
   ```

   The *profile-name* is the profile name of the configured WLAN.

3. Enable coverage hole detection for this WLAN.

   Example

   ```
   Device(config-wlan)# chd
   ```

4. Enable support for Aironet IEs for this WLAN.

   Example

   ```
   Device(config-wlan)# ccx aironet-iesupport
   ```

5. Set the maximum number of clients that can be configured on a WLAN. Configure limits for clients per WLAN, clients per AP, or clients per AP radio.

   Example

   ```
   Device(config-wlan)# client association limit {clients-per-WLAN | ap
   clients-per-AP-per-WLAN | radio clients-per-AP-radio-per-WLAN}
   ```

   Example

   ```
   Device(config-wlan)# client association limit AP 400
   ```

6. Configure the IPv4 WLAN web ACL.

   Example

   ```
   Device(config-wlan)# ip access-group web ACL-name
   ```

   Example

   ```
   Device(config-wlan)# ip access-group web test-ACL-name
   ```

   The *ACL-name* specifies the user-defined IPv4 ACL name.

7. Configure peer-to-peer blocking parameters.

   Example

   ```
   Device(config-wlan)# peer-blocking [allow-private-group | drop |
   forward-upstream]
   ```

   Example

   ```
   Device(config-WLAN)# peer-blocking drop
   ```

   The keywords are:

   - **allow-private-group**: Enables peer-to-peer blocking on the Allow Private Group action.
   - **drop**: Enables peer-to-peer blocking on the drop action.
   - **forward-upstream**: No action is taken and forwards packets to the upstream.

   > **Note**
   >
   > The **forward-upstream** option is not supported for Flex local switching. Traffic is dropped even if this option is configured. Also, peer to peer blocking for local switching SSIDs are available only for the clients on the same AP.

8. Set the channel scan defer priority and defer time.

   Example

   ```
   Device(config-WLAN)# channel-scan {defer-priority {0-7} | defer-time {0-6000}}
   ```

   Example

   ```
   Device(config-WLAN)# channel-scan defer-priority 6
   ```

   The arguments are:

   - **defer-priority**: Specifies the priority markings for packets that can defer off-channel scanning. The range is from 0 to 7. The default is 3.

- **defer-time**: Deferral time in milliseconds. The range is from 0 to 6000. The default is 100.

9. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

The advanced WLAN properties are now configured with the specified settings for coverage hole detection, client limits, access control, peer-to-peer blocking,broadcast filtering, AAA override, session timeout, exclusion list management, and diagnostic channel support.

Configure advanced WLAN properties (GUI)

Configure advanced WLAN properties using the GUI to enable features such as coverage hole detection, Aironet IE, diagnostic channel, and 802.11v BSS Transition support.

Configure advanced WLAN properties to enable enhanced wireless network features including coverage optimization, client connection limits, BSS transition support, and 802.11ax capabilities.

Advanced WLAN properties provide additional configuration options to optimize wireless network performance and enable specific features for client management and network diagnostics.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **WLANs** .
2. Click **Add**.
3. Under the **Advanced** tab, check the **Coverage Hole Detection** check box.
4. Check the **Aironet IE** check box to enable Aironet IE on the WLAN.
5. Check the **Diagnostic Channel** check box to enable diagnostic channel on the WLAN.
6. From the **P2P Blocking Action** drop-down list, choose the required value.
7. Set the **Multicast Buffer** toggle button as enabled or diabled.
8. Check the **Media Stream Multicast-Direct** check box to enable the feature.
9. In the **Max Client Connections** section, specify the maximum number of client connections for the following:
   - In the **Per WLAN** field, enter a value. The valid range is between 0 and 10000.
   - In the **Per AP Per WLAN** field, enter a value. The valid range is between 0 and 400.
   - In the **Per AP Radio Per WLAN** field, enter a value. The valid range is between 0 and 200.
10. In the **11v BSS Transition Support** section, perform the following configuration tasks:
    a) Check the BSS Transition check box to enable 802.11v BSS Transition support.
    b) In the **Disassociation Imminent** field, enter a value. The valid range is between 0 and 3000.
    c) In the **Optimized Roaming Disassociation Timer** field, enter a value. The valid range is between 0 and 40.
    d) Select the check box to enable:
       - BSS Max Idle Service
       - BSS Max Idle Protected
       - Disassociation Imminent Service

- Directed Multicast Service
- Universal Admin
- Load Balance
- Band Select
- IP Source Guard

11. In the 11ax section, perform these configuration tasks:

a) Select the check box to enable these parameters:

- Check the **Enable 11ax** checkbox to enable 802.11ax operation status on the WLAN.
- Check the **Downlink OFDMA** and **Uplink OFDMA** check boxes to enable downlink and uplink connections that use OFDMA.

  Orthogonal Frequency Division Multiple Access (OFDMA) is a channel access mechanism that assures contention-free transmission to multiple clients in both the downlink (DL) and uplink (UL) within a respective single transmit opportunity.
- Check the **Downlink MU-MIMO** and **Uplink MU-MIMO** check boxes to enable downlink and uplink connections that use MU-MIMO.

  With Multiuser MIMO (MU-MIMO), an AP can use its antenna resources to transmit multiple frames to different clients, all at the same time and over the same frequency spectrum.
- Enable the target wake up time configuration on the WLAN by checking the **BSS Target Wake Up Time** checkbox.

  Target wake up time allows an AP to manage activity in the Wi-Fi network to minimize medium contention between stations, and to reduce the required amount of time that a station in the power-save mode needs to be awake. This is achieved by allocating stations to operate at non-overlapping times, and/or frequencies, and concentrate the frame exchanges in predefined service periods.
- Check the **Universal Admin** check box to enable Universal Admin support for the WLAN.
- Enable OKC on the WLAN by checking the **OKC** check box. Opportunistic Key Caching (OKC) allows the wireless client and the WLAN infrastructure to cache only one Pairwise Master Key (PMK) for the lifetime of the client association with this WLAN, even when roaming between multiple APs. This is enabled by default.
- Check the **Load Balance** check box to enable Aggressive Client Load Balancing. This allows lightweight access points to load balance wireless clients across access points.
- Check the **Band Select** check box to enable band selection for the WLAN. Band selection enables client radios that are capable of dual-band (2.4 and 5-GHz) operations to move to a less congested 5-GHz access point. The 2.4-GHz band is often congested with interference from other electronic devices as well as co-channel interference from other access points. Band selection helps prevent these sources of interference and improve overall network performance.
- Enable IP Source Guard on the WLAN by checking the **IP Source Guard** check box. IP Source Guard (IPSG) is a Layer 2 security feature that prevents the wireless controller from forwarding the packets with source IP addresses that are not known to it.

b) From the **WMM Policy** drop-down list, choose the policy as **Allowed**, **Disabled**, or **Required**. By default, the WMM policy is **Allowed**. Wi-Fi Multimedia (WMM) is used to prioritize different types of traffic.

- **Disabled**: Disables WMM on the WLAN.
- **Required**: Requires client devices to use WMM. Devices that do not support WMM cannot join the WLAN.
- **Allowed**: Devices that cannot support WMM can join the WLAN but will not benefit from the 802.11n rates.

c) From the **mDNS** drop-down list, choose **Bridging**, **Gateway**, or **Drop**. Multicast DNS (mDNS) provides the ability to perform DNS-like operations on the local link in the absence of any conventional Unicast DNS server.

- **Bridging**: Packets with mDNS multicast IP and multicast mac will be sent on multicast CAPWAP tunnel.
- **Gateway**: All ingress mDNS packets received from the wired network on a L3 interface (SVI or physical) would be intercepted by the Controller software and processed.
- **Drop**: All ingress mDNS packets will be dropped.

12. In the **Off Channel Scanning Defer** section, choose the appropriate **Defer Priority** values and then specify the required Scan Defer Time value in milliseconds.
13. In the **Assisted Roaming (11k)** section, choose the appropriate status for:
    - Prediction Optimization
    - Neighbor List
    - Dual-Band Neighbor List
14. In the **DTIM Period (in beacon intervals)** section, specify a value for 802.11a/n and 802.11b/g/n radios. The valid range is from 1 to 255.
15. Click **Apply to Device**.

The WLAN profile is updated with advanced properties, and features such as coverage optimization, client connection limits, enhanced roaming, and diagnostics are enabled for improved network performance.

Configure WLAN radio policy (GUI)

Configure WLAN radio policy using the GUI interface.

Configure a WLAN radio policy to define the radio bands and settings for wireless network operation through the graphical user interface.

Use this procedure when you need to set up radio policies for WLANs to control which radio bands (2.4 GHz or 5 GHz) are used and configure related settings for optimal wireless network performance.

Procedure

---

1. On the **Configuration** > **Tags & Profiles** > **WLANs** page, click **Add** to create WLANs.
2. In the **General** tab, enter a **Profile Name**, which is a unique name of the your wireless network.

   The name can be ASCII characters from 32 to 126, without leading and trailing spaces.
3. Enter a valid **SSID** for the WLAN.

   A valid SSID can be up to 32 characters and can contain spaces. A valid SSID can be ASCII characters from 0 to 31, with leading and trailing spaces. This is the broadcast name for your WLAN.
4. Enter the **WLAN ID**.

   The valid range for the different models are listed below:

Model	WLAN ID Range
Cisco Catalyst 9800-80 Wireless Controller	1-4096
Cisco Catalyst 9800-CL Wireless Controller	1-4096

Model	WLAN ID Range
Cisco Catalyst 9800-40 Wireless Controller	1-4096
Cisco Catalyst 9800-L Wireless Controller	1-4096
Cisco Embedded Wireless Controller for an AP	1-16

5. Set the WLAN **Status** to Enabled.
6. To broadcast the SSID of the WLAN, set the status of Broadcast SSID to enabled.

   By default, this is disabled.
7. In the **Radio Policy** section, enable the desired radio band for the WLAN.
   - 2.4ghz: Configures the policy on the 2.4 GHz radio.
   - 5ghz: Configures the policy on the 5 GHz radio.
8. If you enable the 5 Ghz radio band, select the radio slot to broadcast the WLAN on.

   The options are slot 0, slot 1, and slot 2. You can select multiple slots for the WLAN.
9. From the **802.11b/g Policy** drop-down list, choose the radio policy from these options:
   - 802.11g only
   - 802.11b/g
10. Click **Apply to Device**.

The WLAN radio policy is configured and applied to the device with the specified radio bands and settings.

Configure a WLAN radio policy (CLI)

Configure WLAN radio policy using commands to enable the WLAN on specific radio bands and slots.

Configure WLAN radio policy to control which radio bands and slots the WLAN operates on.

Use this procedure to configure WLAN radio policy settings when you need to specify which radio bands (2.4 GHz, 5 GHz, or 6 GHz) and which radio slots the WLAN should use for wireless communication.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Enter WLAN configuration submode.

   Example

   ```
   Device(config)# wlan profile-name
   ```

Example

```
Device(config)# wlan test4
```

The *profile-name* is the profile name of the configured WLAN.

3. Disable the WLAN.

Example

```
Device(config-wlan)# shutdown
```

4. Enable the corresponding radio policy on the WLAN.

Example

```
Device(config-wlan)# radio policy dot11 {5ghz | 24ghz | 6ghz }
```

Example

```
Device(config-wlan)# radio policy dot11 5ghz
```

The options are:

- **2.4ghz**: Configures the WLAN on 2.4-GHz radio only.
- **5ghz**: Configures the WLAN on 5-GHz radio only.
- **6ghz**: Configures the WLAN on 6-GHz radio only.

5. Configure the WLAN radio policy on the slot that you choose.

Example

```
Device(config-wlan-radio-5ghz)# slot slot-number
```

Example

```
Device(config-wlan-radio-5ghz)# slot 1
```

The options are:

- **0**: Configures the WLAN on the 5GHz radio with radio slot 0 (if using 5GHz).
- **1**: Configures the WLAN on the 5GHz radio with radio slot 1.
- **2**: Configures the WLAN on the 5GHz radio with radio slot 2 (if present).

6. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

7. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

The WLAN is now configured with the specified radio policy and will operate on the designated radio band and slot.

Configure AP name advertisement in beacons (GUI)

To enable the inclusion of the AP hostname in beacon and probe responses, preventing truncation.

This feature allows the AP name to be extended to 32 characters, matching the maximum length in the controller. The configuration is available through CLI, NETCONF-YANG, and WebUI. By default, this feature is disabled.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
   The Add WLAN page is displayed.
3. Click the **Advanced** tab.
4. Check the **Advertise AP Name** check box to include the AP hostname in beacon and probe responses. This feature is disabled by default.
5. Click **Apply to Device**.

The AP includes its hostname in beacon and probe responses.

Configure AP name advertisement in beacons

To enable the inclusion of the AP hostname in beacon and probe responses, preventing truncation.

This feature allows the AP name to be extended to 32 characters, matching the maximum length in the controller. The configuration is available via CLI, NETCONF-YANG, and WebUI. By default, this feature is disabled.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter WLAN profile configuration mode for the desired WLAN.

   Example

   ```
   Device(config)# wlan wlan-prof-name wlan-id ssid-name
   ```

3. Disable the WLAN to allow configuration changes.

Example

```
Device(config-wlan)# shutdown
```

4. Enable AP name advertisement.

Example

```
Device(config-wlan)# advertise ap-name
```

5. Enable WLAN.

Example

```
Device(config-wlan)# no shutdown
```

The AP now includes its hostname in beacon and probe responses.

```
Device# configure terminal
Device(config)# wlan wlan-profile1 25 ssid1
Device(config-wlan)# shutdown
Device(config-wlan)# advertise ap-name
Device(config-wlan)# no shutdown
```

WLAN properties verification (CLI)

Lists commands to verify WLAN properties using CLI commands.

To verify the WLAN properties based on the WLAN ID, use this command:

```
Device# show wlan ID wlan-id
```

To verify the WLAN properties based on the WLAN name, use this command:

```
Device# show wlan name wlan-name
```

To verify the WLAN properties of all the configured WLANs, use this command:

```
Device# show wlan all
```

To verify the summary of all WLANs, use this command:

```
Device# show wlan summary
```

To verify the running configuration of a WLAN based on the WLAN name, use this command:

```
Device# show running-config wlan wlan-name
```

To verify the running configuration of all WLANs, use this command:

```
Device# show running-config wlan
```

Verify WLAN-VLAN information of an AP

Lists the command to verify operational WLAN-VLAN mappings per AP.

To verify the operational WLAN-VLAN mappings per AP, use this command:

```
Device# show AP name test wlan vlan
Policy tag mapping
------------------
WLAN Profile Name Name Policy    VLAN   Flex Central Switching  IPv4 ACL
IPv6 ACL
-----------------------------------------------------------------------------------
jey_cwa            pp-local-1     46       Enabled               jey_acl1
Not Configured
swaguest           pp-local-1     46       Enabled               jey_acl1
Not Configured
```

WLAN radio policy verification

Provides command to verify a WLAN radio policy configuration status.

To verify the WLAN radio policy configuration status, use this command:

```
Device# show wlan id 6 | sec Radio Bands
wpa3 enabled wlan:
Configured Radio Bands: All
Operational State of Radio Bands : All Bands Operational
```

```
Configured Radio Bands : All
Operational State of Radio Bands
    2.4ghz                      : UP
    5ghz                        : UP
    6ghz                        : DOWN (Required config: Disable WPA2 and
Enable WPA3 & dot11ax)
```

```
wpa3 not enabled WLAN :
Configured Radio Bands : All
Operational State of Radio Bands
2.4ghz : UP
5ghz   : UP
```

```
5ghz specify slot is enabled :
Configured Radio Bands
5ghz   : Enabled
Slot 0 : Enabled
Slot 1 : Disabled
Slot 2 : Disabled
```

```
Operational State of Radio Bands
5ghz   : UP
Slot 0 : Enabled
Slot 1 : Disabled
Slot 2 : Disabled
```

Verify AP name advertisement

To verify the AP name advertisement, use this command:

```
Device# show wlan id 1
WLAN Profile Name     : wlan-prof-name
================================================
Identifier                                    : 1
Description                                   :
Network Name (SSID)                           : cm71
Status                                        : Enabled
Broadcast SSID                                : Enabled
Advertise-Apname                              : Disabled
Universal AP Admin                            : Disabled
Max Associated Clients per WLAN               : 0
Max Associated Clients per AP per WLAN        : 0
Max Associated Clients per AP Radio per WLAN  : 200
.
.
.
```

203 WLAN Security

Topics:

- WPA1 and WPA2 security protocols
- AAA override
- Fallback for AAA-overridden VLAN
- How to configure WLAN security

Outlines wireless LAN security protocols and features, including WPA1, WPA2, and AAA override, and provides guidance for implementing Layer 2 security, VLAN overrides, prerequisites, and configuration procedures to secure wireless networks.

WPA1 and WPA2 security protocols

Describes Wi-Fi Protected Access (WPA, WPA1, and WPA2) standards-based security solutions from the Wi-Fi Alliance that provide data protection and access control for wireless LAN systems.

WPA1 and WPA2 are standards-based security solutions from the Wi-Fi Alliance that

- provide data protection and access control for wireless LAN systems
- offer different encryption methods, with WPA1 using TKIP and WPA2 using AES-CCMP, and
- support multiple authentication options including 802.1X, PSK, and Cisco Centralized Key Management.

WPA1 and WPA2 implementation details

WPA1 is compatible with the IEEE 802.11i standard but was implemented prior to the standard's ratification; WPA2 is the Wi-Fi Alliance's implementation of the ratified IEEE 802.11i standard.

By default, WPA1 uses Temporal Key Integrity Protocol (TKIP) and Message Integrity Check (MIC) for data protection while WPA2 uses the stronger Advanced Encryption Standard encryption algorithm using Counter Mode with Cipher Block Chaining Message Authentication Code Protocol (AES-CCMP). By default, both WPA1 and WPA2 use the 802.1X for authenticated key management. However, the following options are also available:

- PSK—When you choose PSK (also known as WPA preshared key or WPA passphrase), you need to configure a preshared key (or a passphrase). This key is used as the Pairwise Master Key (PMK) between clients and authentication server.
- Cisco Centralized Key Management uses a fast rekeying technique that enables clients to roam from one access point to another without going through the controller , typically in under 150 milliseconds (ms). Cisco Centralized Key Management reduces the time required by the client to mutually authenticate with the new access point and derive a new session key during reassociation. Cisco Centralized Key Management fast secure roaming ensures that there is no perceptible delay in time-sensitive applications, such as wireless Voice over IP (VoIP), Enterprise Resource Planning (ERP), or Citrix-based solutions. Cisco Centralized Key Management is a CCXv4-compliant feature. If Cisco Centralized Key Management is selected, only Cisco Centralized Key Management clients are supported.

  When Cisco Centralized Key Management is enabled, the behavior of access points differs from the controller 's for fast roaming in the following ways:

  - If an association request sent by a client has Cisco Centralized Key Management enabled in a Robust Secure Network Information Element (RSN IE) but Cisco Centralized Key Management IE is not encoded and only PMKID is encoded in RSN IE, then the controller does not do a full authentication. Instead, the controller validates the PMKID and does a four-way handshake.
  - If an association request sent by a client has Cisco Centralized Key Management enabled in RSN IE and Cisco Centralized Key Management IE is encoded and only PMKID is present in the RSN IE, then the AP does a full authentication. The access point does not use PMKID sent with the association request when Cisco Centralized Key Management is enabled in RSN IE.

- 802.1X+Cisco Centralized Key Management—During normal operation, 802.1X-enabled clients mutually authenticate with a new access point by performing a complete 802.1X authentication, including communication with the main RADIUS server. However, when you configure your WLAN for 802.1X and Cisco Centralized Key Management fast secure roaming, Cisco Centralized Key Management-enabled clients securely roam from one access point to another without the need to reauthenticate to the RADIUS server. 802.1X+Cisco Centralized Key Management is considered as an optional Cisco Centralized Key Management because both Cisco Centralized Key Management and non-Cisco Centralized Key Management clients are supported when this option is selected.

On a single WLAN, you can allow WPA1, WPA2, and 802.1X/PSK/Cisco Centralized Key Management/802.1X+Cisco Centralized Key Management clients to join. All of the access points on such a WLAN advertise WPA1, WPA2, and 802.1X/PSK/Cisco Centralized Key Management/ 802.1X+Cisco Centralized Key Management information elements in

their beacons and probe responses. When you enable WPA1 and/or WPA2, you can also enable one or two ciphers, or cryptographic algorithms, designed to protect data traffic. Specifically, you can enable AES and/or TKIP data encryption for WPA1 and/or WPA2. TKIP is the default value for WPA1, and AES is the default value for WPA2.

Restrictions for WPA2 and WPA3

Outlines restrictions for WPA2 and WPA3 configuration.

Consider these restrictions when configuring WPA2 and WPA3 security settings:

- You cannot enable security ft or ft-adaptive without enabling WPA2 or WPA3.
- You cannot enable ft-dot1x or ft-PSK without enabling WPA2 or WPA3.
- You cannot enable 802.1x or PSK simultaneously with SHA256 key derivation type without enabling WPA2 or WPA3 on a WLAN.
- You cannot configure PMF on WPA1 WLAN without WPA2 security.
- IOS APs do not support WPA3.

Prerequisites for Layer 2 Security

Lists prerequisites for layer 2 security.

WLANs with the same SSID must have unique Layer 2 security policies so that clients can make a WLAN selection based on the information advertised in beacon and probe responses.

The available Layer 2 security policies are as follows:

- None (open WLAN)
- WPA+WPA2

> **Note**
>
> - Although WPA and WPA2 cannot be used by multiple WLANs with the same SSID, you can configure two WLANs with the same SSID with WPA/TKIP with PSK and Wi-Fi Protected Access (WPA)/Temporal Key Integrity Protocol (TKIP) with 802.1X, or with WPA/TKIP with 802.1X or WPA/AES with 802.1X.
> - A WLAN configured with TKIP support will not be enabled on an RM3000AC module.

- Static WEP (not supported on Wave 2 APs)

AAA override

Describes the AAA Override option of a WLAN that enables configuration of WLAN for identity networking.

AAA override is a WLAN option that enables you to configure the WLAN for identity networking by applying VLAN tagging, Quality of Service (QoS), and Access Control Lists (ACLs) to individual clients based on the returned RADIUS attributes from the AAA server.

Configure AAA override (CLI)

Configure AAA override to allow RADIUS server policies to override local wireless policy settings.

Enable AAA override functionality to allow authentication servers to dynamically override wireless policy configurations on a per-client basis.

AAA override allows RADIUS servers to push policy attributes that supersede the locally configured wireless policy settings. This is useful for implementing dynamic policies based on user authentication results.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy profile-policy
   ```

   Example

   ```
   Device(config)# wireless profile policy test-wgb
   ```

3. Configure AAA policy override.

   Example

   ```
   Device(config-wireless-policy)# aaa-override
   ```

   > **Note**
   >
   > If VLAN is not pushed from the RADIUS server, the VLAN Override feature can be disabled from the RADIUS server.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

   Alternatively, you can also press Ctrl-Z to exit global configuration mode.

AAA override is now enabled for the wireless policy profile, allowing RADIUS server attributes to override local policy settings for authenticated clients.

VLAN override

Describes the VLAN override feature that requires AAA Override to be enabled under the Policy Profile.

VLAN override is an authentication feature that

- requires AAA Override to be enabled under the Policy Profile
- enables VLAN assignment from the RADIUS server using IETF RADIUS attributes or Aire-Interface-Name attribute, and
- supports VLAN ID, VLAN name, or VLAN group name assignment.

VLAN assignment methods

You can assign VLAN from the RADIUS server in two ways:

- Using IEFT RADIUS attributes 64, 65, and 81—The attribute 81 can be a VLAN ID, VLAN name, or VLAN group name. Both VLAN name and VLAN group are supported. Therefore, VLAN ID does not need to be predetermined on RADIUS.

  The RADIUS user attributes used for the VLAN ID assignment are:

  - 64 (Tunnel-Type)—Must be set to VLAN (Integer = 13).
  - 65 (Tunnel-Medium-Type)—Must be set to 802 (Integer = 6).
  - 81 (Tunnel-Private-Group-ID)—Must be set to the corresponding VLAN ID, VLAN name, or VLAN group name.

- Using Aire-Interface-Name attribute—Use this attribute to assign a successfully authenticated user to a VLAN interface name (or VLAN ID) as per the user configuration. When you use this attribute, the VLAN name is returned as a string.

> **Note**
>
> If additional 802.1x authentication is executed after VLAN override in MAB authentication, the RADIUS server must be configured to return the same value in VLAN attribute even with 802.1x authentication. VLAN overridden by MAB authentication and the VLAN assigned by 802.1x authentication should have the same value.

The VLAN ID is 12-bits, and takes a value between 1 and 4094, inclusive. Because the Tunnel-Private-Group-ID is of type string, as defined in RFC2868 for use with IEEE 802.1X, the VLAN ID integer value is encoded as a string. When these tunnel attributes are sent, it is necessary to fill in the Tag field.

Configure override VLAN for central switching (CLI)

Configure override VLAN for central switching to enable RADIUS server to dynamically assign VLANs to clients.

Enable dynamic VLAN assignment from a RADIUS server for centrally switched wireless clients.

Override VLAN configuration allows the RADIUS server to dynamically assign VLANs to wireless clients during authentication, providing flexible network segmentation based on user credentials or policies.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Define VLANs that can be pushed from the RADIUS server.

Example

```
Device(config)# vlan vlan-id
```

Example

```
Device(config)# vlan 20
```

The valid VLAN ID range is 1 to 4094.

3. Optional: Change the default name of the VLAN.

Example

```
Device(config-vlan)# name vlan-name
```

Example

```
Device(config-vlan)# name vlan_ascii
```

4. Return to privileged EXEC mode.

Example

```
Device(config-vlan)# end
```

Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The override VLAN is configured and ready to accept dynamic VLAN assignments from the RADIUS server for centrally switched clients.

Configure override VLAN for local switching (CLI)

Configure Override VLAN for local switching.

Configure VLAN name to ID mapping under flex profile for local switching scenarios.

If the VLAN name ID mapping under flex profile is newly added or updated, then the WLAN policy profiles having a matching VLAN name configured, must be shut and unshut. This is to ensure that the updated WLAN-VLAN mapping is pushed to the APs and the client receives the IP address from the intended VLAN.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a Flex profile.

Example

```
Device(config)# wireless profile flex flex-profile-name
```

Example

```
Device(config)# wireless profile flex rr-xyz-flex-profile
```

3. Define VLANs that can be pushed from the RADIUS server.

Example

```
Device(config-wireless-flex-profile)# vlan-name vlan-name
```

Example

```
Device(config-wireless-flex-profile)# vlan-name vlan_123
```

4. Configure VLAN ID.

Example

```
Device(config-wireless-flex-profile-vlan)# vlan-id vlan-id
```

Example

```
Device(config-wireless-flex-profile-vlan)# vlan-id 23
```

The valid VLAN ID range is 1 to 4096.

5. Return to privileged EXEC mode.

Example

```
Device(config-wireless-flex-profile-vlan)# end
```

Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The override VLAN is now configured for local switching with the specified flex profile, enabling VLAN name to ID mapping for wireless clients.

VLAN override on Layer 3 web authentication

Describes VLAN override functionality during layer 3 web authentication processes.

VLAN override on Layer 3 web authentication is a network authentication feature that

- enables the RADIUS server to push a new VLAN assignment during Layer 3 authentication
- triggers client re-association and DHCP renewal when VLAN changes occur, and
- maintains client state during the transition process.

Authentication process and client transition

When a client gets connected to the controller and authenticated using the RADIUS server for Local Web Authentication (LWA) and Central Web Authentication (CWA), the RADIUS server pushes back in access-accept the new VLAN. If the RADIUS server pushes back a new VLAN in the access-accept, the client goes back to IP learn state on the controller. The controller de-associates the client while maintaining the client state for 30 seconds. Once the client re-associates, the client lands immediately to the new VLAN and re-triggers a new DHCP request. The client then learns a new IP and moves to the RUN state on the controller.

The VLAN Override on Layer 3 Web authentication supports:

- Local clients
- Anchored clients
- FlexConnect central authentication, central or local switching

VLAN override verification on Layer 3 web authentication

Provides commands to verify VLAN override on layer 3 web authentication.

Verify VLAN override on layer 3 web authentication using specific show commands to display override status and statistics.

To display the VLAN override after L3 authentication, use this command:

```
Device# show wireless client mac <mac> detail
[…]
        Vlan Override after L3 Auth: True
```

To display the statistics about client, use this command:

```
Device# show wireless stats client detail
[…]
      Total L3 VLAN Override vlan change received      : 1
      Total L3 VLAN Override disassociations sent      : 1
      Total L3 VLAN Override re-associations received  : 1
      Total L3 VLAN Override successful VLAN change    : 1
      […]
      L3 VLAN Override connection timeout                              : 0
```

Fallback for AAA-overridden VLAN

Describes a wireless policy profile feature that provides backup connectivity when AAA-assigned VLANs are unavailable.

Fallback for AAA-overridden VLAN is a wireless policy profile feature that provides backup network connectivity when AAA-assigned VLANs or VLAN groups become unavailable.

Fallback for AAA-overridden VLAN implementation details

From Cisco IOS XE Bengaluru 17.6.1, fallback for AAA-overridden VLAN or VLAN groups is supported on the wireless policy profile.

A new command is introduced in the wireless policy profile to configure the Fallback for AAA-Overridden VLAN feature. In Cisco IOS XE Bengaluru 17.6.1, you cannot configure the Fallback for AAA Overridden VLAN feature using the GUI.

Feature history for fallback for AAA-overridden VLAN

Lists feature history for fallback for AAA-overridden VLAN.

This table provides release and related information for the feature explained in this module.

This feature is available in all the releases subsequent to the one in which it is introduced in, unless noted otherwise.

Table 185: Feature history for fallback for AAA-overridden VLAN

Feature	Release	Feature Information
Fallback for AAA-Overridden VLAN	Cisco IOS XE Bengaluru 17.6.1	In Cisco IOS XE Bengaluru 17.5.1 and earlier releases, if there is a network with a single AAA server dictating policies that need to be applied to a client; and this client moves across different sites that have different policy definitions. If these policy definitions are not defined on the site to which the client needs to connect, the client does not get access to the network. For example, if a client is to be given access in VLAN 1, and VLAN 1 is not defined on the site to which the client connects, the client is excluded and does not get any access to the network. The Fallback for AAA-Overridden VLAN feature is introduced to allow fallback to policy profile VLAN when the overridden VLAN is not available.

Central switching and FlexConnect mode scenarios

Describes scenarios in Central Switching and FlexConnect modes when fallback is enabled for AAA-overridden VLAN or VLAN groups.

Central switching and FlexConnect mode scenarios are network configurations that

- define client VLAN assignment behavior when fallback is enabled for AAA-overridden VLAN or VLAN groups
- determine how clients receive IP addresses based on controller and FlexConnect profile configurations, and
- specify different VLAN assignment rules for Central Switching versus FlexConnect modes.

Central switching mode behavior

FlexConnect mode behavior

If fallback is enabled for AAA-overridden VLAN or VLAN groups, you might encounter these scenarios in Central Switching and FlexConnect modes.

In Central Switching mode, VLAN assignment follows these rules:

- If the AAA server gives a VLAN policy to a client, and the VLAN ID or the VLAN name is defined in the controller, the client is assigned to the VLAN specified by the AAA server.
- If the VLAN is not defined in the controller, the client is assigned to a VLAN that is configured on the wireless policy profile.

- If a VLAN group is configured on a wireless policy profile, the VLAN, as computed by the existing VLAN group logic, is assigned to the client.
- In the VLAN group case, fallback to policy profile VLAN occurs only when all the VLANs in the group are not configured in the controller, or, if the VLAN group is not defined in the controller.
- If both, AAA-overridden VLAN and the VLAN configured on the wireless policy profile are not defined in the controller, the configuration is termed as invalid, and the client is excluded.
- If a VLAN policy is not configured, or, if the default wireless policy profile is configured, the client is assigned a VLAN from the management VLAN.

If the AAA server assigns a VLAN policy to a client configured in the FlexConnect profile, the VLAN is resolved by the controller. If the VLAN is not configured on the FlexConnect profile, the behavior of the VLAN name and the VLAN ID is made consistent, with the help of the fallback feature, and the client receives the IP address from the wireless policy profile configuration.

FlexConnect mode behavior includes these characteristics:

- If AAA VLAN is defined in FlexConnect profile, the client is assigned the AAA VLAN.
- If AAA VLAN is not defined in the FlexConnect profile, FlexConnect VLAN Central Switching is configured, and VLAN is defined in the controller, and the client is assigned AAA VLAN and is centrally switched.
- If AAA VLAN is not defined in the FlexConnect profile, FlexConnect VLAN Central Switching is configured, the VLAN is not defined in the controller, and the client is assigned a VLAN from the wireless policy profile.
- If AAA VLAN is not defined in the FlexConnect profile, and FlexConnect VLAN Central Switching is not configured, the client is assigned a VLAN from the wireless policy profile.

Configure fallback for AAA-overridden VLAN (CLI)

Configure fallback for AAA-overridden VLAN using commands.

Enable fallback to the policy profile VLAN when the AAA-overridden VLAN is not available, ensuring client connectivity continuity.

When AAA servers override client VLAN assignments but the specified VLAN becomes unavailable, fallback configuration allows clients to connect using the default policy profile VLAN instead of being denied access.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile and enter wireless policy profile configuration mode.

   Example

   ```
   Device(config)# wireless profile policy wlan-policy-profile-name
   ```

3. Allow fallback to the policy profile VLAN when the overridden VLAN is not available.

Example

```
Device(config-wireless-policy)# aaa-override vlan fallback
```

The WLAN policy profile is configured with AAA override VLAN fallback, allowing clients to use the default policy VLAN when their AAA-assigned VLAN is unavailable.

Verify fallback for AAA-overridden VLAN

Provides verification commands for fallback on AAA-overridden VLAN configuration.

To verify if the fallback for AAA-overridden VLAN is enabled, use this command:

```
Device# show wireless profile policy detailed default-policy-profile | sec AAA
 Policy Params
AAA Policy Params
  AAA Override                    : DISABLED
  NAC                             : DISABLED
  AAA Policy name                 : default-aaa-policy
  AAA Vlan Fallback               : ENABLED
```

How to configure WLAN security

Configures security settings for a wireless LAN (WLAN) to protect access and data on your network.

Configure static WEP Layer 2 security parameters (GUI)

Configure static WEP layer 2 security parameters using GUI.

Configure static WEP layer 2 security parameters to secure wireless network communication using WEP encryption keys.

Use this procedure when you need to configure WEP layer 2 security for a WLAN. Static WEP provides basic encryption for wireless communications using either 40-bit or 104-bit encryption keys.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. On the **WLANs** page, click the name of the WLAN.
3. In the **Edit WLAN** window, click the **Security** tab.
4. From the **Layer 2 Security Mode** drop-down list, select the **Static WEP** option.
5. (Optional) Check the **Shared Key Authentication** check box to set the authentication type as shared. By leaving the check box unchecked, the authentication type is set to open.
6. Set the **Key Size** as either **40 bits** or **104 bits**.
   - 40 bits: The keys with 40-bit encryption must contain 5 ASCII text characters or 10 hexadecimal characters.
   - 104 bits: The keys with 104-bit encryption must contain 13 ASCII text characters or 26 hexadecimal characters.
7. Set the appropriate **Key Index**; you can choose between 1 to 4.
8. Set the **Key Format** as either **ASCII** or **Hex**.
9. Enter a valid **Encryption Key**.
   - 40 bits: The keys with 40-bit encryption must contain 5 ASCII text characters or 10 hexadecimal characters.

- 104 bits: The keys with 104-bit encryption must contain 13 ASCII text characters or 26 hexadecimal characters.

10. Click **Update & Apply to Device**.

The WLAN is configured with static WEP layer 2 security parameters. The settings are applied to the device and WEP encryption is enabled for the wireless network.

Configure static WEP layer 2 security parameters (CLI)

Configure static WEP layer 2 security parameters using commands.

Set up static WEP encryption and authentication for WLAN security.

Static WEP provides basic wireless security through shared encryption keys. This configuration is used when legacy devices require WEP compatibility, though WPA/WPA2 is recommended for better security.

Before you begin

You must have administrator privileges.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter WLAN configuration submode.

   Example

   ```
   Device# wlan profile-name wlan-id SSID-Name
   ```

   Example

   ```
   Device# wlan test4 1 test4
   ```

   profile-name is the profile name of the configured WLAN.

   wlan-id is the wireless LAN identifier. The range is 1 to 512.

   SSID_Name is the SSID which can contain 32 alphanumeric characters.

   > **Note**
   >
   > If you have already configured this command, enter **wlan** *profile-name* command.

3. Disable fast transition over the data source on the WLAN.

Example

```
Device(config-wlan)# no security ft over-the-ds
```

4. Disable 802.11r Fast Transition on the WLAN.

Example

```
Device(config-wlan)# no security ft
```

5. Disable the WPA/WPA2 support for a WLAN.

Example

```
Device(config-wlan)# no security wpa wpa1 ciphers tkip
```

6. Configure Static WEP Key authentication with authentication type.

Example

```
Device(config-wlan)# security static-wep-key authentication open
```

The keywords are as follows:

- **static-wep-key**: Configures Static WEP Key authentication.
- **authentication**: Specifies the authentication type you can set. The values are open and shared.

7. Configure Static WEP Key encryption parameters.

Example

```
Device(config-wlan)# security static-wep-key encryption 104 ascii 0
encryption-key key-index
```

Example

```
Device(config-wlan)# security static-wep-key encryption 104 ascii 0
1234567890123 1
```

The keywords are as follows:

- **static-wep-key**: Configures Static WEP Key authentication.
- **encryption**: Specifies the encryption type that you can set. The valid values are 104 and 40. 40-bit keys must contain 5 ASCII text characters or 10 hexadecimal characters. 104-bit keys must contain 13 ASCII text characters or 26 hexadecimal characters.
- **ascii**: Specifies the key format as ASCII.
- **hex**: Specifies the key format as HEX.

8. Exit configuration mode and return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

Static WEP layer 2 security is now configured on the WLAN with the specified authentication and encryption parameters.

Configure WPA + WPA2 Layer 2 security parameters (GUI)

Configure WPA + WPA2 layer 2 security parameters using GUI.

This task configures WPA + WPA2 layer 2 security parameters to secure wireless network communications using the graphical user interface.

Use this procedure when you need to set up wireless network security using both WPA and WPA2 protocols through the GUI interface. This configuration provides enhanced security for your wireless network by supporting both protocols simultaneously.

Procedure

1. Click **Configuration** > **Tags and Profiles** > **WLANs**.
2. Click **Add** to add a new WLAN Profile or click the one you want to edit.
3. In the **Edit WLAN** window, click **Security** > **Layer2**.
4. From **Layer 2 Security Mode** drop-down menu, select **WPA + WPA2**.
5. Configure the security parameters and then click **Save and Apply to Device**.

The WPA + WPA2 layer 2 security parameters are configured and applied to the device. The WLAN profile now uses both WPA and WPA2 security protocols to protect wireless network communications.

Configure WPA + WPA2 layer 2 security parameters (CLI)

Configure WPA + WPA2 layer 2 security parameters using commands.

This task configures WPA and WPA2 layer 2 security settings to secure wireless networks using command-line interface commands.

Note

The default values for security policy WPA2 are:

- Encryption is AES.
- Authentication Key Management (AKM) is dot1x.

Before you begin

You must have administrator privileges.

Follow these steps to configure WPA + WPA2 layer 2 security parameters using commands:

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration submode.**wlan**

   Example

   ```
   Device(config)# wlan profile-name wlan-id SSID_name
   ```

   Example

   ```
   Device(config)# wlan test4 1 test4
   ```

   - *profile-name* is the profile name of the configured WLAN.
   - *wlan-id* is the wireless LAN identifier. The range is from 1 to 512.
   - *SSID_name* is the SSID that contains 32 alphanumeric characters.

   Note

   If you have already configured this command, enter **wlan** *profile-name* command.

3. Enables WPA or WPA2 support for WLAN.

   Example

   ```
   Device(config-wlan)# security wpa {akm | wpa1 | wpa2}
   ```

   ```
   Device(config-wlan)# security wpa
   ```

4. Enables WPA.

   Example

   ```
   Device(config-wlan)# security wpa wpa1
   ```

5. Specify the WPA1 cipher.

   Example

   ```
   Device(config-wlan)# security wpa wpa1 ciphers [aes | tkip]
   ```

   ```
   Device(config-wlan)# security wpa wpa1 ciphers aes
   ```

Choose one of the following encryption types:

- **aes**: Specifies WPA/AES support.
- **tkip**: Specifies WPA/TKIP support.

The default values are TKIP for WPA1 and AES for WPA2.

> **Note**
>
> You can enable or disable TKIP encryption only using the CLI. Configuring TKIP encryption is not supported in GUI.

When you have VLAN configuration on WGB, you need to configure the encryption cipher mode and keys for a particular VLAN, for example, **encryption VLAN 80 mode ciphers TKIP**. Then, you need to configure the encryption cipher mode globally on the multicast interface by entering the following command: **encryption mode ciphers TKIP**.

6. Enable or disable Cisco Centralized Key Management, 802.1x, 802.1x with SHA256 key derivation type, Fast Transition, PSK or PSK with SHA256 key derivation type.**security WPA AKM {CCKM| dot1x | dot1x-sha256 | ft | PSK |PSK-sha256}**

   Example

```
Device(config-wlan)# security wpa akm {cckm | dot1x | dot1x-sha256 | ft | psk
 | psk-sha256}
```

```
Device(config-wlan)# security wpa akm psk-sha256
```

> **Note**
>
> - You cannot enable 802.1x and PSK with SHA256 key derivation type simultaneously.
> - When you configure Cisco Centralized Key Management SSID, you must enable the **ccx aironet-iesupport** for Cisco Centralized Key Management to work.
> - WPA3 Enterprise dot1x-sha256 is supported only in local mode.

7. Enter this command to specify a preshared key, if you have enabled PSK.

   Example

```
Device(config-wlan)# security wpa psk set-key {ascii | hex} {0 | 8} password
```

```
Device(config-wlan)# security wpa psk set-key ascii 0 test
```

WPA preshared keys must contain 8 to 63 ASCII text characters or 64 hexadecimal characters.

> **Note**
>
> Avoid using the percent sign (%) in ASCII preshared keys. Some clients or access points may interpret this character specially, which can cause the client to fail to join the WLAN.

8. Enable or disable authentication key management suite for fast transition.

   Example

   ```
   Device(config-wlan)# security wpa akm ft {dot1x | psk | sae}
   ```

   ```
   Device(config-wlan)# security wpa akm ft psk
   ```

   > **Note**
   >
   > You can now choose between PSK and fast transition PSK as the AKM suite.

9. Enable WPA2.

   Example

   ```
   Device(config-wlan)# security wpa wpa2
   ```

10. Configure WPA2 cipher. **security wpa wpa2 ciphers aes**

    Example

    ```
    Device(config-wlan)# security wpa wpa2
    ```

    Example

    - **aes**: Specifies WPA/AES support.

11. **show wireless pmk-cache**

    Displays the remaining time before the PMK cache lifetime timer expires.

    If you have enabled WPA2 with 802.1X authenticated key management or WPA1 or WPA2 with Cisco Centralized Key Management authenticated key management, the PMK cache lifetime timer is used to trigger reauthentication with the client when necessary. The timer is based on the timeout value received from the AAA server or the WLAN session timeout setting.

    If you configure 802.1x with session timeout between 0 and 299, Pairwise Master Key (PMK) cache is created with a timer of 1 day 84600 seconds.

Note

- The command will show VLAN ID with VLAN pooling feature in VLAN-Override field.
- Sticky key caching (SKC) is not supported.

WPA and WPA2 layer 2 security parameters are configured for the WLAN. The wireless network is secured with the specified authentication and encryption settings.

204 Remote LANs

Topics:

- Remote LANs
- RLAN authentication fallback

Explains the concept of Remote LANs (RLANs) for authenticating and managing wired clients using a controller, covering limitations, profile and policy configuration, port and tag assignment, fast switching, and verification procedures.

Remote LANs

Introduces Remote LANs (RLANs), outlining their limitations, and provides instructions for enabling or disabling RLANs, creating and configuring RLAN and policy profiles through GUI and CLI, attaching policy tags, configuring fast switching, and verifying RLAN configuration.

A remote LAN is a network security feature that

- enables authentication of wired clients using the wireless controller
- switches client traffic between central and local switching modes, and
- treats wired client traffic as wireless client traffic for unified management.

Feature history

Table 186: Feature history for Remote LANs

Feature name	Release information	Feature description
Remote LANs	Cisco IOS XE 16.9.1	Remote LAN (RLAN) is a network security feature that allows authentication of wired clients using a wireless controller. It treats wired client traffic as wireless client traffic for unified management. The RLAN in an AP sends authentication requests for wired clients, similar to the process for centrally authenticated wireless clients.
Fast switching support for RLAN	Cisco IOS XE 17.15.2	Fast switching support for RLAN is a network optimization mechanism that reduces latency and improves throughput by enabling VLAN-tagged data packets to be switched directly between WAN and LAN ports using the AP's internal switch processor. This feature is supported on Cisco Catalyst 9105AXW APs starting with release 17.15.2.

The RLAN in an AP sends authentication requests for wired clients. Authentication of a wired client in an RLAN is similar to the process for a centrally authenticated wireless client.

For information about APs that support this feature, see https://www.cisco.com/c/en/us/td/docs/wireless/access_point/feature-matrix/ap-feature-matrix.html

Ethernet (AUX) port

By default, the second Ethernet port in Cisco Aironet 1850, 2800, and 3800 Series APs functions as a link aggregation (LAG) port. When LAG is disabled, you can use this port as an RLAN port.

These APs models use LAG port as an RLAN port:

- 1852E

- 1852I
- 2802E
- 2802I
- 3802E
- 3802I
- 3802P
- 4802

Role of Controller

- The controller acts as an authenticator, receiving Extensible Authentication Protocol (EAP) over LAN (EAPOL) messages from the wired client sent through an AP.
- The controller communicates with the configured Authentication, Authorization, and Accounting (AAA) server.
- The controller configures the LAN ports for an AP and pushes the configuration to the corresponding AP.

Fast switching support for RLAN

A fast switching feature is a network optimization mechanism that

- enables VLAN-tagged data packets to be switched directly between WAN and LAN ports using the access point's internal switch processor
- bypasses CPU processing to reduce latency and improve throughput, and
- supports efficient data handling for latency-sensitive applications such as gaming and video streaming.

On Cisco Catalyst 9105AXW APs from version 17.15.2, hardware fast switching for RLAN client traffic is supported. To configure this feature, enable fast switching on the controller and assign the appropriate VLAN IDs to LAN ports. This optimizes network performance and ensures seamless connectivity.

Starting with release 17.15.2, hardware fast switching for RLAN client traffic is supported on Cisco Catalyst 9105AXW APs. This feature enhances performance by reducing latency and improving throughput. The AP can switch VLAN-tagged data packets directly between WAN and LAN ports using its internal switch processor, which bypasses CPU processing. Fast switching is ideal for latency-sensitive applications such as gaming and video streaming and enables efficient data handling. To configure this feature, enable fast switching on the controller and assign the appropriate VLAN IDs to LAN ports. This optimizes network performance and ensures seamless connectivity.

Limitations of RLAN

Limitations of RLAN.

Review these limitations when using RLAN:

- RLAN supports only a maximum of four wired clients regardless of the AP model.
- RLAN support with Virtual Routing and Forwarding (VRF) is not available.
- In the Catalyst 9105 AXW AP operating in OEAP mode, only LAN1 and LAN2 are configured for RLAN. LAN3 is a local-only port and is enabled by default. Even if you disable LAN3 from the controller, the system skips payload processing for LAN3 because it is designated as a local-only port in OEAP mode.
- The RLAN feature is supported on Fabric.
- RLAN is supported in APs that have more than one Ethernet port.

- In RLAN (local mode - local switching mode), if you want to use the AP native VLAN for client IP, the VLAN should be configured as either **no vlan** or **vlan 1** in the RLAN policy profile. For example, if the native VLAN ID is 80, do not use the numeral 80 in the RLAN policy profile. Also, do not use VLAN name *VLANxxxx* to configure VLAN in the RLAN policy profile.

  When a new client is connected to an AP, the client details are available in the controller initially. However, after the CAPWAP DOWN/UP state, the client details are no longer listed in the controller.
- APs in local mode with central switching do not support VLAN-tagged traffic from RLAN clients, resulting in dropped traffic.
- A VLAN name (without any numerals) that is configured in the remote-lan-policy does not provide the mapped VLAN ID for central switching.
- RLAN does not support these features:
  - Central Web Authentication (CWA)
  - Quality of Service (QoS)
  - Bi-Directional Rate Limiting (BDRL)
  - Identity PSK (iPSK)

Consider these limitations when you use the AUX port in Cisco 2700 APs:

- RLAN supports the AUX port and a non-native VLAN for this port.
- Local mode supports wired client traffic on a central switch, but FlexConnect mode does not support a central switch.
- FlexConnect mode supports wired client traffic on the local switch, but not on the central switch.
- You cannot use the AUX port as a trunk port, and you cannot add switches or bridges behind the port.
- The AUX port does not support dot1x.

Configure remote LANs (RLANs)

Enable or disable all RLANs

Enable or disable all RLANs.

Enable remote LANs (RLANs) on the device to ensure network access for connected clients, or disable RLANs to restrict access.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable or disable all RLANs.

   Example

   ```
   Device(config)# [no] ap remote-lan shutdown
   ```

3. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

All RLANs on the device are enabled (or disabled), and corresponding interfaces become active (or inactive).

Create RLAN profile (GUI)

Create RLAN profile using GUI.

Define a Remote LAN (RLAN) profile so you can enable RLAN services for specific devices.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Remote LAN**.
2. Click **Add**.
3. Enter the **Profile Name**, **RLAN ID**. Enable or disable the **Status** toggle button. The profile name must contain ASCII characters from 32 to 126, without leading and trailing spaces.
4. Click **Apply to Device**.

The new RLAN profile is created and applied to the selected device.

Create RLAN profile (CLI)

Create RLAN profile using commands.

Add a new remote LAN profile for wireless device management using CLI commands.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure remote LAN profile.

   Example

   ```
   Device(config)# ap remote-lan profile-name remote-lan-profile-name rlan-id
   ```

   - *remote-lan-profile*: The remote LAN profile name. The range is from 1 to 32 alphanumeric characters.
   - *rlan-id*: The remote LAN identifier. The range is from 1 to 128.

> **Note**
>
> You can create a maximum of 128 RLANs. You cannot use the *rlan-id* of an existing RLAN while creating another RLAN.
>
> RLAN and WLAN profiles cannot have the same names. Similarly, RLAN and WLAN policy profiles must use distinct names.

The system creates the RLAN profile, making it available for configuration and assignment to APs.

```
Device# configure terminal
Device(config)# ap remote-lan profile-name rlan_profile_name 3
```

Configure RLAN profile parameters (GUI)

Configure RLAN profile parameters using GUI.

Configure Remote LAN (RLAN) profile parameters to control client associations, authentication methods, and security settings.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Remote LAN**.
2. In the **RLAN Profile** page, click **Add**.

   The **Add RLAN Profile** page is displayed.
3. In the **General** tab:
   a) Enter a **Name** and **RLAN ID** for the RLAN profile. The name must use ASCII characters with values from 32 to 126 and must not have leading or trailing spaces.
   b) Set how many clients can connect to each RLAN in the **Client Association Limit** field.
   c) To enable the profile, set the status as **Enable**.
4. In the **Security** > **Layer2** tab:
   a) To enable 802.1X for an RLAN, set the **802.1x** status as **Enabled**.

   > **Note**
   >
   > You can activate either web or 802.1x authentication list at a time.

   b) Choose the authorization list name from the **MAC Filtering** drop-down list.
   c) Choose the 802.1x for an RLAN authentication list name from the **Authentication List** drop-down list.
5. In the **Security** > **Layer3** tab:
   a) To enable web authentication for an RLAN, set the **Web Auth** status as **Enabled**.

> **Note**
>
> You can activate either web or 802.1X authentication list at a time.

b) Choose the web authentication parameter map from the **Webauth Parameter Map** drop-down list.
c) Choose the web authentication list name from the **Authentication List** drop-down list.

6. In the **Security** > **AAA** tab:
   a) Set the **Local EAP Authentication** to enabled. Also, choose the required **EAP Profile Name** from the drop-down list.
7. Save your configuration.

The configured RLAN profile is active. It is available for client associations using the specified security and authentication settings.

Configure RLAN profile parameters (CLI)

Configure RLAN profile parameters using commands.

Set up and customize Remote LAN (RLAN) profile parameters to control client access and authentication for RLAN interfaces.

Before you begin

The configurations in this section are not mandatory for an RLAN profile.

In case of central switching mode, you need to configure both central switching and central DHCP.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure remote LAN profile.

   Example

   ```
   Device(config)# ap remote-lan profile-name rlan-profile-name rlan-id
   ```

   The RLAN ID range is from 1 to 128.

3. Configure client connections per RLAN.

   Example

   ```
   Device(config-remote-lan)# client association limit client-connections
   ```

   client-connections: Is the maximum client connections per RLAN. The range is from 0 to 10000. 0 refers to unlimited.

4. Configure RLAN IP configuration commands.

Example

```
Device(config-remote-lan)# ip access-group web ipv4-acl_name
```

ipv4-acl-name: Refers to the IPv4 ACL name or ID.

5. Set EAP profile on an RLAN.

Example

```
Device(config-remote-lan)# local-auth profile-name
```

profile-name: Is the EAP profile on an RLAN.

6. Set MAC filtering support on an RLAN.

Example

```
Device(config-remote-lan)# mac-filtering mac-filter-name
```

mac-filter-name: Is the authorization list name.

7. Configure 802.1X for an RLAN.

Example

```
Device(config-remote-lan)# security dot1x authentication-list list-name
```

list-name: Is the authentication list name.

8. Configure web authentication for an RLAN.

Example

```
Device(config-remote-lan)# security web-auth authentication-list list-name
```

list-name: Is the authentication list name.

> **Note**
>
> You can activate either web or dot1x authentication list at a time.

9. Enable or disable RLAN profile.

Example

```
Device(config-remote-lan)# [no] shutdown
```

The RLAN profile is configured with your desired parameters for client access, authentication, and network policies.

Create RLAN policy profile (GUI)

Create RLAN policy profile using GUI.

Create a new Remote LAN (RLAN) policy profile for controlled wireless network access.

Procedure

1. Choose **Configuration** > **Wireless** > **Remote LAN** > **RLAN Policy**.
2. Click **Add** to create a new RLAN policy profile for the selected device..
3. In the **General** tab, enter the **Policy Name**.
4. Click **Apply to Device**.

The new RLAN policy profile is created and applied to the selected device.

Create RLAN policy profile (CLI)

Create RLAN policy profile using commands.

Configure an RLAN policy profile on a Cisco device. Wireless attributes for the RLAN are defined.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure RLAN policy profile and enter wireless policy configuration mode.

   Example

   ```
   Device(config)# ap remote-lan-policy policy-name rlan_policy_prof_name
   ```

After you complete these steps, the system creates a new RLAN policy profile. You can then assign it to remote LANs.

Configure RLAN policy profile parameters (GUI)

Configure RLAN policy profile parameters using GUI.

Configure RLAN policy profile parameters to define how Remote LAN (RLAN) networks operate.

Procedure

1. Choose **Configuration** > **Wireless** > **Remote LAN**.
2. On the **Remote LAN** page, click **RLAN Policy** tab.
3. On the **RLAN Policy** page, click the name of the **Policy** or click **Add** to create a new one.

   The **Add/Edit RLAN Policy** page is displayed.

4. In the **General** tab:

a) Enter a **Name** and **Description** for the policy profile.
b) Set **Central Authentication** to **Enabled** state.
c) Set **Central DHCP** to **Enabled** state.
d) Set the **PoE** check box to enable or disable power over Ethernet.
e) Set the status as **Enable** to activate the policy.

5. In the **Access Policies** Tab, choose the VLAN name or number from the **VLAN** drop-down list.

> **Note**
>
> When central switching is disabled, the VLAN in the RLAN policy cannot be configured as the AP's native VLAN. To use the AP native VLAN for client IP, the VLAN should be configured as either **no vlan** or **vlan 1** in the RLAN policy profile.

6. From the **Host Mode** drop-down list, choose the **Host Mode** for the remote-LAN802.1x from these options:

- Single-Host Mode: Is the default host mode. In this mode, the switch port allows only a single host to be authenticated and passes traffic one by one.
- Multi-Host Mode: The first device to authenticate opens the switch port, allowing all other devices to use the port. You do not need to authenticate other devices independently. If the authenticated device becomes unauthorized, the switch port closes.
- Multi-Domain Mode: The authenticator allows one host from the data domain and another from the voice domain. This is a typical configuration on switch ports with IP phones connected.

> **Note**
>
> - For an RLAN profile with open-auth configuration, you must map the RLAN-policy with single host mode. Mapping RLAN-policy with multi-host or multi-domain mode is not supported.
> - The controller does not assign data versus voice VLAN, based on traffic. RLAN only supports multiple VLAN assignments through 802.1x AAA override. You must create data and voice VLANs and then assign these VLANs to respective clients, based on their authentication through the 802.1x AAA override.

7. Configure IPv6 ACL or Flexible NetFlow.

- Under the **Access Policies** > **Remote LAN ACL** section, choose the **IPv6 ACL** from the drop-down list.
- Under the **Access Policies** > **AVC** > **Flow Monitor IPv6** section, check the **Egress Status** and **Ingress Status** check boxes and choose the policies from the drop-down lists.

8. Click the **Advanced** tab.

a) Configure the violation mode for Remote-LAN 802.1x from the **Violation Mode** drop-down list, choose the violation mode type from these options:

- Shutdown: Disables the port
- Replace: Removes the current session and initiates authentication for the new host. This is the default behavior.
- Protect: Drops packets with unexpected MAC addresses without generating a system message.

b) Enter the **Session Timeout (sec)** value to define the client's duration of a session.

The range is from 20 to 86400 seconds.

c) Under **AAA Policy Params** section, check the **AAA Override** check box to enable AAA override.
d) Under the **Exclusionlist Params** section, check the **Exclusionlist** check box and enter the **Exclusionlist Timeout** value.

This sets the exclusion time for a client. The range is from 0 to 2147483647 seconds. 0 refers to no timeout.

9. Save the configuration.

The RLAN policy profile is created or modified with the specified parameters, enabling precise control of RLAN behavior, access policies, and operational modes.

Configure RLAN policy profile parameters (CLI)

Configure RLAN policy profile parameters using commands.

Set up and customize a Remote LAN (RLAN) policy profile on your device.

Procedure

---

1. Configure central switching.

Example

```
Device(config-remote-lan-policy)# central switching
```

2. Configure central DHCP.

Example

```
Device(config-remote-lan-policy)# central dhcp
```

3. Set exclusion-listing on RLAN.

Example

```
Device(config-remote-lan-policy)# exclusionlist timeout timeout-value
```

timeout: Sets the duration that the client remains in the excluded state. The range is from 0 to 2147483647 seconds. 0 refers to no timeout.

4. Configures VLAN name or ID.**vlan***vlan*

Example

```
Device(config-remote-lan-policy)# vlan vlan
```

vlan: Represents the VLAN name or VLAN ID.

5. Configure AAA policy override.

Example

```
Device(config-remote-lan-policy)# aaa-override
```

6. Configure client session timeout.

   Example

   ```
   Device(config-remote-lan-policy)# session-timeout timeout in seconds
   ```

   timeout in seconds: Specifies the duration of a session. The range is from 20 to 86,400 seconds.

   > **Note**
   >
   > If the session timeout is less than 300 seconds for Dot1x clients, the session timeout is set to one day (86,400 seconds).

7. Configure host mode for remote-LAN 802.1x.

   Example

   ```
   Device(config-remote-lan-policy)# host-mode {multidomain voice domain |
   multihost | singlehost}
   ```

   voice domain: Specifies the RLAN voice domain VLAN ID. The range is from 0 to 65,535.

   You can configure these IEEE 802.1X authentication modes:

   - Multi-Domain Mode: The authenticator allows one host from the data domain and another from the voice domain. This is a typical configuration on switch ports with IP phones connected.
   - Multi-Host Mode: The first device to authenticate opens up to the switch port, so that all other devices can use the port. You need not authenticate other devices independently, if the authenticated device becomes authorized the switch port is closed.
   - Single-Host Mode: This is the default host mode. In this mode, the switch port allows only a single host to be authenticated and passes traffic one by one.

8. Configure a fabric RLAN profile by entering a name for the fabric profile.

   Example

   ```
   Device(config-remote-lan-policy)# fabric fabric-profile-name
   ```

9. Configure violation mode for Remote-LAN 802.1x.

   Example

   ```
   Device(config-remote-lan-policy)# violation-mode {protect | replace | shutdown}
   ```

   When a security violation occurs, a port is protected based on these configured violation actions:

   - Shutdown: Disables the port.
   - Replace: Removes the current session and initiates authentication for the new host. This is the default behavior.

- Protect: Drops packets with unexpected MAC addresses without generating a system message. In the single-host authentication mode, a violation is triggered when more than one device is detected in data VLAN. In a multi-host authentication mode, a violation is triggered when more than one device is detected in data VLAN or voice VLAN.

10. Enables or disables PoE.

Example

```
Device(config-remote-lan-policy)# [no] poe
```

11. Enable or disable an RLAN policy profile.

Example

```
Device(config-remote-lan-policy)# shutdown
```

12. Return to privileged EXEC mode.

Example

```
Device(config-remote-lan-policy)# end
```

The RLAN policy profile is configured as specified. Devices governed by this profile will now operate according to the parameters you set.

```
Device(config-remote-lan-policy)# central switching
Device(config-remote-lan-policy)# central dhcp
Device(config-remote-lan-policy)# exclusionlist timeout 200
Device(config-remote-lan-policy)# vlan vlan1
Device(config-remote-lan-policy)# aaa-override
Device(config-remote-lan-policy)# session-timeout 21
Device(config-remote-lan-policy)# host-mode multihost
Device(config-remote-lan-policy)# fabric fabric-profile-name
Device(config-remote-lan-policy)# violation-mode protect
Device(config-remote-lan-policy)# poe
Device(config-remote-lan-policy)# shutdown
Device(config-remote-lan-policy)# end
```

Configure policy tag and map an RLAN policy profile to an RLAN profile (CLI)

Configure policy tag and mapping an RLAN policy profile to an RLAN profile using commands.

Define a policy tag and associate your RLAN policy profile with an RLAN profile.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a policy tag and enter policy tag configuration mode.

Example

```
Device(config)# wireless tag policy policy-tag-name
```

3. Map your RLAN policy profile to an RLAN profile.

Example

```
Device(config-policy-tag)# remote-lan rlan-profile-name policy
rlan_profile_name policy rlan_policy_profile port-id port-id
```

- *remote-lan-profile-name*: The name of the RLAN profile.
- *rlan-policy-profile-name*: The name of the policy profile.
- *port-id*: The LAN port number on the AP. The port number can be any integer from 1 to 4.

4. Return to privileged EXEC mode.

Example

```
Device(config-policy-tag)# end
```

You can press **Ctrl-Z** to exit global configuration mode.

The controller maps your specified RLAN policy profile to the RLAN profile under the policy tag.

```
Device# configure terminal
Device(config)# wireless tag policy remote-lan-policy-tag
Device(config-policy-tag)# remote-lan remote-lan1 policy rlan_profile_name
port-id 2
Device(config-policy-tag)# end
```

Configure a LAN port (CLI)

Configure LAN port using commands.

Enable or disable a LAN port on an AP using commands.

Procedure

Configure a LAN port.

Example

```
Device# ap name ap-name lan port-id lan-port-id {disable | enable}
Device# ap name L2_1810w_2 lan port-id 1 enable
```

- **enable**: Enables the LAN port.
- **disable**: Disables the LAN port.

The selected LAN port on the AP is enabled or disabled as specified.

Attach policy tag to an access point (GUI)

Attach policy tag to an AP using GUI.

Assign a specific policy to an AP to enforce wireless configurations.

Procedure

1. Choose **Configuration** > **Wireless** > **Access Points**.
2. Select an AP.
3. Under the **Tags** section, use the **Policy** drop-down to select a policy tag.
4. Click **Update & Apply to Device**.

The AP is associated with the policy tag, and the new configuration is enforced.

Attach policy tag to an access point (CLI)

Attach policy tag to an AP using commands.

Assign a predefined policy tag to a specific AP.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an Ethernet MAC address for an AP and enter the AP configuration mode.

   Example

   ```
   device(config)# ap ap-ethernet-mac
   ```

3. Attach a policy tag to the AP.

   Example

   ```
   Device(config-ap-tag)# policy-tag remote-lan-policy-tag
   ```

   policy-tag-name: The name of the policy tag defined earlier.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-tag)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The AP is now associated with the specified policy tag.

```
Device# configure terminal
Device(config)# ap 00a2.891c.21e0
Device(config-ap-tag)# policy-tag remote-lan-policy-tag
Device(config-ap-tag)# end
```

Configure RLAN fast switching (CLI)

Enable RLAN fast switching to improve handoff performance for wireless clients on the AP.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP profile and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap profile default-ap-profile
   ```

3. Enable RLAN fast switching.

   Example

   ```
   Device(config-ap-profile)# [no] rlan fast-switching
   ```

   Use the **no** form of this command to disable the feature.

4. Exit RLAN configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-ap-profile)# end
   ```

RLAN fast switching is enabled for the selected AP profile, providing faster handoffs for wireless clients.

Verify RLAN configuration

Verify RLAN configuration.

Use this command to display the summary of all RLANs:

```
Device# show remote-lan summary

Number of RLANs: 1

RLAN        Profile Name                      Status
------------------------------------------------------------
1            rlan_test_1                       Enabled
```

Use this command to display the RLAN configuration by ID:

```
Device# show remote-lan id <id>

Remote-LAN Profile Name              : rlan_test_1
=================================================
Identifier                                  : 1
Status                                      : Enabled
Mac-filtering                               : Not Configured
Number of Active Clients                    : 1
Security_8021X                              : Disabled
8021.x Authentication list name             : Not Configured
Local Auth eap Profile Name                 : Not Configured
Web Auth Security                           : Disabled
Webauth Authentication list name            : Not Configured
Web Auth Parameter Map                      : Not Configured
Client association limit                    : 0
Ipv4 Web Pre Auth Acl                       : Not Configured
Ipv6 Web Pre Auth Acl                       : Not Configured
```

Use this command to display the RLAN configuration by profile name:

```
Device# show remote-lan name <profile-name>

Remote-LAN Profile Name     : rlan_test_1
================================================
Identifier                                     : 1
Status                                         : Enabled
Mac-filtering                                  : mac-auth
Number of Active Clients                       : 0
Security_8021x_dot1x                           : Enabled
8021.x Authentication list name                : Not Configured
Local Auth eap Profile Name                    : Not Configured
Web Auth Security                              : Disabled
Webauth Authentication list name               : Not Configured
Web Auth Parameter Map                         : Not Configured
Client association limit                       : 0
Ipv4 Web Pre Auth Acl                          : Not Configured
Ipv6 Web Pre Auth Acl                          : Not Configured
mDNS Gateway Status                            : Bridge
Fabric Profile Name                         : rlan-fabric-profile
```

Use this command to display detailed output for all RLANs:

```
Device# show remote-lan all

Remote-LAN Profile Name            : rlan_test_1
================================================
Identifier                         : 1
Status                             : Enabled
Mac-filtering                      : Not Configured
Number of Active Clients           : 1
Security_8021X                     : Disabled
8021.x Authentication list name    : Not Configured
Local Auth eap Profile Name        : Not Configured
Web Auth Security                  : Disabled
Webauth Authentication list name   : Not Configured
Web Auth Parameter Map             : Not Configured
Client association limit           : 0
Ipv4 Web Pre Auth Acl              : Not Configured
Ipv6 Web Pre Auth Acl              : Not Configured
```

```
Remote-LAN Profile Name            : rlan_test_2
================================================
Identifier                         : 2
Status                             : Enabled
Mac-filtering                      : Not Configured
Number of Active Clients           : 1
Security_8021X                     : Disabled
8021.x Authentication list name    : Not Configured
Local Auth eap Profile Name        : Not Configured
Web Auth Security                  : Disabled
Webauth Authentication list name   : Not Configured
Web Auth Parameter Map             : Not Configured
Client association limit           : 0
Ipv4 Web Pre Auth Acl              : Not Configured
Ipv6 Web Pre Auth Acl              : Not Configured
```

```
Device# show remote-lan policy summary
Number of Policy Profiles: 1

Profile Name                      Description                          Status

-------------------------------------------------------------------------------------------
rlan_named_pp1                Testing RLAN policy profile             Enabled
```

Use this command to display the LAN port configuration of a Cisco AP:

```
Device# show ap name <ap_name> lan port summary
LAN Port status for AP L2_1815w_1
Port ID      status       vlanId      poe
---------------------------------------------
LAN1         Enabled       20          Disabled
LAN2         Enabled       20          NA
LAN3         Disabled      0           NA
```

Use this command to display the summary of all clients:

```
Device# show wireless client summary
Number of Local Clients: 1

MAC Address       AP Name        WLAN         State    Protocol    Method
 Role
-------------------------------------------------------------------------------------
d8eb.97b6.fcc6    L2_1815w_1      1           * Run     Ethernet    None
 Local
```

Use this command to display client details for the specified username:

```
Device# show wireless client username cisco
MAC Address        AP Name          Status      WLAN      Auth Protocol
-------------------------------------------------------------------------------------------
0014.d1da.a977    L2_1815w_1        Run 1 *      Yes        Ethernet
d8eb.97b6.fcc6    L2_1815w_1        Run 1 *      Yes        Ethernet
```

Use this command to display detailed information for a client by MAC address:

```
Device# show wireless client mac-address 2cea.7f18.5bb3 detail
Client MAC Address : 2cea.7f18.5bb3
Client MAC Type : Universally Administered Address
Client DUID: NA
```

```
Client IPv4 Address : 10.56.33.21
Client IPv6 Addresses : fe80::d60:2e8:4cc2:6212
Client Username: N/A
AP MAC Address : 4ca6.4d22.1a80
AP Name: AP3C57.31C5.799C
AP slot : 16
Client State : Associated
Policy Profile : fabric-rlan-policy
Flex Profile : default-flex-profile
Remote LAN Id: 1 <----------
Remote LAN Name: fabric-rlan <--------
Wireless LAN Network Name (SSID): fabric-rlan <----------
BSSID : 4ca6.4d22.1a81
Connected For : 211 seconds
Protocol : Ethernet <--------
Channel : 0
Port ID: 1 <-----------
Client IIF-ID : 0xa0000002
Association Id : 0
Authentication Algorithm : Open System
<--------o/p trimmed ------>
```

Use this command to display the summary of all AP tags:

```
Device# show ap tag summary
Number of APs: 2

AP Name             AP Mac               Site Tag Name         Policy Tag Name
         RF Tag Name               Misconfigured     Tag Source
-------------------------------------------------------------------------------------------------
L2_1810d_1        0008.3296.24c0       default-site-tag
default-policy-tag        default-rf-tag             No               Default

L2_1810w_2        00b0.e18c.5880       rlan-site-tag              rlan_pt_1
           default-rf-tag             No               Static
```

Use this command to display the summary of all policy tags:

```
Device# show wireless tag policy summary
Number of Policy Tags: 2

Policy Tag Name                   Description
------------------------------------------------------------------------
rlan_pt_1
default-policy-tag                default policy-tag
```

Use this command to display details of a specific policy tag:

```
Device# show wireless tag policy detailed <rlan_policy_tag_name>
Policy Tag Name : rlan_pt_1
Description     :

Number of WLAN-POLICY maps: 0

Number of RLAN-POLICY maps: 2
REMOTE-LAN Profile Name           Policy Name                          Port
 Id
-------------------------------------------------------------------------------------------
rlan_test_1                       rlan_named_pp1
 1
```

```
rlan_test_1                      rlan_named_pp1
 2
```

Use this command to display the fabric client summary:

```
Device# show wireless fabric client summary

Number of Fabric Clients : 0

MAC Address    AP Name                          WLAN State              Protocol
 Method     L2 VNID    RLOC IP
```

Use this command to display the RLAN client summary:

```
Device# show wireless client summary

Number of Clients: 1

MAC Address        AP Name       Type   ID  State    Protocol   Method   Role
-------------------------------------------------------------------------------
2cea.7f18.5bb3 AP3C57.31C5.799C  RLAN    1    Run    Ethernet   None    Local

Number of Excluded Clients: 0
```

RLAN authentication fallback

Introduces RLAN authentication fallback, explaining how client authentication dynamically switches between 802.1X and MAB on OEAP RLAN ports, details prerequisites and operational behavior, and highlights feature availability in Cisco IOS XE Cupertino 17.8.1 and later.

An RLAN authentication fallback is a WLAN authentication mechanism that

- enables client authentication to switch between 802.1X and MAC authentication bypass (MAB) when the initial method fails
- supports dynamic fallback from 802.1X to MAB and vice versa according to client status and registration, and
- requires that both 802.1X and MAB are enabled for fallback support.

Feature history

Table 187: Feature history for RLAN authentication fallback

Feature name	Release information	Feature description
RLAN authentication fallback	Cisco IOS XE 17.8.x	RLAN authentication fallback alternates client authentication between 802.1X and MAC authentication bypass (MAB) when the initial method fails. It dynamically switches based on client status and registration, requiring both 802.1X and MAB to be enabled. From Cisco IOS XE Cupertino 17.8.1, Remote LAN (RLAN) ports on OfficeExtend Access Points (OEAPs) support authentication fallback.

How RLAN authentication fallback works

From Cisco IOS XE Cupertino 17.8.1, Remote LAN (RLAN) ports on OfficeExtend Access Points (OEAPs) support authentication fallback. If a client using IEEE 802.1X fails to authenticate, the system attempts MAC authentication bypass (MAB). Conversely, if the client's MAC address is not registered for MAC authentication bypass, the system falls back to IEEE 802.1X. Enable both methods to ensure successful authentication.

By default, the RLAN fallback mechanism is disabled. You must explicitly enable this mechanism. When both 802.1X and MAB are enabled, the device must succeed in both authentication methods for successful authentication.

Configure RLAN authentication fallback (CLI)

Configure RLAN authentication fallback using commands.

Enable authentication fallback for remote LAN (RLAN) profiles on your device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a remote LAN profile.

   Example

   ```
   Device(config)# ap remote-lan profile-name rlan_profile_name rlan-id
   ```

3. Enable 802.1X authentication on MAC filter failure.

   Example

   ```
   Device(config-remote-lan)# security {dot1x on-macfilter-failure | mac-filter
    on-dot1x-failure}
   ```

   Note

   You can either configure 802.1X authentication on MAC filter failure or MAC filter authentication on 802.1X failure. You cannot configure both.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-remote-lan)# end
   ```

If the primary authentication method fails, the RLAN profile attempts an alternative authentication method."

```
Device# configure terminal
Device(config)# ap remote-lan profile-name rlan_profile_name 3
Device(config-remote-lan)# security dot1x on-macfilter-failure
Device(config-remote-lan)# end
```

Modify 802.1X EAP timers for RLAN clients

Modify 802.1X EAP timers for RLAN clients.

Adapt 802.1X EAP authentication timers for remote LAN (RLAN) clients to ensure successful endpoint authentication.

To adapt the 802.1X EAP timers for RLAN clients, use this procedure:

> **Note**
>
> When you modify the 802.1X EAP timers, ensure that the timer is long enough to allow 802.1X-capable endpoints to authenticate. A timer that is too short may result in 802.1X-capable endpoints being subject to a fallback authentication or authorization technique.
>
> If 802.1X EAP timers are not configured using this procedure, the timer configuration done using the **wireless security dot1x request** and **wireless security dot1x identity-request** commands are applied.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the remote LAN profile.

   Example

   ```
   Device(config)# ap remote-lan profile-name rlan_profile_name rlan-id
   ```

3. Configure the maximum number of EAP ID request retransmissions.

   Example

   ```
   Device(config-remote-lan)# security dot1x identity-request retries retry-num
   ```

   The valid range is from 1 to 20.

4. Configure the EAP ID request-timeout value in seconds.

   Example

   ```
   Device(config-remote-lan)# security dot1x identity-request timeout
   timeout-value
   ```

The valid range is from 1 to 120.

5. Configure the maximum number of EAP request retransmissions.

   Example

   ```
   Device(config-remote-lan)# security dot1x request retries retry-num
   ```

   The valid range is from 0 to 20.

6. Configure the EAP request retransmission timeout value in seconds.

   Example

   ```
   Device(config-remote-lan)# security dot1x request timeout timeout-value
   ```

   The valid range is from 1 to 120.

7. Return to privileged EXEC mode.

   Example

   ```
   Device(config-remote-lan)# end
   ```

The EAP authentication timers for RLAN clients are adapted to your specified values, improving authentication reliability for 802.1X-capable endpoints.

```
Device# configure terminal
Device(config)# ap remote-lan profile-name rlan_profile_name 3
Device(config-remote-lan)# security dot1x identity-request retries 20
Device(config-remote-lan)# security dot1x identity-request timeout 120
Device(config-remote-lan)# security dot1x request retries 20
Device(config-remote-lan)# security dot1x request timeout 120
Device(config-remote-lan)# end
```

Verify RLAN authentication fallback

Verify RLAN authentication fallback.

Use this command to check the status of the fallback authentication mechanism.

```
Device# show remote-lan all

Remote-LAN Profile Name     : rlan_profile_name
================================================
Identifier                                    : 3
Status                                        : Disabled
Mac-filtering                                 : Not Configured
Number of Active Clients                      : 0
Security_8021x_dot1x                          : Enabled
8021.x Authentication list name               : Not Configured
Local Auth eap Profile Name                   : Not Configured
Web Auth Security                             : Disabled
Webauth Authentication list name              : Not Configured
Web Auth Parameter Map                        : Not Configured
Client association limit                      : 0
```

```
Ipv4 Web Pre Auth Acl                          : Not Configured
Ipv6 Web Pre Auth Acl                          : Not Configured
mDNS Gateway Status                            : Bridge
Authentication Fallback Status                 : MAC-filtering to Dot1X
```

205 RLAN External Module

Topics:

- External module

External module

Explains the External Module feature that enables traffic to flow in and out from the Cisco Aironet Developer Platform module.

The External Module feature is a traffic management capability that

- enables traffic to flow in and out from the Cisco Aironet Developer Platform module
- operates when an access point (AP) is in both local and flex connect mode, and
- provides connectivity for external platform modules through the AP infrastructure.

Requirement: Prerequisites for configuring external module

Outlines the prerequisites that must be met before configuring an external module.

Before you begin, you must ensure the following:

- The external module is powered on.
- The RLAN status is enabled.

Configure external module (GUI)

Configure external module using GUI to assign RLAN policy profiles and profiles to external module ports through policy tags.

This task allows you to configure external module settings through the graphical user interface by creating policy tag mappings that associate RLAN profiles and policy profiles with external module ports.

Before you begin

Follow these steps to configure external module using GUI:

Procedure

1. Choose **Configuration** > **Tags & Profiles > Tags**.
2. In the **Policy** tab, select one of the **Policy Tag Name** and click **Add**.
3. In **Add Policy Tag** page and **RLAN-POLICY Maps** section, click **Add**.
4. From the **Port ID** drop-down list, choose **ext-module**.
5. From the **RLAN Profile** drop-down list, choose an RLAN profile.
6. From the **RLAN Policy Profile** drop-down list, choose an RLAN policy profile.
7. Click the check mark icon.

The external module is configured with the selected RLAN profile and policy profile through the policy tag mapping.

Configure external module (CLI)

Configure external module using commands.

Configure external module settings to enable remote LAN functionality on wireless devices.

External modules provide additional capabilities for remote LAN connectivity in wireless networks. This configuration assigns policy tags and remote LAN policies to external modules.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a policy tag to the external module for the remote LAN.

   Example

   ```
   Device(config)# wireless tag policy default-policy-tag
   ```

3. Configure a remote LAN policy to the external module.

   Example

   ```
   Device(default-policy-tag)# remote-lan rlan-profile policy rlan-policy
   ext-module
   ```

   Example

   ```
   Device(default-policy-tag)# remote-lan rlan policy abc ext-module
   ```

The external module is now configured with the specified policy tag and remote LAN policy, enabling remote LAN functionality.

External module verification

Provides commands to verify external module configuration and inventory details.

Verify external module configuration and inventory details using show commands.

To view the external module remote LAN configuration, use the following command:

```
Device# show ap name ap_name lan port summary

LAN Port status for AP ap_name

Port ID      status      vlanId      poe        power-level     RLAN
-------------------------------------------------------------------
ext-module   Enabled     39          NA         NA              Enabled
```

To view the external module inventory details, use the following command:

```
Device# show ap name abc inventory
NAME: AP3800, DESCR: Cisco Aironet 3800 Series (IEEE 802.11ac) Access Point
PID: AIR-AP3802I-D-K9, VID: 01, SN: xxxxxxxxxxx

MODULE NAME: Expansion Module, DESCR: Cisco HDK Module (rev2)
```

```
PID: Unknown, SN: xxxxxxxxxxx, MaxPower: 2700mW
VersionID: V22, Capabilities: RLAN (UP)
```

206 802.11ax Per WLAN

Topics:

- 802.11ax mode per WLAN

802.11ax mode per WLAN

Explains how 802.11ax mode is configured per WLAN to provide more granular control over wireless network settings.

An 802.11ax mode per WLAN is a wireless configuration feature that

- enables or disables 802.11ax functionality at the individual WLAN level rather than per radio band
- provides granular control over 11ax settings for specific SSIDs
- allows mixed 11ax and legacy client connectivity on the same access point.

Configuration evolution and behavior

Prior to Cisco IOS XE Bengaluru Release 17.4.1, the 802.11ax mode was configured per radio band. In this configuration, the 11ax mode was either enabled or disabled for all WLANs (AP) that were configured per radio, all at once. When 11ax was enabled per radio, the 11ac clients were not able to scan or connect to the SSID if the beacon had 11ax information elements. Client could not probe an access point (AP), if the beacon has 11ax IE.

Therefore, a 11ax configuration knob per AP is introduced, from Cisco IOS XE Bengaluru Release 17.5.1. This knob is introduced under the WLAN profile. By default, the 11ax knob per WLAN is now enabled on the controller.

> **Note**
>
> For 6-GHz radio, the 802.11ax parameters are taken from the multi BSSID profile tagged to the corresponding 6-GHz RF profile of the AP. So, the WLAN dot11ax parameters are overridden by multi BSSID profile parameters in the case of 6-GHz. There are no changes for 2.4 and 5-GHz band WLANs. They continue to use the WLAN parameters for 802.11ax.

Configure 802.11ax mode per WLAN (GUI)

Configure 802.11ax mode per WLAN using GUI.

Enable or disable 802.11ax mode on a specific WLAN to control Wi-Fi 6 functionality.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.

   The **Add WLAN** window is displayed.
3. Click the **Advanced** tab.
4. In the **11ax** section, check the **Enable 11ax** check box to enable 802.11ax operation status on the WLAN.

> **Note**
>
> When 11ax is disabled, beacons will not display 11ax IE, and all the 11ax features will be operationally disabled on the WLAN.

5. Click **Apply to Device**.

The 802.11ax mode is now configured or disabled on the specified WLAN according to your requirements.

Configure 802.11ax mode per WLAN (CLI)

Configure 802.11ax mode per WLAN using commands.

Enable or disable 802.11ax mode on a specific WLAN to control Wi-Fi 6 functionality.

802.11ax (Wi-Fi 6) provides improved performance and efficiency. This configuration allows you to enable or disable 802.11ax mode on individual WLANs based on your network requirements.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the WLAN name and enter the WLAN configuration mode.

   Example

   ```
   Device(config)# wlan wlan-profile-name
   ```

3. Configure 802.11ax on a WLAN.

   Example

   ```
   Device(config-wlan)# dot11ax
   ```

4. Disable 802.11ax on the WLAN profile.

   Example

   ```
   Device(config-wlan)# no dot11ax
   ```

The 802.11ax mode is now configured or disabled on the specified WLAN according to your requirements.

Verify 802.11ax mode per WLAN

This topic explains how to verify 802.11ax mode per WLAN to confirm the operational status of 802.11ax parameters.

This topic verifies the operational status of 802.11ax parameters for a specific WLAN configuration to ensure the wireless network is operating with the expected 802.11ax features enabled.

Verify 802.11ax mode per WLAN

To display the status of the 11ax parameter, use this command:

```
Device# show wlan id 6
            WLAN Profile Name     : power
            ================================================
            Identifier                                    : 6
            Description                                   :
            Network Name (SSID)                           : power
            Status                                        : Enabled
            Broadcast SSID                                : Enabled
            Advertise-Apname                              : Disabled
            Universal AP Admin                            : Disabled
            Max Associated Clients per WLAN               : 0
            Max Associated Clients per AP per WLAN        : 0
            Max Associated Clients per AP Radio per WLAN  : 200
            .
            .
            .
            802.11ac MU-MIMO                              : Enabled
            802.11ax parameters
            802.11ax Operation Status                  : Enabled
            OFDMA Downlink                             : Enabled
            OFDMA Uplink                               : Enabled
            MU-MIMO Downlink                           : Enabled
            MU-MIMO Uplink                             : Enabled
            BSS Target Wake Up Time                    : Enabled
            BSS Target Wake Up Time Broadcast Support  : Enabled
            .
            .
            .
```

207 BSS Coloring

Topics:

- BSS coloring

BSS coloring

Describes a Wi-Fi 6 mechanism that allows devices to distinguish between packets from their own BSS and overlapping BSS by using color values in the PHY header.

BSS coloring is a Wi-Fi 6 mechanism that

- allows devices operating in the same frequency space to quickly distinguish between packets from their own BSS and packets from an Overlapping BSS (OBSS) by looking at the BSS color value in the HE PHY header
- enables spatial reuse opportunities where devices can transmit simultaneously with OBSS packets instead of deferring transmissions due to legacy interference thresholds, and
- increases power savings by allowing devices to drop non-BSS packets earlier without demodulating the entire packet.

BSS coloring background information

The 802.11 Wi-Fi standard minimizes the chance of multiple devices interfering with one another by transmitting at the same time. This carrier-sense multiple access with collision avoidance (CSMA/CA) technology is based on static thresholds that allow Wi-Fi devices to avoid interfering with each other on air. However, with an increase in density and the number of Wi-Fi devices, these static thresholds often lead to CSMA/CA causing devices to defer transmissions unnecessarily.

For example, if two devices that are associated with different BSS, can hear every transmission from each other at relatively low signal strengths, each device should defer its transmission when it receives a transmission from the other. But if both the devices were to transmit at the same time, it is likely that neither would cause enough interference at the other BSS' receiver to cause reception failure for either transmission.

Devices today must demodulate packets to look at the MAC header in order to determine whether or not a received packet belongs to their own BSS. This process of demodulation consumes power, which can be saved if devices can quickly identify the BSS by looking at the PHY header alone, and subsequently drop packets that are from a different BSS. Prior to Wi-Fi 6, there was no provision for devices to do this.

The new 802.11ax (Wi-Fi 6) standard addresses both of the issues discussed above, through the new BSS Coloring and Spatial Reuse mechanism. Since every Wi-Fi 6 device understands the BSS color, it can be leveraged to increase power savings by dropping packets earlier, and to identify spatial reuse opportunities.

BSS coloring

Explains a method used to differentiate between the BSS of access points and their clients on the same RF channel.

BSS coloring is a Wi-Fi 6 method that

- differentiates between the BSS of access points and their clients on the same RF channel
- enables each AP radio to assign a value (from 1 to 63), known as BSS color, to be included in the PHY header of all HE transmissions from devices in its BSS, and
- allows devices to quickly and easily distinguish transmissions coming from their BSS from those of neighboring BSS.

Supported platforms

The following platforms support this feature:

- Cisco Catalyst 9800 Series Wireless Controllers
- Cisco Catalyst 9115 Access Points
- Cisco Catalyst 9120AX Series Access Points
- Cisco Catalyst 9124AX Series Access Points

- Cisco Catalyst 9130AX Access Points

OBSS-PD and spatial reuse

Explains OBSS-PD spatial reuse technology that improves Wi-Fi 6 network efficiency by allowing controlled transmission over inter-BSS packets.

OBSS-PD is a Wi-Fi packet detection mechanism that

- uses a more aggressive packet detect threshold for inter-BSS packets, typically higher than the legacy -82 dBm
- identifies inter-BSS packets by comparing the BSS color in the HE PHY header with the device's BSS color, and
- enables Wi-Fi 6 or 802.11ax devices to transmit over inter-BSS packets with RSSI below the OBSS-PD threshold to improve throughput and network efficiency.

> **Note**
>
> Cisco Catalyst 9120AX Series Access Points do not support OBSS-PD.

Configure BSS color on AP (GUI)

Configure BSS color on AP using GUI.

Procedure

1. Choose **Configuration > Wireless > Access Points**.
2. Click the **5 GHz Radios** section or the **2.4 GHz Radios** section.

   The list of the AP radios in the band is displayed.
3. Click the required AP name.

   The **Edit Radios** window is displayed.
4. From the **Edit Radios** window, select the **Configure** tab.

   The general information, Antenna Parameters, RF Channel Assignment, Tx Power Level Assignment, and BSS Color are displayed.
5. In the **BSS Color** area and from the **BSS Color Configuration** drop-down list, choose **Custom** configuration
   - **Custom**: To manually select the BSS color configuration for the AP radio.
     a. Click the **BSS Color Status** field to disable or enable the feature.
     b. In the **Current BSS Color** field, specify a corresponding BSS color for the AP radio. The valid range is between 1 and 63.
6. Click **Update & Apply to Device**.

Configure BSS color in the privileged EXEC mode

Configure BSS color in the privileged EXEC mode.

Procedure

1. **enable**

   Example

   ```
   Device> enable
   ```

   Enables privileged EXEC mode. Enter your password, if prompted.

2. **ap name** *ap-name* **dot11 24ghz 5ghz 6ghz | dual-band slot** *slot-id* **dot11ax bss-color** *<1-63>*

   Example

   ```
   Device#ap name apn dot11 24ghz slot 0 dot11ax bss-color 12
   ```

   Sets the BSS color on the 2.4-GHz, 5-GHz, 6-GHz, or dual-band radio, for a specific access point on the following slots:

   - 5 GHz: Slot 1 and 2
   - 2.4 GHz: Slot 0
   - 6-GHz: Slot 3
   - Dual-band: Slot 0

   Use the **no** form of this command to disable BSS color.

   Example

   ```
   Device#ap name apn no dot11 24ghz slot 0 dot11ax bss-color
   ```

Configure BSS color globally

Configure BSS color globally using GUI.

Procedure

1. Choose **Configuration > Radio Configurations > Parameters**.
2. In the **11ax Parameters** section, enable BSS color globally for the 5 GHz and 2.4 GHz radios by checking the **BSS Color** check box.

Configure BSS color in the configuration mode

Configure BSS color in the configuration mode.

Enable 802.11ax BSS color functionality to improve spatial reuse and reduce interference in wireless networks.

BSS color is used in 802.11ax networks to distinguish between overlapping basic service sets, allowing devices to better manage interference and improve overall network performance.

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

   Enter your password if prompted.
2. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Enable the 802.11ax BSS color on all radios for the specified frequency band.

   Example

   ```
   Device(config)# [no] ap dot11 {24ghz | 5ghz | 6ghz} dot11ax bss-color
   ```

   Example

   ```
   Device(config)# ap dot11 24ghz dot11ax bss-color
   ```

   Use the `no` form of this command to disable BSS color.

BSS color is now enabled on the specified frequency band, improving spatial reuse and network efficiency for 802.11ax devices.

Configure overlapping BSS packet detect (GUI)

Configure overlapping BSS packet detect using GUI.

Procedure

1. Choose **Configuration** > **Radio Configurations** > **Parameters**.

   The parameters page is displayed where you can configure global parameters for 5 GHz Band and 2.4 GHz Band radios.
2. In the **11ax Parameters** section, check the **OBSS PD** check box to enable the overlapping BSS packet detect (OBSS PD) feature.
3. In the **Non-SRG OBSS PD Max Threshold** field, enter the threshold in decibel-milliwatts. Value range is between -82 dBm and -62 dBm.

Configure OBSS-PD spatial reuse globally (CLI)

Configure OBSS-PD spatial reuse globally using commands.

Enable 802.11ax OBSS-PD spatial reuse functionality to improve network efficiency by allowing concurrent transmissions when interference levels are acceptable.

OBSS-PD (Overlapping Basic Service Set Packet Detection) spatial reuse is an 802.11ax feature that improves network capacity by allowing access points to transmit simultaneously when the detected interference is below a specified threshold. Configure this feature globally across all radios of a specific frequency band.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure 802.11ax OBSS PD based spatial reuse on all 2.4 GHz or 5 GHz radios.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} dot11ax spatial-reuse obss-pd
   ```

   Example

   ```
   Device(config)# ap dot11 24ghz dot11ax spatial-reuse obss-pd
   ```

   Use the `no` form of this command to disable this feature.

3. Configure 802.11ax non-SRG OBSS PD max on all 2.4-GHz or 5-GHz radios.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz} dot11ax spatial-reuse obss-pd
   non-srg-max threshold-value
   ```

   Example

   ```
   Device(config)# ap dot11 24ghz dot11ax spatial-reuse obss-pd non-srg-max -62
   ```

   The default value is -62. The valid range is -82 to -62 dBm.

OBSS-PD spatial reuse is now configured globally on the specified frequency band, allowing access points to optimize concurrent transmissions based on the configured interference threshold.

Configure OBSS PD in an RF profile (GUI)

Configure OBSS PD in an RF profile using GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **RF**.
2. On the **RF Profile** page, click **Add** to configure the following:

- General
- 802.11
- RRM
- Advanced

3. In the **Advanced** tab, under the **11ax Parameters** section, complete the following:
   a) Use the toggle button to enable or disable the **OBSS PD** field.
   b) In the **Non-SRG OBSS PD Max Threshold (dBm)**, enter the threshold value. The default value is -62 dBm. Values range between -82 dBm and -62 dBm.
4. Click **Save & Apply to Device**.

Configure OBSS-PD spatial reuse in the RF profile mode (CLI)

Configure OBSS-PD spatial reuse in the RF profile mode using commands.

Enable OBSS-PD spatial reuse functionality on access points to improve network efficiency by allowing concurrent transmissions in overlapping basic service sets.

OBSS-PD spatial reuse is an 802.11ax feature that helps optimize spectrum utilization in dense wireless environments by dynamically adjusting transmission thresholds based on neighboring network activity.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an RF profile and enter RF profile configuration mode.

   Example

   ```
   Device(config)# ap dot11 {24ghz | 5ghz | 6ghz} rf-profile rf-profile-name
   ```

   Example

   ```
   Device(config)# ap dot11 24ghz rf-profile rfprof24_1
   ```

3. Configure 802.11ax OBSS PD based spatial reuse in the RF profile configuration mode.

   Example

   ```
   Device(config-rf-profile)# dot11ax spatial-reuse obss-pd
   ```

   Use the no form of this command to disable this feature.

4. Configure 802.11ax non-SRG OBSS PD maximum threshold.

Example

```
Device(config-rf-profile)# dot11ax spatial-reuse obss-pd non-srg-max
threshold-value
```

Example

```
Device(config-rf-profile)# dot11ax spatial-reuse obss-pd non-srg-max -62
```

The valid range is -82 to -62 dBm. The default value is -62.

OBSS-PD spatial reuse is configured on the RF profile and will be applied to all access points using this profile, enabling improved spectrum efficiency in overlapping coverage areas.

BSS color and OBSS-PD verification

Lists commands to verify BSS color and OBSS-PD configuration and status.

This reference provides commands to verify if the global per-band BSS color and OBSS-PD are enabled, view RF profile OBSS-PD configuration, check BSS color configuration of all AP radios on a band, and examine BSS color configuration and capability of specific AP radios.

To verify if the global per-band BSS color and OBSS-PD are enabled, use the following **show** command:

```
Device# show ap dot11 24ghz network
802.11b Network                              : Enabled
11gSupport                                   : Enabled
11nSupport                                   : Enabled
.
.
.
802.11ax                                     : Enabled
  DynamicFrag                                : Enabled
  MultiBssid                                 : Enabled
  Target Wakeup Time                         : Enabled
  Target Wakeup Time Broadcast               : Enabled
  BSS Color                                  : Enabled
  OBSS PD                                    : Enabled
  Non-SRG OBSS PD Max                        : -62 dBm
802.11ax MCS Settings:
  MCS  7, Spatial Streams = 1                : Supported
.
.
.
```

To view the RF profile OBSS-PD configuration, use the following **show** command:

```
Device# show ap rf-profile name rf-profile-name detail
Description                       : pre configured rfprofile for 5gh radio
RF Profile Name                   : rf-profile-name
Band                              : 5 GHz
Transmit Power Threshold v1       : -65 dBm
Min Transmit Power                : 7 dBm
Max Transmit Power                : 30 dBm
.
.
.
802.11ax
```

```
  OBSS PD                          : Enabled
  Non-SRG OBSS PD Max              : -62 dBm
NDP mode                           : Auto
```

To view the BSS color configuration of all the AP radios on a band in the summary list, along with Channel, TX Power and so on, use the following **show** command:

```
Device# show ap dot11 24ghz summary extended
AP Name                          Mac Address     Slot    Admin State    Oper
 State    Width  Txpwr           Channel                        BSS Color
-----------------------------------------------------------------------------------------------

Ed2-JFW-AP1                       84b2.61ba.4730  1       Enabled        Up
          40     1/6 (17 dBm)    (136,132)*

11AX-9120-AP1                     d4ad.bda2.3fc0  1       Enabled        Up
          20     1/8 (23 dBm)    (36)                           30
Ed2-JFW-AP2                       f8c2.8885.59f0  1       Enabled        Up
          20     1/5 (15 dBm)    (40)
```

To view the BSS color configuration and the capability of an AP radio, use the following **show** commands:

```
Device# show ap name AP7069.5A74.816C config dot11 24ghz
Cisco AP Identifier                             : 502f.a876.1e60
Cisco AP Name                                   : AP7069.5A74.816C
Attributes for Slot 0
  Radio Type                                    : 802.11b
  Radio Mode                                    : REAP
  Radio Role                                    : Auto
  Radio SubType                                 : Main
  Administrative State                          : Enabled
  Operation State                               : Up
  .
  .
  .
Phy OFDM Parameters
    Configuration                               : Automatic
    Current Channel                             : 6
    Channel Width                               : 20 MHz
    TI Threshold                                : 1157693440
    Antenna Type                                : External
    External Antenna Gain (in .5 dBi units)     : 8
    .
    .
    .
  !BSS color details are displayed below:
  802.11ax Parameters
    HE Capable                                  : Yes
    BSS Color Capable                           : Yes
    BSS Color Configuration                     : Customized
    Current BSS Color                           : 34
```

```
Device# show ap name AP70XX.5XX4.8XXX config slot 0
Cisco AP Identifier                             : 502f.a876.1e60
Cisco AP Name                                   : AP70XX.5XX4.8XXX
Country Code                                    : US
AP Country Code                                 : US  - United States
AP Regulatory Domain                            : -A
MAC Address                                     : 7069.5a74.816c
IP Address Configuration                        : DHCP
IP Address                                      : Disabled
```

```
.
.
.
Attributes for Slot 0
  Radio Type                                    : 802.11n - 2.4 GHz
  Radio Role                                    : Auto
  Radio Mode                                    : REAP
  Radio SubType                                 : Main
  Administrative State                          : Enabled
.
.
.
  Phy OFDM Parameters
    Configuration                               : Automatic
    Current Channel                             : 6
    Channel Assigned By                         : DCA
    Extension Channel                           : NONE
    Channel Width                               : 20
    Allowed Channel List                        : 1,2,3,4,5,6,7,8,9,10,11
    TI Threshold                                : 1157693440
    DCA Channel List                            :
    Antenna Type                                : EXTERNAL_ANTENNA
    External Antenna Gain (in .5 dBi units)     : 8
Diversity                                       : DIVERSITY_ENABLED
    802.11n Antennas
      A                                         : ENABLED
      B                                         : ENABLED
      C                                         : ENABLED
      D                                         : ENABLED
  .
  .
  .
  !BSS color details are displayed below:
  802.11ax Parameters
    HE Capable                                  : Yes
    BSS Color Capable                           : Yes
    BSS Color Configuration                     : Customized
    Current BSS Color                           : 34
.
.
.
```

208 DHCP for WLANs

Topics:

- Dynamic Host Configuration Protocol
- Configure DHCP for WLANs
- Configure internal DHCP server
- Configure DHCP-Required for FlexConnect

Dynamic Host Configuration Protocol

Describes a network protocol that dynamically assigns IP addresses to devices on a network for IP communication.

Dynamic Host Configuration Protocol (DHCP) is a network protocol that

- dynamically assigns IP addresses to devices on a network for IP communication
- enables configuration of WLANs to use the same or different DHCP servers or no DHCP server, and
- provides two types of DHCP servers: internal and external.

Internal DHCP servers

Describes internal DHCP servers that are typically used in branch offices without a dedicated DHCP server.

An internal DHCP server is a device-based DHCP service that

- provides DHCP addresses to wireless clients, direct-connect APs, and DHCP requests that are relayed from APs
- supports only lightweight APs, and
- requires SVI configuration for the client VLAN with the IP address set as DHCP server IP address.

General guidelines

The device contains an internal DHCP server. This server is typically used in branch offices that do not have a DHCP server.

DHCP option 43 is not supported on the internal server. Therefore, the APs must use an alternative method to locate the management interface IP address of the device, such as local subnet broadcast, Domain Name System (DNS), or priming.

When clients use the internal DHCP server of the device, IP addresses are not preserved across reboots. As a result, multiple clients can be assigned to the same IP address. To resolve any IP address conflicts, clients must release their existing IP address and request a new one.

Key configuration requirements:

- Internal DHCP server serves both wireless client and wired client (wired client includes AP).
- To serve wireless client with internal DHCP server, an unicast DHCP server IP address must be configured for wireless client. Internal DHCP server IP address must be configured under the server facing interface, which can be loopback interface, SVI interface, or L3 physical interface.
- To use internal DHCP server for both wireless and wired client VLAN, an IP address must be configured under client VLAN SVI interface.
- For wireless client, in DHCP helper address configuration, the IP address of the internal DHCP server must be different from address of wireless client VLAN SVI interface.
- For wireless client with internal DHCP server support, the internal DHCP server can be configured using global configuration command, under the client VLAN SVI interface or under the wireless policy profile.
- An internal DHCP server pool can also serve clients of other controllers .

Note

- VRF is supported in the internal DHCP servers.
- DHCPv6 is not supported in the internal DHCP servers.

External DHCP servers

Explains external DHCP servers that dynamically assign IP addresses within a network from a separate server outside the device.

An external DHCP server is a separate server outside the device that

- dynamically assigns IP addresses within a network
- operates with industry-standard DHCP Relay support, and
- maintains client IP addresses during roaming scenarios.

External DHCP server operation

The operating system is designed to appear as a DHCP relay to the network and as a DHCP server to clients with industry-standard external DHCP servers that support DHCP Relay, which means that each controller appears as a DHCP relay agent to the DHCP server, and as a DHCP server in the virtual IP address to wireless clients.

Because the controller captures the client IP address that is obtained from a DHCP server, it maintains the same IP address for that client during intra controller, inter controller, and inter-subnet client roaming.

Note

External DHCP servers support DHCPv6.

DHCP assignments

Describes DHCP server configuration options on a per-interface or per-WLAN basis.

A DHCP assignment is a network configuration method that

- configures DHCP servers on a per-interface or per-WLAN basis,
- allows assignment of primary and secondary DHCP servers to individual interfaces, and
- enables DHCP server definition on WLANs to override interface-level DHCP server addresses.

DHCP assignment configuration

Security considerations

You can configure DHCP on a per-interface or per-WLAN basis. We recommend that you use the primary DHCP server address that is assigned to a particular interface.

You can assign DHCP servers for individual interfaces. You can configure the management interface, AP manager interface, and dynamic interface for a primary and secondary DHCP server, and configure the service-port interface to enable or disable DHCP servers. You can also define a DHCP server on a WLAN (in this case, the server overrides the DHCP server address on the interface assigned to the WLAN).

For enhanced security, we recommend that you ask all clients to obtain their IP addresses from a DHCP server. To enforce this requirement, you can configure all the WLANs with a DHCP Address. Assignment Required setting, which disallows client static IP addresses. If DHCP Address Assignment Required is selected, clients must obtain an IP address through DHCP. Any client with a static IP address is not allowed on the network. The controller monitors DHCP traffic because it acts as a DHCP proxy for the clients.

Note

- WLANs that support management over wireless must allow management (device-servicing) clients to obtain an IP address from a DHCP server.
- The operating system is designed to appear as a DHCP relay to the network and as a DHCP server to clients with industry-standard external DHCP servers that support DHCP relay. This means that each controller appears as a DHCP relay to the DHCP server and as a DHCP server at the virtual IP address to wireless clients.

You can create WLANs with DHCP Address Assignment Required disabled. If you do this, clients have the option of using a static IP address or obtaining an IP address from a designated DHCP server. However, note that this might compromise security.

Note

DHCP Address Assignment Required is not supported for wired guest LANs.

You can create separate WLANs with DHCP Address Assignment Required configured as disabled. This is applicable only if DHCP proxy is enabled for the controller. You must not define the primary or secondary configuration DHCP server instead you should disable the DHCP proxy. These WLANs drop all the DHCP requests and force clients to use a static IP address. These WLANs do not support management over wireless connections.

DHCP option 82

Describes additional security when DHCP is used to allocate network addresses.

DHCP option 82 is a DHCP relay agent feature that

- provides additional security when DHCP is used to allocate network addresses
- enables the controller to act as a DHCP relay agent to prevent DHCP client requests from untrusted sources, and
- allows the controller to add option 82 information to DHCP requests from clients before forwarding the requests to the DHCP server.

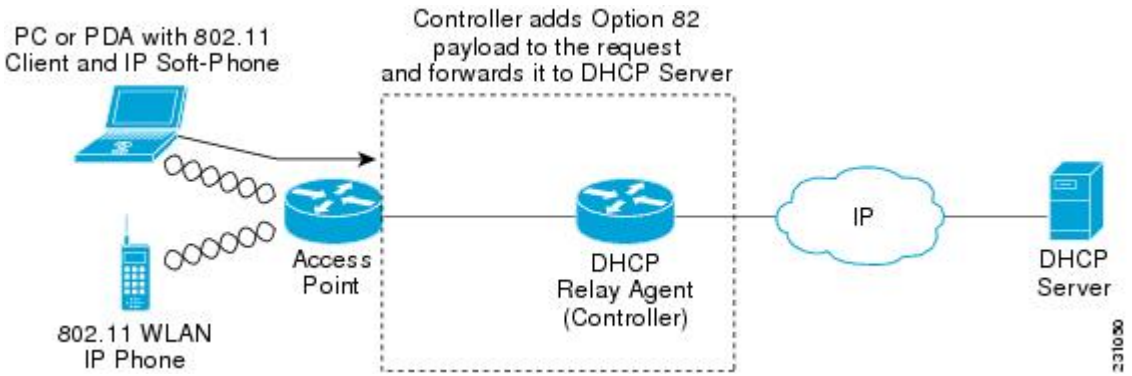
DHCP option 82 operation

The AP forwards all the DHCP requests from a client to the controller. The controller adds the DHCP option 82 payload and forwards the request to the DHCP server. The payload can contain the MAC address or the MAC address and SSID of the AP, depending on how you configure this option.

> **Note**
>
> DHCP packets that already include a relay agent option are dropped at the controller.

For DHCP option 82 to operate correctly, DHCP proxy must be enabled.

Figure 36: DHCP Option 82



Restrictions for configuring DHCP for WLANs

Outlines restrictions for configuring DHCP for WLANs.

When configuring DHCP for WLANs, ensure proper server reachability and service enablement to avoid connectivity issues.

- If you override the DHCP server in a WLAN, you must ensure that you configure the underlying Cisco IOS configuration to make sure that the DHCP server is reachable.
- WLAN DHCP override works only if DHCP service is enabled on the controller.

  You can configure DHCP service in either of the following ways:

  - Configuring the DHCP pool on the controller.
  - Configuring a DHCP relay agent on the SVI. Note that the VLAN of the SVI must be mapped to the WLAN where DHCP override is configured.

Guidelines for DHCP relay configuration

Outlines the guidelines for configuring DHCP relay agent source IP and server addresses to ensure proper functionality.

Relay agent source IP

Configure the relay agent source IP according to the following precedence order to ensure proper DHCP relay functionality.

- If you configure source interface VLAN in the SVI interface, the IP address of the VLAN interface configured as source is used.
- If the Relay Agent source IP is not mentioned, the IP address of the SVI interface created for the corresponding client's VLAN is used.
- If the Relay Agent source IP is not mentioned, the source address specified at the global level is used.

DHCP server

Configure DHCP server addresses according to the following precedence order to ensure proper server selection.

- If the DHCP server address is configured in the wireless policy profile, the server address configured in the policy profile takes precedence.
- If the DHCP server address is not configured in the policy profile, the server address configured in SVI takes precedence.

> **Note**
>
> You can configure two server addresses in the SVI. In this case, the DHCP packets from the client are sent to both the servers.
>
> The Option 82 configured in policy profile, SVI, and globally is considered and honored together.

These guidelines apply to central DHCP and local switching configurations. The DHCP packets are sourced from the IP address of the Wireless Management Interface (WMI), if VLAN is not configured in the policy profile and AAA override. The SVI interface configuration is mandatory to achieve the DHCP relay functionality in central DHCP or local switching. Even though many interface options are available in the **ip dhcp relay source-interface** <> command, only VLAN interface is applicable.

Configure DHCP for WLANs

Provides instructions for configuring DHCP scopes using GUI and CLI, including setting IP address ranges, subnet masks, lease options, and network parameters such as routers, DNS and NetBIOS name servers, domains, and DHCP options for wireless clients.

Configure DHCP scopes (GUI)

Configure DHCP scopes using the graphical user interface to manage IP address assignment and network configuration parameters for wireless clients.

Configure DHCP scopes to automatically assign IP addresses and network configuration parameters to wireless clients on your network.

DHCP scopes define the range of IP addresses available for assignment to clients, along with associated network configuration parameters such as subnet masks, default routers, and DNS servers.

Procedure

1. Choose **Administration > DHCP Pools**.
2. In the **Pools** section, click **Add** to add a new DHCP pool.

   The **Create DHCP Pool** dialog box is displayed.
3. In the **DHCP Pool Name** field, enter a name for the new DHCP pool.
4. From the **IP Type** drop-down list, choose the IP address type.
5. In the **Network** field, enter the network served by this DHCP scope.

   This IP address is used by the management interface with netmask applied, as configured in the **Interfaces** window.
6. In the **Subnet Mask** field, enter the subnet mask assigned to all the wireless clients.
7. In the **Starting IP** field, enter the starting IP address.

8. In the **Ending IP** field, enter the trailing IP address.
9. In the **Reserved Only** field, enable or disable it.
10. From the **Lease** drop-down list, choose the lease type as either **User Defined** or **Never Expires**.

    If you choose User Defined, you can enter the amount of time that an IP address is granted to a client.

11. To perform advanced configuration for DHCP scope, click **Advanced**.
12. Check the **Enable DNS Proxy** check box to enable DNS proxy.
13. In the **Default Router(s)** field, enter the IP address of the optional router or routers that connect to the device and click the **+** icon to add them to the list.

    Each router must include a DHCP forwarding agent that enables a single device to serve the clients of multiple devices.

14. In the **DNS Server(s)** field, enter the IP address of the optional DNS server or servers and click the **+** icon to add them to the list.

    Each DNS server must be able to update a client's DNS entry to match the IP address assigned by the DHCP scope.

15. In the **NetBios Name Server(s)** field, enter the IP address of the optional Microsoft NetBIOS name server or servers, such as Microsoft Windows Internet Naming Service (WINS) server, and click the **+** icon to add them to the list.
16. In the **Domain** field, enter the optional domain name of the DHCP scope for use with one or more DNS servers.
17. To add **DHCP** options, click **Add** in the **DHCP Options List** section.

    DHCP provides an internal framework for passing configuration parameters and other control information, such as DHCP options, to the clients on your network. DHCP options carry parameters as tagged data stored within protocol messages exchanged between the DHCP server and its clients.

18. Enter the **DHCP** option that you want to add.
19. Click **Save & Apply to Device**.

The DHCP scope is configured and ready to assign IP addresses and network configuration parameters to wireless clients.

Configure DHCP scopes (CLI)

Configure DHCP scopes using commands.

Set up DHCP scopes to automatically assign IP addresses and network configuration to client devices.

DHCP scopes define the range of IP addresses and network parameters that a DHCP server can assign to clients. This configuration is essential for automated network address management in enterprise environments.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the DHCP pool address.

   Example

   ```
   Device(config)# ip dhcp pool pool-name
   ```

Example

```
Device(config)# ip dhcp pool test-pool
```

3. Specify the network number in dotted-decimal notation and the mask address.

Example

```
Device(dhcp-config)# network network-name mask-address
```

Example

```
Device(dhcp-config)# network 209.165.200.224 255.255.255.0
```

4. Specify the DNS name server.

Example

```
Device(dhcp-config)# dns-server hostname
```

Example

```
Device(dhcp-config)# dns-server example.com
```

You can specify an IP address or a hostname.

5. Return to privileged EXEC mode.

Example

```
Device(dhcp-config)# end
```

The DHCP scope is now configured and ready to assign IP addresses, network masks, and DNS server information to DHCP clients.

Configure internal DHCP server

Provides instructions for configuring an internal DHCP server to automatically assign IP addresses to wireless clients. It includes steps for adjusting DHCP settings within a wireless policy profile using both the web interface and CLI commands. The configuration covers setting up centralized DHCP services, defining network pools, exclusions, and static IP reservations. Additionally, it outlines commands for verifying the DHCP configuration, including client binding checks and analyzing DHCP relay and packet punt statistics.

Configure the internal DHCP server under client VLAN SVI (GUI)

Configure the internal DHCP server under client VLAN SVI using GUI.

This task configures the internal DHCP server settings for a client VLAN Switched Virtual Interface (SVI) to enable DHCP relay functionality.

Use this procedure when you need to set up DHCP relay on an SVI to forward DHCP requests to a designated helper address. This configuration is typically performed on network switches to enable DHCP services across VLANs.

Procedure

1. Choose **Configuration** > **Layer2** > **VLAN** > **SVI**.
2. Click an SVI.
3. Click the **Advanced** tab.
4. Under **DHCP Relay** settings, enter the **IPV4 Helper Address**.
5. Click **Update & Apply to Device**.

The internal DHCP server is configured under the client VLAN SVI with the specified helper address, enabling DHCP relay functionality for the selected SVI.

Configure the internal DHCP server under client VLAN SVI (CLI)

Configure the internal DHCP server under client VLAN SVI using commands.

Set up an internal DHCP server to provide IP address assignment for wireless and wired clients through the client VLAN SVI configuration.

The internal DHCP server can be configured under the client VLAN SVI to serve both wireless and wired clients. This configuration requires proper IP addressing and helper configuration to function correctly with the wireless infrastructure.

Before you begin

- For wireless clients, only two DHCP servers are supported.
- To use the internal DHCP server for both wireless and wired client VLAN, an IP address must be configured under the client VLAN SVI.
- For wireless clients, the IP address of the internal DHCP server must be different from the address of the wireless client VLAN SVI (in the DHCP helper address configuration).
- For wireless clients, the internal DHCP server can be configured under the client VLAN SVI or under the wireless policy profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a loopback interface and enter interface configuration mode.

   Example

   ```
   Device(config)# interface loopback interface-number
   ```

   Example

   ```
   Device(config)# interface Loopback0
   ```

3. Configure the IP address for the interface.

Example

```
Device(config-if)# ip address ip-address subnet-mask
```

Example

```
Device(config-if)# ip address 10.10.10.1 255.255.255.255
```

4. Exit interface configuration mode.

Example

```
Device(config-if)# exit
```

5. Configure the VLAN ID.

Example

```
Device(config)# interface vlan vlan-id
```

Example

```
Device(config)# interface vlan 32
```

6. Configure the IP address for the interface.

Example

```
Device(config-if)# ip address ip-address subnet-mask
```

Example

```
Device(config-if)# ip address 192.168.32.100 255.255.255.0
```

7. Configure the destination address for UDP broadcasts.

Example

```
Device(config-if)# ip helper-address ip-address
```

Example

```
Device(config-if)# ip helper-address 10.10.10.1
```

> **Note**
>
> If the IP address used in the **IP helper-address** command is an internal address of the controller an internal DHCP server is used. Otherwise, the external DHCP server is used.

8. Disable the Maintenance Operation Protocol (MOP) for an interface.

   Example

   ```
   Device(config-if)# no mop enabled
   ```

9. Disable the task of sending MOP periodic system ID messages.

   Example

   ```
   Device(config-if)# no mop sysid
   ```

10. Exit interface configuration mode.

    Example

    ```
    Device(config-if)# exit
    ```

11. Specify the IP address that the DHCP server should not assign to DHCP clients.

    Example

    ```
    Device(config)# ip dhcp excluded-address ip-address
    ```

    Example

    ```
    Device(config)# ip dhcp excluded-address 192.168.32.1
    ```

12. Specify the IP addresses that the DHCP server should not assign to DHCP clients.

    Example

    ```
    Device(config)# ip dhcp excluded-address ip-address
    ```

    Example

    ```
    Device(config)# ip dhcp excluded-address 192.168.32.100
    ```

13. Configure the DHCP pool address.

    Example

    ```
    Device(config)# ip dhcp pool pool-name
    ```

Example

```
Device(config)# ip dhcp pool pool-vlan32
```

14. Specify the network number in dotted-decimal notation, along with the mask address.

Example

```
Device(dhcp-config)# network network-address subnet-mask
```

Example

```
Device(dhcp-config)# network 192.168.32.0 255.255.255.0
```

15. Specify the IP address of the default router for a DHCP client.

Example

```
Device(dhcp-config)# default-router ip-address
```

Example

```
Device(dhcp-config)# default-router 192.168.32.1
```

16. Exit DHCP configuration mode.

Example

```
Device(dhcp-config)# exit
```

17. Configure the WLAN policy profile and enter wireless policy configuration mode.

Example

```
Device(config)# wireless profile policy profile-policy
```

Example

```
Device(config)# wireless profile policy default-policy-profile
```

18. Configure central association for locally switched clients.

Example

```
Device(config-wireless-policy)# central association
```

19. Configure the central DHCP for locally switched clients.

Example

```
Device(config-wireless-policy)# central dhcp
```

20. Configure WLAN for central switching.

    Example

    ```
    Device(config-wireless-policy)# central switching
    ```

21. Add a description for the policy profile.

    Example

    ```
    Device(config-wireless-policy)# description "policy-profile-description"
    ```

    Example

    ```
    Device(config-wireless-policy)# description "default policy profile"
    ```

22. Assign the profile policy to the VLAN.

    Example

    ```
    Device(config-wireless-policy)# vlan vlan-id
    ```

    Example

    ```
    Device(config-wireless-policy)# vlan 32
    ```

23. Enable the wireless profile policy.

    Example

    ```
    Device(config-wireless-policy)# no shutdown
    ```

The internal DHCP server is now configured under the client VLAN SVI and can provide IP address assignment to both wireless and wired clients connected to the specified VLAN.

Configure the internal DHCP server under a wireless policy profile (GUI)

Configure the internal DHCP server under a wireless policy profile using the GUI.

Configure DHCP settings to enable automatic IP address assignment for wireless clients connected to the network under a specific policy profile.

Use this procedure when you need to set up internal DHCP server functionality within a wireless policy profile to provide IP address allocation for connected wireless devices.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click a policy name.
3. Click the **Advanced** tab.

4. Under **DHCP** settings, check or uncheck the **IPv4 DHCP Required** check box and enter the **DHCP Server IP Address**.
5. Click **Update & Apply to Device**.

The internal DHCP server is configured under the wireless policy profile and the settings are applied to the device.

Configure the internal DHCP server under a wireless policy profile (CLI)

Configure the Internal DHCP server under a wireless policy profile.

Enable DHCP services for wireless clients by configuring an internal DHCP server within a wireless policy profile.

The internal DHCP server provides IP address assignment and network configuration to wireless clients. This configuration includes setting up loopback and VLAN interfaces, defining DHCP pools, and associating the DHCP server with a wireless policy profile.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a loopback interface and enter interface configuration mode.

   Example

   ```
   Device(config)# interface loopback interface-number
   ```

   Example

   ```
   Device(config)# interface Loopback0
   ```

3. Configure the IP address for the interface.

   Example

   ```
   Device(config-if)# ip address ip-address subnet-mask
   ```

   Example

   ```
   Device(config-if)# ip address 10.10.10.1 255.255.255.255
   ```

4. Exit interface configuration mode.

   Example

   ```
   Device(config-if)# exit
   ```

5. Configure the VLAN ID.

Example

```
Device(config)# interface vlan vlan-id
```

Example

```
Device(config)# interface vlan 32
```

6. Configure the IP address for the interface.

Example

```
Device(config-if)# ip address ip-address subnet-mask
```

Example

```
Device(config-if)# ip address 192.168.32.100 255.255.255.0
```

7. Disable the Maintenance Operation Protocol (MOP) for an interface.

Example

```
Device(config-if)# no mop enabled
```

8. Disable the task of sending MOP periodic system ID messages.

Example

```
Device(config-if)# no mop sysid
```

9. Exit interface configuration mode.

Example

```
Device(config-if)# exit
```

10. Specify the IP address that the DHCP server should not assign to DHCP clients.

Example

```
Device(config)# ip dhcp excluded-address ip-address
```

Example

```
Device(config)# ip dhcp excluded-address 192.168.32.100
```

11. Configure the DHCP pool address.

Example

```
Device(config)# ip dhcp pool pool-name
```

Example

```
Device(config)# ip dhcp pool pool-vlan32
```

12. Specify the network number in dotted-decimal notation along with the mask address.

Example

```
Device(dhcp-config)# network network-address subnet-mask
```

Example

```
Device(dhcp-config)# network 192.168.32.0 255.255.255.0
```

13. Specify the IP address of the default router for a DHCP client.

Example

```
Device(dhcp-config)# default-router ip-address
```

Example

```
Device(dhcp-config)# default-router 192.168.32.1
```

14. Exit DHCP configuration mode.

Example

```
Device(dhcp-config)# exit
```

15. Configure a WLAN policy profile and enter wireless policy configuration mode.

Example

```
Device(config)# wireless profile policy profile-policy
```

Example

```
Device(config)# wireless profile policy default-policy-profile
```

16. Configure central association for locally switched clients.

Example

```
Device(config-wireless-policy)# central association
```

17. Configure local switching.

Example

```
Device(config-wireless-policy)# central switching
```

18. Add a description for the policy profile.

Example

```
Device(config-wireless-policy)# description "policy-profile-name"
```

Example

```
Device(config-wireless-policy)# description "default policy profile"
```

19. Enable DHCP Option 82 for the wireless clients.

Example

```
Device(config-wireless-policy)# ipv4 dhcp opt82
```

20. Enable ASCII on DHCP Option 82.

Example

```
Device(config-wireless-policy)# ipv4 dhcp opt82 ascii
```

21. Enable VLAN ID.

Example

```
Device(config-wireless-policy)# ipv4 dhcp opt82 format vlan-id
```

Example

```
Device(config-wireless-policy)# ipv4 dhcp opt82 format vlan32
```

22. Support the addition of Cisco 2-byte Remote ID (RID) for DHCP Option 82.

Example

```
Device(config-wireless-policy)# ipv4 dhcp opt82 rid
```

23. Configure the WLAN's IPv4 DHCP server.

Example

```
Device(config-wireless-policy)# ipv4 dhcp server ip-address
```

Example

```
Device(config-wireless-policy)# ipv4 dhcp server 10.10.10.1
```

24. Assign the profile policy to the VLAN.

Example

```
Device(config-wireless-policy)# vlan vlan-id
```

Example

```
Device(config-wireless-policy)# vlan 32
```

25. Enable the wireless profile policy.

Example

```
Device(config-wireless-policy)# no shutdown
```

The internal DHCP server is now configured and operational within the wireless policy profile, ready to provide IP addresses and network configuration to wireless clients.

Configure the internal DHCP server globally (GUI)

Configure the internal DHCP server globally using GUI.

Configure a DHCP pool to enable automatic IP address assignment for network devices through the internal DHCP server.

Use this procedure when you need to set up centralized IP address management for your network infrastructure using the GUI interface.

Procedure

---

1. Choose **Administration** > **DHCP Pools** > **Pools**.
2. Click **Add**.

   The **Create DHCP Pool** window is displayed.
3. Enter the **DHCP Pool Name**, **Network**, **Starting IP**, and **Ending IP**.
4. From the **IP Type**, **Subnet Mask**, and **Lease** drop-down lists, choose a value.
5. Click the **Reserved Only** toggle button.
6. Click **Apply to Device**.

The DHCP pool is created and applied to the device, enabling automatic IP address assignment within the specified range.

Configure the internal DHCP server globally (CLI)

Configure the internal DHCP server globally using commands.

Enable centralized DHCP services for wireless clients by configuring the internal DHCP server with appropriate network pools and policies.

The internal DHCP server configuration allows the wireless controller to provide IP addresses to wireless clients directly, eliminating the need for external DHCP servers in certain network deployments.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a loopback interface and enter interface configuration mode.

   Example

   ```
   Device(config)# interface loopback interface-number
   ```

   Example

   ```
   Device(config)# interface Loopback0
   ```

3. Configure the IP address for the interface.

   Example

   ```
   Device(config-if)# ip address ip-address subnet-mask
   ```

   Example

   ```
   Device(config-if)# ip address 10.10.10.1 255.255.255.255
   ```

4. Exit interface configuration mode.

   Example

   ```
   Device(config-if)# exit
   ```

5. Configure the VLAN ID.

   Example

   ```
   Device(config)# interface vlan vlan-id
   ```

   Example

   ```
   Device(config)# interface vlan 32
   ```

6. Configure the IP address for the interface.

   Example

   ```
   Device(config-if)# ip address ip-address subnet-mask
   ```

Example

```
Device(config-if)# ip address 192.168.32.100 255.255.255.0
```

7. Disable the Maintenance Operation Protocol (MOP) for an interface.

Example

```
Device(config-if)# no mop enabled
```

8. Disable the task of sending the MOP periodic system ID messages.

Example

```
Device(config-if)# no mop sysid
```

9. Exit the interface configuration mode.

Example

```
Device(config-if)# exit
```

10. Specify the target DHCP server parameters.

Example

```
Device(config)# ip dhcp-server ip-address
```

Example

```
Device(config)# ip dhcp-server 10.10.10.1
```

11. Specify the IP address that the DHCP server should not assign to DHCP clients.

Example

```
Device(config)# ip dhcp excluded-address ip-address
```

Example

```
Device(config)# ip dhcp excluded-address 192.168.32.100
```

12. Configure the DHCP pool address.

Example

```
Device(config)# ip dhcp pool pool-name
```

Example

```
Device(config)# ip dhcp pool pool-vlan32
```

13. Specify the network number in dotted-decimal notation along with the mask address.

Example

```
Device(dhcp-config)# network network-address subnet-mask
```

Example

```
Device(dhcp-config)# network 192.168.32.0 255.255.255.0
```

14. Specify the IP address of the default router for a DHCP client.

Example

```
Device(dhcp-config)# default-router ip-address
```

Example

```
Device(dhcp-config)# default-router 192.168.32.1
```

15. Exit DHCP configuration mode.

Example

```
Device(dhcp-config)# exit
```

16. Configure a WLAN policy profile and enter wireless policy configuration mode.

Example

```
Device(config)# wireless profile policy profile-policy
```

Example

```
Device(config)# wireless profile policy default-policy-profile
```

17. Configure central association for locally switched clients.

Example

```
Device(config-wireless-policy)# central association
```

18. Configure central DHCP for locally switched clients.

Example

```
Device(config-wireless-policy)# central dhcp
```

19. Configure local switching.

Example

```
Device(config-wireless-policy)# central switching
```

20. Add a description for the policy profile.

Example

```
Device(config-wireless-policy)# description policy-profile-description
```

Example

```
Device(config-wireless-policy)# description "default policy profile"
```

21. Assign the profile policy to the VLAN.

Example

```
Device(config-wireless-policy)# vlan vlan-id
```

Example

```
Device(config-wireless-policy)# vlan 32
```

22. Enable the profile policy.

Example

```
Device(config-wireless-policy)# no shutdown
```

The internal DHCP server is now configured globally with the specified pool, exclusions, and wireless policy profile settings to serve DHCP requests from wireless clients.

Configure IP reservations in the internal DHCP server (CLI)

Configure IP reservations in the internal DHCP server using CLI steps.

Reserve specific IP addresses for clients in the internal DHCP server to ensure consistent address assignment.

IP reservations allow DHCP servers to assign the same IP address to specific clients based on their client ID or MAC address. This is useful for servers, printers, or other devices that require consistent network addressing.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the DHCP pool address.

Example

```
Device(config)# ip dhcp pool pool-name
```

Example

```
Device(config)# ip dhcp pool dhcp-pool-add
```

3. Specify the network number in dotted-decimal notation along with the mask address.

Example

```
Device(dhcp-config)# network network-address mask-address
```

Example

```
Device(dhcp-config)# network 192.168.32.0 255.255.255.0
```

4. Configure a reserved address using either the client ID or the MAC address.

Example

```
Device(dhcp-config)# address ip-address client-id client-id
```

Example

```
Device(dhcp-config)# address ip-address hardware-address client-mac-id
```

Example

```
Device(dhcp-config)# address 209.165.200.224 client-id dhcp-client-id
```

> **Note**
>
> The IP reservation is contingent on the type of identifier, be it the client ID or the MAC address that the client provides. If the client sends its DHCP DISCOVER or REQUEST using the client ID, set your IP reservation using the client ID and not the hardware address.

IP reservations are configured in the internal DHCP server, ensuring that specified clients receive consistent IP addresses based on their identifiers.

Verify internal DHCP configuration

Provides commands to verify internal DHCP configuration.

Verify internal DHCP configuration using specific show commands to check client binding, DHCP relay statistics for wireless clients, and DHCP packet punt statistics.

Client binding verification

To verify client binding, use this command:

```
Device# show ip dhcp binding

Bindings from all pools not associated with VRF:
IP address      Client-ID/              Lease expiration        Type
State      Interface
                Hardware address/
                User name
192.168.32.3    0130.b49e.491a.53       Mar 23 2018 06:42 PM    Automatic
Active     Loopback0
```

DHCP relay statistics verification

To verify the DHCP relay statistics for a wireless client, use this command:

```
Device# show wireless dhcp relay statistics

DHCP Relay Statistics
---------------------

DHCP Server IP :   10.10.10.1

Message               Count
-------------------------
DHCPDISCOVER      :  1
BOOTP FORWARD     :  137
BOOTP REPLY       :  0
DHCPOFFER         :  0
DHCPREQUEST       :  54
DHCPACK           :  0
DHCPNAK           :  0
DHCPDECLINE       :  0
DHCPRELEASE       :  0
DHCPINFORM        :  82

Tx/Rx Time :
------------
LastTxTime : 18:42:18
LastRxTime : 00:00:00

Drop Counter :
-------------
TxDropCount : 0
```

DHCP packet punt statistics verification

To verify the DHCP packet punt statistics in CPP, use this command:

```
Device# show platform hardware chassis active qfp feature wireless punt
statistics

CPP Wireless Punt stats:

                                App Tag     Packet Count
                                -------     ------------
         CAPWAP_PKT_TYPE_DOT11_PROBE_REQ            14442
```

```
            CAPWAP_PKT_TYPE_DOT11_MGMT              50
            CAPWAP_PKT_TYPE_DOT11_IAPP            9447
            CAPWAP_PKT_TYPE_DOT11_RFID               0
             CAPWAP_PKT_TYPE_DOT11_RRM               0
          CAPWAP_PKT_TYPE_DOT11_DOT1X                0
    CAPWAP_PKT_TYPE_CAPWAP_KEEPALIVE             2191
CAPWAP_PKT_TYPE_MOBILITY_KEEPALIVE                   0
      CAPWAP_PKT_TYPE_CAPWAP_CNTRL                7034
       CAPWAP_PKT_TYPE_CAPWAP_DATA                   0
   CAPWAP_PKT_TYPE_MOBILITY_CNTRL                    0
                  WLS_SMD_WEBAUTH                    0
                SISF_PKT_TYPE_ARP                5292
               SISF_PKT_TYPE_DHCP                 140
             SISF_PKT_TYPE_DHCP6                 1213
          SISF_PKT_TYPE_IPV6_ND                   350
        SISF_PKT_TYPE_DATA_GLEAN                   44
     SISF_PKT_TYPE_DATA_GLEAN_V6                   51
        SISF_PKT_TYPE_DHCP_RELAY                  122
  CAPWAP_PKT_TYPE_CAPWAP_RESERVED                   0
```

Configure DHCP-Required for FlexConnect

Details the FlexConnect DHCP-required feature, ensuring wireless clients in local switching mode obtain IP addresses through DHCP. Includes FlexConnect-specific restrictions and limitations. Configuration is possible through the GUI or commands, with specific commands available for verification.

FlexConnect DHCP-Required

Describes a FlexConnect feature that enforces DHCP IP address acquisition for wireless clients in local switching mode.

FlexConnect DHCP-Required is a policy profile feature that

- forces connected wireless clients to obtain IP addresses from DHCP
- creates IP-MAC bindings by tracking IP addresses learned by the AP or controller, and
- maintains client connectivity during roaming within the same L2 network without requiring DHCP renegotiation.

FlexConnect DHCP-Required operation

When the DHCP-Required knob on a policy profile is enabled, a connected wireless client must get the IP address from DHCP. When the client completes the DHCP process and acquires an IP address, this IP address is learnt by the controller and only then the client traffic is switched on to the network. The DHCP-Required feature is already supported in central switching.

In Cisco IOS XE Amsterdam 17.2.1, the feature is supported on FlexConnect local switching clients. Prior to Release 17.2.1, DHCP-Required was not enforced on FlexConnect local switching clients. The IP address learnt by the AP or the controller for the wireless client is tracked to create an IP-MAC binding. As part of this feature, when a FlexConnect local switching client roams from one AP to another, the client need not do the DHCP again in the same L2 network, because the controller tracks the IP address and pushes the binding to the newly roaming AP.

The FlexConnect DHCP-Required feature can be configured from open configuration models, CLI, and from the GUI. The CLI and GUI configurations are described in this chapter. For more information about the open configuration modes, see the https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/prog/configuration/172/b_172_programmability_cg.html.

Restrictions for FlexConnect DHCP-Required

Restrictions and limitations for FlexConnect DHCP-required.

- The DHCP-Required feature is applicable for IPv4 addresses only.
- The IP-MAC binding can be pushed to other APs only through the custom policy profile. IP-MAC binding is not available in the default policy. The mapping is propagated to all the APs in the same custom policy profile.
- The DHCP-Required feature works on IP-MAC binding basis and is not supported with third party workgroup bridge (WGB), where WGB wired client information is not shared to AP by the WGB.
- Cisco Wave 2 APs take 180 seconds to remove a client entry with static IP, when DHCP-required is enabled.

Caution: Enabling IP-MAC address binding in SDA can cause high CAPWAP control traffic and CPU spikes

If you enable the **ipv4 dhcp required** setting with client IP-MAC address binding in Software-Defined Access (SDA) environments, then:

- Expect high CAPWAP control traffic during client join or slow roaming events.
- Be aware that Wireless Network Controller Device (WNCD) CPU utilization may spike above 95%, which can cause client connectivity issues.

  .
- Monitor your system for spikes in client count and CPU utilization per WNCD, especially during roaming events, and be prepared to respond if performance issues occur.

Normally, WNCD CPU utilization should remain consistent between SDA and non-SDA networks.

These are the symptoms.

- WLC CPU utilization may spike above 90% temporarily during client roaming events.
- High CAPWAP control traffic is observed due to distribution of client IP-MAC mappings to all APs in the flex group of the currently connected AP.

Conditions affected include:

- Cisco Catalyst 9800-40 controller
- SDA enabled with **ipv4 dhcp required** configured
- 9120 or 9130 APs, approximately 250 APs per WNCD
- SDA-enabled SSID

Monitoring steps:

1. Monitor client count and CPU utilization per WNCD.
2. Observe CPU utilization spikes during roaming events. CPU usage should stabilize after roaming ends.

Configure FlexConnect DHCP-Required (GUI)

Configure FlexConnect DHCP-required using GUI.

This task enables the FlexConnect DHCP-Required feature through the GUI interface to ensure proper DHCP configuration for FlexConnect access points.

Use this procedure when you need to configure the FlexConnect DHCP-Required feature through the GUI interface for policy profiles.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.

2. On the **Policy** window, click the name of the corresponding Policy Profile.
   The **Edit Policy Profile** window is displayed.
3. Click the **Advanced** tab.
4. In the **DHCP** section, check the **IPv4 DHCP Required** check box to enable the feature.
5. Click **Update & Apply to Device**.

The FlexConnect DHCP-Required feature is now enabled for the selected policy profile, and the configuration is applied to the associated devices.

Configure FlexConnect DHCP-required (CLI)

Configure FlexConnect DHCP-required using commands.

Enable the FlexConnect DHCP-required feature to ensure clients obtain IP addresses through DHCP before being allowed network access.

FlexConnect DHCP-required forces clients to obtain an IP address through DHCP before they can communicate on the network. This configuration is performed through the CLI on a WLAN policy profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy profile-policy
   ```

   Example

   ```
   Device(config)# wireless profile policy rr-xyz-policy-1
   ```

3. Enable the FlexConnect DHCP-Required feature.

   Example

   ```
   Device(config-wireless-policy)# ipv4 dhcp required
   ```

4. Save the configuration.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

FlexConnect DHCP-required is now configured on the policy profile, requiring clients to obtain IP addresses through DHCP before network access is granted.

Verify FlexConnect DHCP-required

Lists commands to verify FlexConnect DHCP-required functionality.

Verify FlexConnect DHCP-required functionality using wireless client commands to confirm IP address learning and client states.

- To verify the IP address learnt for a client on an IP DHCP-Required policy-enabled WLAN, use the **show wireless client summary** command:

> **Note**
>
> The controller or AP does not learn the IP address through other means such as ARP or data gleaning, when IPv4 DHCP-Required is enabled.

```
Device# show wireless client summary
Number of Clients: 1
MAC Address          AP Name            Type   ID  State          Protocol
Method     Role
-------------------------------------------------------------------------------------------
1cXX.bXXX.59XX       APXXXX.7XXX.4XXX   WLAN   3   IP Learn       11ac
Dot1x      Local
```

- This example shows that the client IP is in the **Run** state, indicating that the client has received the IP address from DHCP:

```
Device# show wireless client summary
Number of Clients: 1
MAC Address        AP Name              Type        ID        State
Protocol       Method      Role
-------------------------------------------------------------------------------------------
5XXX.37XX.c3XX     APXXXX.4XXX.4XXX     WLAN        3         Run          11n(5)
          None         Local
```

209 Aironet Extensions IE (CCX IE)

Topics:

- Cisco Aironet Extensions Information Element

Use this procedure to enable Aironet extensions IE for a WLAN using the graphical user interface.

Cisco Aironet Extensions Information Element

Explains the Cisco Aironet Extensions Information Element (IE), an attribute used by Cisco devices for better connectivity.

The Cisco Aironet Extensions Information Element (IE) is an attribute that

- contains information such as the AP name, device type, radio type, AP load, and the number of associated clients in the beacon and probe responses of the WLAN
- is used by Cisco devices for better connectivity, and
- provides information that Cisco Client Extensions use to associate with the best AP.

Configuration characteristics

The Aironet Extensions IE configuration has the following characteristics:

- Configuration is disabled by default
- Allows setting the AP name without enabling the whole IE extension by just inserting the AP name

> **Note**
>
> WLAN shutdown is required only when configuring Aironet Extensions (CCX IE) through the CLI. If CCX IE is enabled through the WebUI, WLAN shutdown is not necessary.

Configure Aironet Extensions IE using GUI

Configure Aironet extensions IE using GUI.

Enable Aironet extensions IE to enhance WLAN interoperability and support Aironet-specific features.

Before you begin

Follow these steps to configure Aironet Extensions IE using GUI:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs** .
2. In the **WLANs** window, click **Add**.
3. In the **Add WLAN** window, under the **Advanced** tab, check the **Aironet IE** check box to enable Aironet IE on the WLAN.
4. Click **Apply to Device**.

The WLAN is updated to include Aironet extensions IE, enabling enhanced interoperability.

Configure Aironet Extensions IE (CLI)

Configure Aironet extensions IE using commands.

Enable the Aironet Extensions IE feature on a WLAN to support enhanced Cisco client capabilities.

Perform this procedure to create a WLAN and enable the Aironet Extensions IE feature on the WLAN. For more information about the open configuration models, refer to the Programmability Configuration Guide, Cisco IOS XE Amsterdam 17.1.x.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Specify the WLAN name and ID.

   Example

   ```
   Device(config)# wlan profile-name wlan-id ssid
   ```

   Example

   ```
   Device(config)# wlan mywlan 34 mywlan-ssid
   ```

   - *profile-name*: Profile name. The range is from 1 to 32 alphanumeric characters.
   - *wlan-id*: WLAN ID. The range is from 1 to 512.
   - *ssid*: Service Set Identifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID.

   Note

   By default, the WLAN is disabled.

3. Configure the Cisco Client Extensions option and set the support of Aironet IE on the WLAN.

   Example

   ```
   Device(config-wlan)# ccx aironet-iesupport
   ```

   Use the **no** form of this command to disable the configuration.

The WLAN is created with Aironet Extensions IE enabled, allowing enhanced capabilities for Cisco wireless clients. You must now create a policy tag and map it to the AP to complete the configuration.

AP name addition verification

Provides verification of the AP name addition.

Verify the addition of AP name using Open Configuration in the beacon without enabling IE.

This example shows how to verify the addition of the AP Name (using Open Configuration) in the beacon without enabling IE:

```
Device# show wlan id 1
WLAN Profile Name     : wlan-test
================================================
Identifier                                    : 1
Description                                   :
Network Name (SSID)                           : wlan2
Status                                        : Disabled
Broadcast SSID                                : Enabled
Advertise-Apname                              : Enabled
Universal AP Admin                            : Disabled
Max Associated Clients per WLAN               : 0
Max Associated Clients per AP per WLAN        : 0
Max Associated Clients per AP Radio per WLAN  : 200
OKC                                           : Enabled
Number of Active Clients                      : 0
CHD per WLAN                                  : Enabled
WMM                                           : Allowed
Channel Scan Defer Priority:
  Priority (default)                          : 5
  Priority (default)                          : 6
Scan Defer Time (msecs)                       : 100
Media Stream Multicast-direct                 : Disabled
CCX - AironetIe Support                      : Enabled
Peer-to-Peer Blocking Action                  : Disabled
```

210 Device Analytics

Topics:

- Device analytics
- Adaptive 802.11r

Device analytics

Describes the Device Analytics feature that enhances the enterprise Wi-Fi experience for client devices to ensure seamless connectivity.

Device Analytics is a feature that

- enhances the enterprise Wi-Fi experience for client devices to ensure seamless connectivity
- provides a set of data analytics tools for analyzing wireless client device behavior, and
- encrypts exchanged data using AES-256-CBC to ensure device security when device profiling is enabled.

Device support information

The Device Analytics feature supports different device types with specific compatibility requirements:

Starting from Cisco IOS XE Bengaluru 17.6.1, this feature is supported on Intel devices with AC9560, AC8561, AX201, AX200, AX1650, AX210, AX211, and AX1675 chipsets. Device information and other information received from the Intel devices are shared with Cisco Catalyst Center. It will also be used to enhance device profiling on the controller.

> **Note**
>
> From Cisco IOS XE Dublin 17.12.1, MacBook Analytics is supported on the controller when the MacBook device sends 11k action frames along with the model information.

> **Note**
>
> Apple clients such as iPhones and iPads use 802.11k action frames to send device information to the controller. When they fail to send 802.11k action frames, the controller will not perform device classification based on the 802.11 protocol. Hence, this falls back to legacy device classification which is based on HTTP and DHCP protocols.

Restrictions for device analytics

Outlines the restrictions and requirements for implementing device analytics functionality.

Observe these restrictions when implementing device analytics:

- This feature is applicable only for Cisco device ecosystem partners.
- This feature is supported only on the 802.11ax and Wave 2 APs.
- This feature is supported using central authentication in either local mode or FlexConnect mode.
- To support Intel devices, AP should have PMF capability and PMF should set to optional or required on the WLAN.

Configure device analytics (GUI)

Configure device analytics using the GUI to enable wireless device monitoring and data collection.

Configure device analytics to monitor wireless device performance and collect analytical data for network optimization.

Device analytics provides insights into wireless device performance and network behavior. This configuration is performed at the WLAN level to control which wireless networks support analytics features.

Before you begin

Follow these steps to configure device analytics using the GUI:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. On the **WLANs** page, click the name of the WLAN.
3. In the **Edit WLAN** page, click the **Advanced** tab.
4. In the **Device Analytics** section, select the **Advertise Support** check box.
5. Select the **Advertise PC Analytics Support** check box to enable PC analytics on the WLAN.
6. Optional: In the **Device Analytics** section, select the **Share Data with Client** check box.
7. Click **Update & Apply to Device**.

Device analytics is configured for the WLAN and will begin collecting performance data from connected wireless devices.

Configure device analytics (CLI)

Configure device analytics using commands.

Enable device analytics on a WLAN to collect and optionally export device information for network analysis and optimization.

Device analytics allows the network to collect information from connected devices, which can be used for network monitoring and analysis. This feature can be enabled per WLAN and optionally configured to export information to compatible clients.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration sub-mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id SSID-name
   ```

   Example

   ```
   Device(config)# wlan device_analytics 1 device_analytics
   ```

   - *WLAN-name*: Enter the profile name. The range is from 1 to 32 alphanumeric characters.
   - *WLAN-ID*: Enter the WLAN ID. The range is from 1 to 512.
   - *SSID-name*: Enter the Service Set Identifier (SSID) for this WLAN. If the SSID is not specified, the WLAN profile name is set as the SSID.

> **Note**
>
> If you have already configured WLAN, enter the **wlan** *wlan-name* command.

3. Set the maximum number of clients, clients per AP, or clients per AP radio that can be configured on a WLAN.

   Example

   ```
   Device(config)# client association limit {clients-per-wlan | ap
   clients-per-ap-per-wlan | radio clients-per-ap-radio-per-wlan}
   ```

   Example

   ```
   Device(config)# client association limit 1 1
   ```

4. Enable or disable device analytics. WLANs advertise analytics capability in beacons & probe responses.

   Example

   ```
   Device(config)# [no] device-analytics
   ```

   This is enabled by default.

5. Configure device analytics export option.

   Example

   ```
   Device(config)# [no] device-analytics [export]
   ```

   Example

   ```
   Device(config)# device-analytics export
   ```

   When export option is set, the information from Cisco devices are shared with compatible clients (such as, Samsung devices). Here, information from Cisco devices refer to the Cisco controller details, AP version, and model number.

   This configuration is disabled by default.

6. Enable PC analytics on the WLAN.

   Example

   ```
   Device(config)# device-analytics pc-analytics
   ```

   Enables PC analytics on the WLAN. WLANs advertise analytics capability in beacons & probe responses.

7. Enable the WLAN.

   Example

   ```
   Device(config)# no shutdown
   ```

8. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

Device analytics is now configured on the WLAN, enabling the collection of device information and optionally exporting data to compatible clients.

Verify device analytics

Verify device analytics reports and client properties for wireless clients.

Verify device analytics to review client properties, statistics, and PC Analytics reports including neighbor AP information, candidate BSSIDs, and various diagnostic reports.

Use this procedure to access and review comprehensive analytics data for wireless clients, including connection quality metrics and troubleshooting information.

Procedure

1. On the **Monitoring > Wireless > Clients** page, click on a client in the table to view its properties and statistics.
2. In the **General** tab, click **Client Properties** to view the **PC Analytics** reports. This section displays the neighbor AP information, candidate BSSIDs, and reports for low RSSI, beacon miss, failed APs, and unknown APs.

The PC Analytics reports display comprehensive information about the selected client, including neighbor AP data, candidate BSSIDs, and diagnostic reports for connectivity issues such as low RSSI, beacon misses, failed APs, and unknown APs.

Verify device analytics configuration

Lists commands to verify device analytics configuration and view client device information.

This reference provides commands to verify device analytics configuration status, view client device information, check last disconnect reasons, and examine per-client pc-analytics reports.

Device analytics status verification

To view the status of device analytics export, use this command:

```
Device# show wlan 1 test-wlan

WLAN Profile Name     : test-wlan
================================================
Identifier                       : 1
Description                      :
Network Name (SSID)              : test-open-ssid
Status                           : Enabled
Broadcast SSID                   : Enabled
Advertise-Apname                 : Disabled
Universal AP Admin               : Disabled

Device Analytics
    Advertise Support            : Enabled
```

```
    Share Data with Client      : Disabled
```

Client device information

To view client device information, use this command:

```
Device# show device classifier mac-address 0040.96ae.xxx detail

  Client Mac: 0040.96ae.xxxx
  Device Type: Samsung Galaxy S10e(Phone)
  Confidence Level: 40
  Device Name: android-dhcp-9
  Software Version(Carrier Code): SD7(TMB)
  Device OS: Android 9
  Device Vendor: android-dhcp-9
  Country: US
```

Last disconnect reason

To view the last disconnect reason, use this command:

```
Device# show device classifier mac-address 0040.96ae.xxxx detail

Client MAC Address : 0040.96ae.xxxx
Client IPv4 Address : 12.1.0.52
Client IPv6 Addresses : fe80::631b:5b4f:f9b6:53cc
Client Username: N/A
AP MAC Address : 7069.5a51.53c0
AP Name: AP4C77.6D9E.61B2
AP slot : 1
Client State : Associated

Assisted Roaming Neighbor List
Nearby AP Statistics:
EoGRE : No/Simple client
Last Disconnect Reason : User initiated disconnection - Device was powered off
 or Wi-Fi turned off
```

Per-client pc-analytics reports

To view the per client pc-analytics reports, use this command:

```
Device# show wireless client mac-address 3413.e8b6.xxxx stats pc-analytics

-------------------------
Neighbor APs Info:
-------------------------
Reported time:: 06/21/2021 18:50:34
-------------------------
Roaming Reasons:
-------------------------
Selected AP RSSI:: -67
Candidate BSSIDs:
-----------------
Neighbor AP RSSI(dB)
a4b2.3903.d10e -70
-------------------------
```

```
PC Analytics report stats
-------------------------
------------------------------------------------------------------------
Report Type Processed Reports Dropped Reports
------------------------------------------------------------------------

STA Info 1 0
Neigh AP 1 0
Low RSSI 0 0
Beacon Miss 0 0
Failed AP 0 0
Unknown APs 0 0
```

Adaptive 802.11r

Describes the adaptive 802.11r functionality that enables device ecosystem partnerships and supports 802.11r-compatible devices on Cisco networks.

Adaptive 802.11r is a WLAN feature that

- enables device ecosystem partnerships for 802.11r functionality
- allows information sharing between client devices and the controller and AP, and
- is enabled by default when creating a WLAN.

Adaptive 802.11r functionality

The Cisco device ecosystem partner now supports 11r functionality on an adaptive 802.11r SSID. Samsung is one of the partners.

> **Note**
>
> The Adaptive 802.11r is enabled by default. This means that when you create a WLAN, the adaptive 802.11r is configured by default.

Client device information such as its model number, supported operating system is shared with the controller and AP while the device receives information such as controller and AP type, software release, etc. Also, this enables 802.11r-compatible devices to benefit from adaptive 802.11r on Cisco networks. This ecosystem comes handy especially for troubleshooting device disconnection from the AP as the controller receives information such as the disconnect reason code from the client device.

> **Note**
>
> Devices without 11r support cannot join an SSID where 11r is enabled.
>
> To use the 11r functionality on devices, you need to create a separate SSID with 11r enabled and another with 11r disabled to support the non-11r devices in the network.
>
> Adaptive dot11r is supported by Apple iPad, Apple iPhone, and Samsung S10 devices. However; some software update creates a MIC mismatch error in these devices. But these errors are transient and clients will successfully be able to associate to the SSID in subsequent results.

Configure adaptive 802.11r (GUI)

Configure adaptive 802.11r using GUI.

Configure adaptive 802.11r to enable fast roaming capabilities for wireless clients while maintaining backward compatibility with devices that do not support 802.11r.

Adaptive 802.11r allows wireless networks to automatically detect client capabilities and enable fast transition features when supported, providing seamless roaming without requiring separate WLAN configurations.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. On the **WLANs** page, click the name of the WLAN.
3. In the **Edit WLAN** window, click the **Security** > **Layer2** tab.
4. In the **WPA Parameters** section and **Fast Transition** drop-down list, choose **Adaptive Enabled**.
5. Click **Update & Apply to Device**.

The WLAN is configured with adaptive 802.11r, enabling automatic fast transition capabilities for compatible clients while maintaining connectivity for non-802.11r devices.

Adaptive 802.11r verification

Provides the command to verify adaptive 802.11r configuration.

Verify adaptive 802.11r configuration using command line interface.

To view the details, use this command:

```
Device# show running-config all
wlan test-psk 2 test-psk
 security ft adaptive
"adaptive" is  optional
```

Note

This command is used to enable or disable adaptive 11r:

[no] security ft adaptive

This command is used to enable or disable 802.11r:

[no] security ft

211 Device Classifier Dynamic XML Support

Topics:

- Device classifier dynamic XML support

Introduces the dynamic XML support feature for device classifier functionality and explains how it enables flexible device identification and classification.

Device classifier dynamic XML support

Explains device classifier dynamic XML support that enables better device classification without upgrading the device to a new release.

Device classifier dynamic XML support is a feature that

- enables better device classification without upgrading the device to a new release
- allows updating XML files with new rules, checks, and profiles based on connected devices, and
- provides dynamic device classification by copying XML files to the device file system.

Device classification workflow and configuration

The current device classifier uses static XML file wherein you define checks, rules, and profiles based on MAC, DHCP, and HTTP TLVs in wireless devices. The static XML file is converted to a text file and integrated with the image. When you enable the device classified functionality using **device classifier** command, the contents in the text file is read and populated into the device classifier structures.

> **Note**
>
> The subsequent device classification is based on the populated device classifier structures.

Presently, if you find any unclassified devices in a controller, the static XML file is updated with the new rules, checks, and profiles to get the devices classified. You will need to wait till the subsequent release as the static XML file is integrated with the image and cannot be changed from the controller.

The device classifier dynamic XML support enhancement addresses this problem for device classifier dynamic XML file.

With the introduction of dynamic XML support, you are provided with a new dynamic device classifier XML file support.

> **Note**
>
> The filename will be **dc_user_profiles.XML** and you can update the dynamic XML file with the new rules, checks, and profiles based on the devices connected and according to the provided schema.You can then copy this XML file to the device file system to enable better device classification without the need to upgrade the device to a new release.

The static XML file support is still available. If a device is connected, its TLVs are checked with the dynamic XML user profiles first and if it matches it is classified as per that profile. If you search for non-static XML profiles and if it matches it is classified as per that profile.

Workflow to classify unclassified devices with dynamic XML file:

1. The dynamic XML filename must be **dc_user_profiles.XML**.

> **Note**
>
> Files with any other name are not read and parsed even if they have the correct schema.

2. Copy the sample dynamic XML file to your system using this command:

   copy {flash:} {ftp: | tftp:}

3. Provide a new version for the dynamic XML file:

   <Version>1.1</Version>

4. Edit the dynamic XML file with the new rules, checks, and profiles as defined in the schema after examining the TLV values of the unclassified devices.

   > **Note**
   >
   > To check the TLV values, use this command:
   >
   > **show wireless client MAC-address** *MAC* **detail**

5. Copy the dynamic XML file to the device flash using this command:

   copy {ftp: | tftp:} {flash:}

Once the file is copied to the device file system, the newly connected clients are classified according to the new profiles defined in the dynamic XML file. You need to reconnect the already unclassified devices to send the DHCP and HTTP TLVs, and then classify them according to the new profiles. The already classified devices remain as classified until they are reconnected.

Note

The sample dynamic XML file is available in the device at *flash:dc_profile_dir/*. You can consider these sample dynamic XML file schema and copy this to your system using **copy {flash:} {ftp: | tftp:}** command, and append or replace the content with your own profiles, rules, and checks:

```
<?xml version="1.0" encoding="UTF-8"?>
<DeviceList>
             <CopyRight>Copyright (c) 2021-2022 by Cisco Systems, Inc. All
rights reserved.</CopyRight>
             <Version>1.0</Version>
             <Device>
                   <DeviceType>Sample_Profile_1</DeviceType>
                   <RuleName>Sample_Rule_1</RuleName>
                   <RuleOperator>OR</RuleOperator>
                   <RuleCertaintyMetric>20</RuleCertaintyMetric>
                   <Check>
                              <Protocol>DHCP</Protocol>
                              <TLV-Type>12</TLV-Type>
                              <TLV-Value-Type>String</TLV-Value-Type>
                              <TLV-Value>test</TLV-Value>
                   </Check>
                   <Check>
                              <Protocol>HTTP</Protocol>
                              <TLV-Type>3</TLV-Type>
                              <TLV-Value-Type>Integer</TLV-Value-Type>
                              <TLV-Value>23</TLV-Value>
                    </Check>
             </Device>
             <Device>
                  <DeviceType>Sample_Profile_2</DeviceType>
                  <RuleName>Sample_Rule_2</RuleName>
                  <RuleOperator>AND</RuleOperator>
                  <RuleCertaintyMetric>30</RuleCertaintyMetric>
                  <Check>
                         <Protocol>DHCP</Protocol>
                        <TLV-Type>12</TLV-Type>
                        <TLV-Value-Type></TLV-Value-Type>
                        <TLV-Value>test</TLV-Value>
                  </Check>
                  <Check>
                         <Protocol>MAC</Protocol>
                         <TLV-Value-Type>String</TLV-Value-Type>
                         <TLV-Value>Cisco</TLV-Value>
                   </Check>
</Device>
</DeviceList>
```

Each time you copy a new dynamic XML file, the older user profiles are erased completely and newer profiles are populated. After copying the dynamic XML files, only the newly connected clients are classified based on the new dynamic file whereas the already classified devices still remain as classified with older profiles until they are reconnected.

MAC OUI-based profiles configuration

The Organizational Unique Identifier (OUI) of a MAC address is part of the MAC address that identifies the vendor of the network adapter. The OUI is the first three bytes of the six-byte field and administered by the IEEE.

To define MAC-based profiles in the dynamic XML file, see https://standards-oui.ieee.org/.

For example, if the Client MAC address is **7035.094d.000**, then OUI is **0x703509**. You can find the corresponding entry in the https://standards-oui.ieee.org/ as shown below:

```
70-35-09   (hex)  Cisco Systems, Inc
703509     (base 16)    Cisco Systems, Inc
      80 West Tasman Drive
      San Jose  CA  94568
      US
```

Feature history for device classifier dynamic XML support

Lists feature history for device classifier dynamic XML support.

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 188: Feature history for device classifier dynamic XML support

Release	Feature	Feature Information
Cisco IOS XE Dublin 17.10.1	Device Classifier Dynamic XML Support	You can: • Add rules, checks, and profile name to an XML file. • Upload the XML file to the device file system. This feature enables better device classification without upgrading the device to a new release. **Note** Device classifier dynamic XML support is applicable for these: • Devices that are not classified previously: The classification takes effect from the latest file without any reboot. • Devices that are already classified: The clients have to rejoin for the classification to take effect. • Client previously classified with higher protocol values such as DHCP + HTTP: If the same client wants to be classified with only DHCP, use the **clear wireless client device cache** command.

Enable device classifier (CLI)

Enable device classifier using commands.

Enable the device classifier feature to automatically identify and classify attached devices on the network.

Device classifier is used to identify and classify devices that are connected to the network based on their behavior and characteristics. This feature helps with network management and policy enforcement.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enable the classification of attached devices.

Example

```
Device(config)# device classifier
```

3. Return to privileged EXEC mode.

Example

```
Device(config)# end
```

Device classifier is now enabled and will automatically identify and classify devices that connect to the network.

Update dynamic XML file

Configure dynamic XML file for device classification.

Update the dynamic XML file to classify devices by adding device configuration information.

Device classification requires specific XML entries in the dynamic XML file to properly identify and categorize devices.

Procedure

Add these lines in the dynamic XML file to classify a device:

Example

```
<DeviceList>
          <CopyRight>Copyright (c) 2021-2022 by Cisco Systems, Inc. All rights
 reserved.</CopyRight>
           <Version>1.1</Version>
           <Device>
                  <DeviceType>Device-test"</DeviceType>
                 <RuleName>Rule-Test</RuleName>
                <RuleOperator>AND</RuleOperator>
                <RuleCertaintyMetric>20</RuleCertaintyMetric>
                <Check>
                        <Protocol>DHCP</Protocol>
                        <TLV-Type>12</TLV-Type>
                       <TLV-Value-Type>String</TLV-Value-Type>
                       <TLV-Value>test</TLV-Value>
                 </Check>
          </Device>

.............................
</DeviceList>
```

The dynamic XML file is updated with the device classification configuration.

Verify TLV value

Provides the command to verify TLV values.

To verify the TLV values, use this command:

```
Device# show  wireless  client  mac-address 7035.094d.0001 detail
Client MAC Address : 7035.094d.0001

……

Nearby AP Statistics:

EoGRE : Pending Classification

Device Classification Information:

  Device Type         : Un-Classified Device

  Device Name      : Unknown Device

  Protocol Map     : 0x000009  (OUI, DHCP)

  Device Protocol  : DHCP                       -→ <Protocol>DHCP</Protocol>

    Type             : 12   14                           -->
<TLV-Type>12</TLV-Type>

    Data             : 0e

    00000000  00 0c 00 0a 74 65 73 74  2d 30 30 30 30 31        |....test-00001
  |           --><TLV-Value>test</TLV-Value>

    Type             : 60   8

    Data             : 08

    00000000  00 3c 00 04 74 65 73 74                           |.<..test
     |

    Type             : 55   11

    Data             : 0b

    00000000  00 37 00 07 01 1c 02 03  0f 06 0c                 |.7.........
     |

Max Client Protocol Capability: Wi-Fi6 (802.11ax)
```

Old classification cache clearing

Provides the command to clear old classification cache when classified clients use type-length-values with lower combination values.

If an already classified client uses any of the three type-length-values (TLVs) [OUI, DHCP, or HTTP] and if the combination value is lower, the existing value is ignored. To avoid such a scenario, use the clear wireless client device cache command.

```
Device# clear wireless client device cache
```

Note

The priority of the TLVs is as follows:

- OUI
- DHCP
- HTTP

After executing the clear command, you must rejoin the client to get it classified as per the latest XML file.

212 BSSID Counters

Topics:

- BSSID counters

BSSID counters

Explains a feature that helps to retrieve the BSSID statistics at certain intervals for a client associated with a WLAN.

A BSSID counter is a statistical feature that

- retrieves BSSID statistics when a client is associated with a WLAN for every configured interval
- requires configuration in the controller per AP profile to enable or disable BSSID statistics on the access points, and
- is disabled by default.

BSSID counter limitations

BSSID counter is not supported on the Cisco Aironet 1800 series APs and Cisco Catalyst 9100 series APs.

Enable BSSID statistics and BSSID neighbor statistics (CLI)

Configure BSSID statistics and BSSID neighbor statistics to monitor wireless network performance and neighboring access points.

Enable BSSID statistics collection and neighbor statistics reporting to monitor AP performance and neighboring wireless networks.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the AP profile configuration submode.

   Example

   ```
   Device(config)# ap profile ap-profile-name
   ```

   ap-profile-name is the profile name of the configured AP.

3. Enable BSSID statistics.

   Example

   ```
   Device(config-ap-profile)# bssid-stats
   ```

   Use the **no** form of the command to disable the feature.

4. Set the BSSID stats frequency timer.

   Example

   ```
   Device(config-ap-profile)# bssid-stats bssid-stats-frequency
   bssid-timer-seconds
   ```

Example

```
Device(config-ap-profile)# bssid-stats bssid-stats-frequency 40
```

BSSID statistics frequency timer is in the range of 1 to 180 seconds.

5. Enable BSSID neighbor statistics.

Example

```
Device(config-ap-profile)# bssid-neighbor-stats
```

Use the **no** form of the command to disable the feature.

6. Set the interval at which BSSID neighbor statistics is sent from the AP.

Example

```
Device(config-ap-profile)# bssid-neighbor-stats interval bssid-interval
```

Example

```
Device(config-ap-profile)# bssid-neighbor-stats interval 50
```

The BSSID neighbor stats interval is in the range of 1 to 180 seconds.

BSSID statistics and BSSID neighbor statistics are now enabled and will be collected at the specified intervals for network monitoring and analysis.

BSSID statistics verification on the controller

Lists commands to verify BSSID statistics on the controller.

To verify the BSSID statistics on the controller, use these commands:

- **show wireless stats ap name** *ap-name* **dot11 24ghz slot** *0***wlan-id** *<wlan-id>* **statistics**

```
Device# show wireless stats ap name APXXXX.6DXX.58XX dot11 24ghz slot 0
wlan-id 18 stat
BSSID                  : 7069.5a38.112e
WLAN ID                : 18
Client Count           : 1
TX Statistics
------------------------------------------------------------------------------
Mgmt         Retries      Data Bytes      Data Retries       Subframe Retries

------------------------------------------------------------------------------
12            18           16081           18                  0

RX Statistics
------------------------------------------------------------------------------
Mgmt          Data Bytes

------------------------------------------------------------------------------
74            17693
Data Distribution
------------------------------------------------------------------------------
```

```
Bytes                         RX                          TX
------------------------------------------------------------------------------
0-64                          55                          93
65-128                        66                          40
129-256                       21                          5
257-512                       10                          3
513-1024                      1                           9
1025-2048                     0                           1
2049-4096                     0                           0
4097-8192                     0                           0
8193-16384                    0                           0
16385-32768                   0                           0
32769-65536                   0                           0
65537-131072                  0                           0
131073-262144                 0                           0
262145-524288                 0                           0
524289-1048576                0                           0
WMM Statistics
------------------------------------------------------------------------------
                              RX                          TX
------------------------------------------------------------------------------
Voice                         0                           43
Video                         0                           0
Best Effort                   154                         39
Background                    0                           0
MCS
------------------------------------------------------------------------------
MCS                           RX                          TX
------------------------------------------------------------------------------
mcs0                          39                           0
mcs1                          2                            0
mcs2                          5                            0
mcs3                          7                            0
mcs4                          25                           0
```

```
mcs5                        59                          0

mcs6                        290                         0

mcs7                        1148                        3

mcs8                        2288                        0

mcs9                        4440                        2
```

- **show ap name *ap_name*neighbor summary**

```
Device#show ap name APXXXX.6DXX.59XX neighbor summary

BSSID                   Channel   Channel-width Slot        RSSI
   Last-Heard                           SSID
       Neighbour
-------------------------------------------------------------------------------------------------------
0008.2f1c.8040          1         20 Mhz        0           -39
   03/17/2020 18:25:14                  aprusty-un-dot1x
       FALSE
0008.2f1c.8041          1         20 Mhz        0           -39
   03/17/2020 18:25:14                  aprusty-sim-11
       FALSE
0008.2f1c.8042          1         20 Mhz        0           -39
   03/17/2020 18:25:14                  one-ph
       FALSE
0008.2f1c.8044          1         20 Mhz        0           -38
   03/17/2020 18:25:14                  aprusty-test
       FALSE
0008.3296.f340          11        20 Mhz        0           -51
   03/18/2020 10:39:27                  ewlc-ap-dot1x
       FALSE
0008.3296.f341          11        20 Mhz        0           -49
   03/18/2020 10:39:27                  vewlc_small_psk
       FALSE
002a.1022.d950          1         20 Mhz        0           -57
   03/17/2020 18:25:14                  ewlc-ap-dot1x
       FALSE
002a.105c.bfd0          1         20 Mhz        0           -36
   03/17/2020 18:25:14                  ewlc-ap-dot1x
       FALSE
002a.105c.bfd1          1         20 Mhz        0           -37
   03/17/2020 18:25:14                  vewlc_small_psk
       FALSE
002c.c864.76d0          11        20 Mhz        0           -61
   03/18/2020 10:37:37                  rajwlan
       FALSE

BSSID                   Channel   Channel-width Slot        RSSI
   Last-Heard                           SSID
       Neighbour
-------------------------------------------------------------------------------------------------------
002c.c8de.59e0          1         20 Mhz        0           -48
   03/17/2020 18:25:14                  WQ
       FALSE
002c.c8de.5d80          11        20 Mhz        0           -54
   03/18/2020 10:39:27                  ewlc-ap-dot1x
       FALSE
002c.c8de.5d81          11        20 Mhz        0           -55
   03/18/2020 10:39:27                  vewlc_small_psk
       FALSE
```

```
002c.c8de.7260          11         20 Mhz        0           -53
   03/18/2020 10:39:27                  ewlc-ap-dot1x
       FALSE
002c.c8de.7261          11         20 Mhz        0           -54
   03/18/2020 10:39:27                  vewlc_small_psk
       FALSE
005d.7390.e1e0          1          20 Mhz        0           -54
   03/17/2020 18:25:14                  rlan
       FALSE
006b.f114.95a0          1          20 Mhz        0           -60
   03/17/2020 18:25:14                  zavc
       FALSE
006b.f114.b0e0          1          20 Mhz        0           -46
   03/17/2020 18:25:14                  ewlc-ap-dot1x
       FALSE
006c.bc61.2340          1          20 Mhz        0           -63
   03/17/2020 18:24:44                  dnac-swim
       FALSE
006c.bc72.5ce0          11         20 Mhz        0           -58
   03/18/2020 10:39:17                  dnac-swim
       FALSE
```

213 Fastlane+

Topics:

- Fastlane+

Introduces the Fastlane+ feature for enhanced network performance and provides comprehensive guidance on configuration, monitoring, and verification across both command-line and graphical interfaces.

Fastlane+

Describes the Fastlane+ feature that improves the effectiveness of estimating the uplink buffer status for clients.

Fastlane+ is a feature that

- improves the effectiveness of estimating the uplink buffer status for clients
- enhances the user experience for latency-sensitive applications, and
- allows scheduled access-based uplink transmissions by periodically collecting buffer status reports from clients.

Fastlane+ configuration and support

The Fastlane+ feature can be enabled or disabled on a per-WLAN basis. Support for this feature is indicated in the beacons and probe responses transmitted by an AP.

> **Note**
>
> This feature works only if Protected Management Frame (PMF) is configured as optional or mandatory for a WLAN.

Configure Fastlane+ on a WLAN (CLI)

Configure Fastlane+ on a WLAN using commands.

This task enables the Fastlane+ feature on a WLAN to optimize wireless network performance and user experience.

Use this procedure when you need to enable Fastlane+ on a specific WLAN through the command-line interface.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a WLAN and enters WLAN configuration submode.

   Example

   ```
   Device(config)# wlan profile-name WLAN-ID SSID-name
   ```

   Example

   ```
   Device(config)# wlan wlan-test 3 ssid-test
   ```

> **Note**
>
> If you have already configured a WLAN, enter the **WLAN** *profile-name* command.

3. Configure Fastlane+ feature on a WLAN.

Example

```
Device(config-wlan)# scheduler asr
```

Fastlane+ is now configured on the specified WLAN. The WLAN will use optimized scheduling to improve wireless performance.

Configure Fastlane+ on a WLAN (GUI)

Configure Fastlane+ on a WLAN using the GUI.

Enable advanced scheduling requests handling on a per-WLAN basis to optimize wireless network performance through Fastlane+ configuration.

Fastlane+ provides enhanced scheduling capabilities for wireless networks. This feature can be configured on individual WLANs to improve network efficiency and client experience.

Procedure

1. Choose **Configuration > Tags & Profiles > WLANs**.
2. Select a WLAN.
3. Click **Advanced** tab.
4. Check the **Advanced Scheduling Requests Handling** check box to enable the feature on a per-WLAN basis.
5. Click **Update & Apply to Device**.

Fastlane+ is now enabled on the selected WLAN with advanced scheduling requests handling configured.

Monitor Fastlane+ (GUI)

Monitor Fastlane+ client statistics and capabilities through the wireless client monitoring interface.

This task enables you to monitor Fastlane+ client performance and view uplink latency statistics and client capabilities information.

Use this procedure to access and review Fastlane+ monitoring data for wireless clients, including latency distribution and feature-related capabilities.

Procedure

1. Choose **Monitoring > Wireless > Clients**.
2. Click a client name from the client list.

The **Client** window with multiple tabs is activated.

3. Click **General** tab.
4. Click **Client Statistics** tab.

   The most recent uplink latency statistics received from the client is displayed in the **Uplink Latency Distribution** section.

5. Click **Client Properties** tab.

   The Fastlane+ feature-related client capabilities information is displayed at the bottom of the window.

You can now view Fastlane+ client statistics including uplink latency distribution and client capabilities information through the monitoring interface.

Fastlane+ verification

Lists commands to verify Fastlane+ configuration, capability information, and statistics.

Verify whether Fastlane+ is enabled or disabled for a WLAN, capability information, client uplink latency statistics, and scheduling statistics for Fastlane+ clients.

WLAN Fastlane+ status verification

This example shows how to verify whether Fastlane+ is enabled or disabled for a WLAN:

```
Device# show wlan 2 | include ASR

Advanced Scheduling Requests Handling : Enabled
```

Client capability and latency statistics

This example shows how to verify Fastlane+ capability information and the most recent client uplink latency statistics:

```
Device# show wireless client mac-address f45c.89b0.xxxx detail
.
.
.
Regular ASR support: : ENABLED
Non-default Fastlane Profile: : Active
Range Voice Video Background Best-Effort
----------------------------------------------------------------------------------------
[0-20ms] 400 300 200 100
[20-40ms] 401 301 201 101
[40-100ms] 402 302 202 102
[>100ms] 403 303 203 103
```

All clients statistics verification

This example shows how to verify Fastlane+ statistics along with Fastlane+ capability and uplink latency statistics for all the Fastlane+ clients on a WLAN.

> **Note**
>
> **show interfaces dot11radio asr-info all** is an AP command, and does not work on the controller.

```
Device# show interfaces Dot11Radio 1 asr-info all

[*10/12/2020 18:45:21.0149]
[*10/12/2020 18:45:21.0150]  Client-MAC:[26:52:CF:C8:D0:1C] AID:[3]
ASR-Capability:[0x1]
[*10/12/2020 18:45:21.0150]  BE- LAT[0-20]:[267] LAT[20-40]:[57]
LAT[40-100]:[32] LAT[>100]:[26]
[*10/12/2020 18:45:21.0150]  BK- LAT[0-20]:[0] LAT[20-40]:[0] LAT[40-100]:[0]
 LAT[>100]:[0]
[*10/12/2020 18:45:21.0150]  VI- LAT[0-20]:[0] LAT[20-40]:[0] LAT[40-100]:[0]
 LAT[>100]:[0]
[*10/12/2020 18:45:21.0150]  VO- LAT[0-20]:[2222] LAT[20-40]:[409]
LAT[40-100]:[224] LAT[>100]:[163]
[*10/12/2020 18:45:21.0150]
[*10/12/2020 18:45:21.0206] HTT_PEER_DETAILS_TLV:
[*10/12/2020 18:45:21.0206] peer_type = 0
[*10/12/2020 18:45:21.0206] sw_peer_id = 98
[*10/12/2020 18:45:21.0206] vdev_id = 25
[*10/12/2020 18:45:21.0206] pdev_id = 0
[*10/12/2020 18:45:21.0206] ast_idx = 1187
[*10/12/2020 18:45:21.0206] mac_addr = 26:52:cf:c8:d0:1c
[*10/12/2020 18:45:21.0206] peer_flags = 0x200006f9
[*10/12/2020 18:45:21.0206] qpeer_flags = 0x8
[*10/12/2020 18:45:21.0206]
[*10/12/2020 18:45:21.0206] HTT_STATS_PEER_ASR_STATS_TLV
[*10/12/2020 18:45:21.0206] asr_bmap: 0x8
[*10/12/2020 18:45:21.0206] asr_muedca_update_cnt: 1
[*10/12/2020 18:45:21.0206] asr_muedca_reset_cnt: 1
[*10/12/2020 18:45:21.0206] asr_ul_mu_bsr_trigger: 2376
[*10/12/2020 18:45:21.0206] asr_min_trig_intv- BE:0         BK:0 VI:0 VO:19
[*10/12/2020 18:45:21.0206] asr_max_trig_intv- BE:0         BK:0 VI:0 VO:20
[*10/12/2020 18:45:21.0207] asr_min_alloc_rate- BE:0         BK:0 VI:0 VO:12
[*10/12/2020 18:45:21.0207] asr_ul_su_data_ppdu_cnt- BE:0         BK:0 VI:0
VO:2149
[*10/12/2020 18:45:21.0207] asr_ul_su_data_ppdu_bytes- BE:0         BK:0 VI:0
 VO:757546
[*10/12/2020 18:45:21.0207] asr_ul_mu_trig_ppdu_cnt- BE:0         BK:0 VI:0
VO:5002
[*10/12/2020 18:45:21.0207] asr_ul_mu_trig_ppdu_bytes- BE:0         BK:0 VI:0
 VO:2400960
[*10/12/2020 18:45:21.0207] asr_ul_mu_data_ppdu_cnt- BE:0         BK:0 VI:0
VO:2134
[*10/12/2020 18:45:21.0207] asr_ul_mu_data_ppdu_bytes- BE:0         BK:0 VI:0
 VO:736578
[*10/12/2020 18:45:21.0207] asr_ul_mu_data_padding_bytes- BE:0         BK:0
VI:0 VO:2953488
```

Individual client statistics verification

This example shows how to verify scheduling statistics along with capability and uplink latency statistics for a given client on a WLAN:

Note

The **show interfaces dot11radio asr-info** is an AP command and it will not work on the controller.

```
Device# show interfaces Dot11Radio 1 asr-info 26:XX:CF:XX:D0:XX

[*10/12/2020 18:45:21.0149]
[*10/12/2020 18:45:21.0150]  Client-MAC:[26:52:CF:C8:D0:1C] AID:[3]
ASR-Capability:[0x1]
[*10/12/2020 18:45:21.0150]  BE- LAT[0-20]:[267] LAT[20-40]:[57]
LAT[40-100]:[32] LAT[>100]:[26]
[*10/12/2020 18:45:21.0150]  BK- LAT[0-20]:[0] LAT[20-40]:[0] LAT[40-100]:[0]
 LAT[>100]:[0]
[*10/12/2020 18:45:21.0150]  VI- LAT[0-20]:[0] LAT[20-40]:[0] LAT[40-100]:[0]
 LAT[>100]:[0]
[*10/12/2020 18:45:21.0150]  VO- LAT[0-20]:[2222] LAT[20-40]:[409]
LAT[40-100]:[224] LAT[>100]:[163]
[*10/12/2020 18:45:21.0150]
[*10/12/2020 18:45:21.0206] HTT_PEER_DETAILS_TLV:
[*10/12/2020 18:45:21.0206] peer_type = 0
[*10/12/2020 18:45:21.0206] sw_peer_id = 98
[*10/12/2020 18:45:21.0206] vdev_id = 25
[*10/12/2020 18:45:21.0206] pdev_id = 0
[*10/12/2020 18:45:21.0206] ast_idx = 1187
[*10/12/2020 18:45:21.0206] mac_addr = 26:xx:cf:xx:d0:xx
[*10/12/2020 18:45:21.0206] peer_flags = 0x200006f9
[*10/12/2020 18:45:21.0206] qpeer_flags = 0x8
[*10/12/2020 18:45:21.0206]
[*10/12/2020 18:45:21.0206] HTT_STATS_PEER_ASR_STATS_TLV
[*10/12/2020 18:45:21.0206] asr_bmap: 0x8
[*10/12/2020 18:45:21.0206] asr_muedca_update_cnt: 1
[*10/12/2020 18:45:21.0206] asr_muedca_reset_cnt: 1
[*10/12/2020 18:45:21.0206] asr_ul_mu_bsr_trigger: 2376
[*10/12/2020 18:45:21.0206] asr_min_trig_intv- BE:0         BK:0 VI:0 VO:19
[*10/12/2020 18:45:21.0206] asr_max_trig_intv- BE:0         BK:0 VI:0 VO:20
[*10/12/2020 18:45:21.0207] asr_min_alloc_rate- BE:0         BK:0 VI:0 VO:12
[*10/12/2020 18:45:21.0207] asr_ul_su_data_ppdu_cnt- BE:0         BK:0 VI:0
VO:2149
[*10/12/2020 18:45:21.0207] asr_ul_su_data_ppdu_bytes- BE:0         BK:0 VI:0
 VO:757546
[*10/12/2020 18:45:21.0207] asr_ul_mu_trig_ppdu_cnt- BE:0         BK:0 VI:0
VO:5002
[*10/12/2020 18:45:21.0207] asr_ul_mu_trig_ppdu_bytes- BE:0         BK:0 VI:0
 VO:2400960
[*10/12/2020 18:45:21.0207] asr_ul_mu_data_ppdu_cnt- BE:0         BK:0 VI:0
VO:2134
[*10/12/2020 18:45:21.0207] asr_ul_mu_data_ppdu_bytes- BE:0         BK:0 VI:0
 VO:736578
[*10/12/2020 18:45:21.0207] asr_ul_mu_data_padding_bytes- BE:0         BK:0
VI:0 VO:2953488
```

214 Workgroup Bridges

Topics:

- Cisco Workgroup Bridges
- Configure AP as WGB
- Simplified WGB configuration

This chapter describes configuring and managing Cisco Workgroup Bridges (WGB), including supported platforms, WLAN and SSID setup, radio and timeout settings, certificate enrollment, RADIUS/EAP configuration, and verification commands to bridge wired clients over wireless

Cisco Workgroup Bridges

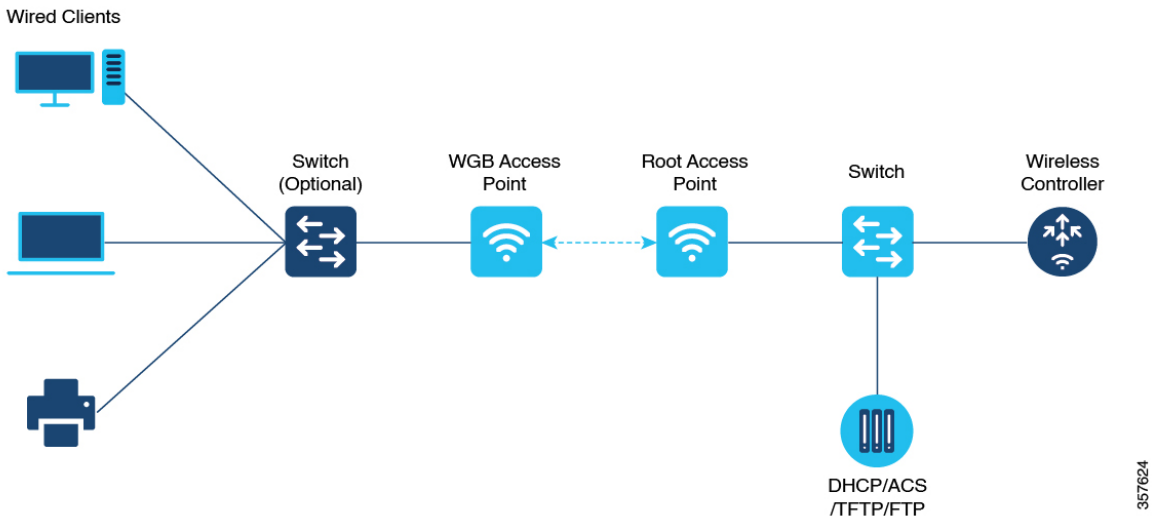
Introduces Cisco Workgroup Bridges, describing key operational concepts, listing restrictions, and providing instructions for configuring Workgroup Bridge on WLAN as well as monitoring status for effective network integration.

A workgroup bridge (WGB) is an access point mode that

- provides wireless connectivity to wired clients connected on its Ethernet port
- connects a wired network to an existing wireless infrastructure through a single wireless segment, and
- learns the MAC addresses of its wired clients and reports them to the wireless LAN controller (WLC) using IAPP messaging.

The WGB establishes a single wireless connection to the root access point (AP). The root AP treats the WGB as a wireless client, allowing the wired devices behind the WGB to access the wireless network seamlessly. Suppose you have a group of wired devices in a conference room with no Ethernet drop available. You can connect these devices to a WGB, which allows them to communicate over the enterprise wireless network as if they were wireless clients.

Figure 37: Example of a WGB



Starting from Cisco IOS XE Cupertino 17.8.1, WGB is supported on the following Cisco Catalyst 9100 Series Access Points.

- Cisco Catalyst 9105
- Cisco Catalyst 9115
- Cisco Catalyst 9120

Starting from Cisco IOS XE Dublin 17.10.1, WGB is supported on the following Cisco Catalyst 9100 Series Access Points.

- Cisco Catalyst 9124
- Cisco Catalyst 9130

From Cisco IOS XE Cupertino 17.9.1 onwards, WGB supports one radio for uplink (backhaul) connectivity and another radio for serving wireless clients. This feature is supported on the Cisco 11AX APs such as Cisco Catalyst 9105 APs, Cisco Catalyst 9115 APs, Cisco Catalyst 9120 APs.

OPEN and PSK security (WPA2 Personal) based wireless clients can be associated to WGB independent of its uplink connectivity, but they will not be able to pass traffic unless WGB has uplink connectivity. Radius server must be configured

and the WGB should have uplink connectivity for authentication of wireless clients to 802.1x security (WPA2 Enterprise) WLAN. Both IPv4 and IPv6 traffic forwarding is supported for wireless clients. Static IP and Passive Client support is enabled by default on these WLANs.

These features are supported for use with WGB:

Table 189: WGB Feature Matrix

Feature	Cisco Wave 2 APs	Cisco 11AX APs
802.11r	Supported	Supported
QOS	Supported	Supported
UWGB mode	Supported	Not supported
IGMP Snooping or Multicast	Supported	Supported
802.11w	Supported	Supported
PI support (without SNMP)	Not supported	Not supported
IPv6	Supported	Supported
VLAN	Supported	Supported
802.11i (WPAv2)	Supported	Supported
Broadcast tagging/replicate	Supported	Supported
Unified VLAN client	Supported	Supported
WGB client	Supported	Supported
802.1x - PEAP, EAP-FAST, EAP-TLS	Supported	Supported
NTP	Supported	Supported
Wired client support on all LAN ports	Supported in all Wired-0, 1 and LAN ports 1, 2, and 3	Supported in all Wired-0, 1 and LAN ports 1, 2, and 3
Second radio wireless client support	Not supported	Supported

This table shows the supported and unsupported authentication and switching modes for Cisco APs when connecting to a WGB.

> **Note**
>
> Workgroup Bridge mode is supported on the WiFi6 Pluggable Module from Cisco IOS XE Bengaluru 17.6.1.

Table 190: Supported Access Points and Requirements

Access Points	Requirements
Cisco Aironet 2800, 3800, 4800, 1562, and Cisco Catalyst 9105, 9115, 9120, 9124, and 9130, IW6300 and ESW6300 Series	CAPWAP image starting from Cisco AireOS 8.8 release.

Restrictions for WGB

These are the restrictions for WGB:

- Cisco Wave 2 and 802.11ax APs operating in WGB mode can connect only to a Cisco root APs with Aironet IE/CCX enabled on the WLAN. WGB connectivity to a non-Cisco root access point is not supported in this mode.

  For non-Cisco root AP connectivity, use uWGB mode where applicable. However, uWGB mode supports only one wired client behind the WGB.

- MAC filtering is not supported for wired clients.
- Idle timeout is not supported for both WGB and wired clients.
- Session timeout is not applicable for wired clients.
- Web authentication is not supported.
- The total number of clients supported by WGB (wired + wireless) is limited to 20 clients.
- If you want to use a chain of certificates, copy all the CA certificates to a file and install it under a trust point on the WGB, else server certificate validation may fail.
- Wired clients connected to a WGB inherit the WGB's QoS and AAA override attributes.
- To enable the WGB to communicate with the root AP, create a WLAN and make sure that Aironet IE is enabled under the Advanced settings.
- WPA2 Enterprise security works only if the uplink WLAN is enabled for FlexConnect local switching or Fabric enabled WLAN.
- Radius override is not supported for wireless clients that are associated with WGB WLANs.
- WGB does not support 802.1X wired client authentication when used with power injector.

  The power-injector drops all EAPOL packets received from the wired client and does not forward it to the WGB's wired0 interface. In such cases, use PoE plus hub behind the wired0 interface and connect the wired clients to the hub.

- After WGB reload, the WGB 802.1X wired clients behind a hub do not trigger authentication automatically, unless done manually.

  After WGB is reloaded the WGB dot1x wired clients which are behind a hub remain authenticated or connected on their side and do not get notified that the WGB is reloaded. Clients are also not shown on the WGB bridge table. The client interfaces must be manually disabled and enabled back to trigger authentication.

- When the 802.1X wired client Ethernet interface is disabled and then enabled again, client authentication might fail for some of dot1x wired clients, at times.

Configure a WGB on a WLAN (CLI)

Use this procedure to configure a workgroup bridge (WGB) on a WLAN to enable network connectivity for wired devices through a wireless link.

Deploy a WGB so that connected wired devices can access the network via WLAN.

WGBs allow wired devices to join a wireless network by bridging their wired connections over Wi-Fi. This configuration is used in environments where direct wireless capability is not available for all devices.

Before you begin

Ensure WLAN security (such as authentication and encryption) is already configured.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter WLAN configuration submode.

   Example

   ```
   Device(config)# wlan profile-name
   ```

   The *profile-name* is the profile name of the configured WLAN.

3. Configure the Cisco Client Extensions option and sets the support of Aironet IE on the WLAN.

   Example

   ```
   Device(config-wlan)# ccx aironet-iesupport
   ```

4. Exit the WLAN configuration submode.

   Example

   ```
   Device(config-wlan)# exit
   ```

5. Configure WLAN policy profile and enters the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy profile-policy
   ```

6. Add a description for the policy profile.

Example

```
Device(config-wireless-policy)# description "test-wgb"
```

7. Assign the profile policy to the VLAN.

   Example

   ```
   Device(config-wireless-policy)# vlan vlan-no
   ```

8. Configure WGB VLAN client support.

   Example

   ```
   Device(config-wireless-policy)# wgb vlan
   ```

9. Configure WGB broadcast tagging on a WLAN.

   Example

   ```
   Device(config-wireless-policy)# wgb broadcast-tagging
   ```

10. Restart the policy profile.

    Example

    ```
    Device(config-wireless-policy)# no shutdown
    ```

11. Exit the wireless policy configuration mode.

    Example

    ```
    Device(config-wireless-policy)# exit
    ```

12. Configure policy tag and enters policy tag configuration mode.

    Example

    ```
    Device(config)# wireless tag policy policy-tag
    ```

13. Map a policy profile to a WLAN profile.

    Example

    ```
    Device(config-policy-tag)# wlan profile-name policy profile-policy
    ```

14. Exit policy tag configuration mode, and returns to privileged EXEC mode.

    Example

    ```
    Device(config-policy-tag)# end
    ```

The WGB is now configured and can join the designated WLAN, bridging its wired clients to the wireless network.

Verify WGB status on the controller

Lists commands to verify the status of a workgroup bridge on the controller.

Use these commands to verify the status of a WGB.

To display the wireless-specific configuration of active clients, use this command:

```
Device# show wireless client summary
```

To display the WGBs on your network, use this command:

```
Device# show wireless wgb summary
```

To display the details of wired clients that are connected to a particular WGB, use this command:

```
Device# show wireless wgb mac-address 00:0d:ed:dd:25:82 detail
```

Configure AP as WGB

Guides configuration of access points as Workgroup Bridges, including tasks to enter autonomous mode, set up Wave 2 APs, configure SSID and EAP profiles, authentication servers, 802.1X credentials, certificate enrollment, radio interface, timeouts, and bridge forwarding using CLI commands.

Configure Cisco Wave 2 APs or 11AX APs in WGB or CAPWAP AP mode (CLI)

Learn how to configure Cisco Wave 2 or 11AX Access Points as workgroup bridges or CAPWAP APs using command-line interface (CLI) commands.

Set up the AP mode and management configuration for Cisco APs using the CLI.

This task is applicable when you need to quickly reconfigure the AP operation mode between Workgroup Bridge and CAPWAP using the AP's command-line interface.

Procedure

1. Enter the privileged mode of the AP.

   Example

   ```
   Device> enable
   ```

2. Move the AP in to the Workgroup Bridge mode.

   Example

   ```
   Device# ap-type workgroup-bridge
   ```

3. Configure DHCP or static IP address using the **configure ap address ipv4 dhcp** or **configure ap address ipv4 static***ip-address netmask gateway-ipaddress* command.

Example

DHCP IP Address

```
Device# configure ap address ipv4 dhcp
```

Static IP Address

```
Device# configure ap address ipv4 static 10.10.10.2 255.255.255.234 192.168.4.1
```

4. Configure an username for the AP management.

Example

```
Device# configure ap management add username username password password secret
 secret
```

5. Configure the AP hostname.

Example

```
Device# configure ap hostname host-name
```

The AP is successfully configured in the selected mode with the desired network and management settings.

Configure an SSID profile for Cisco Wave 2 and 11AX APs (CLI)

Configure an SSID profile for Cisco Wave 2 and 11AX APs using CLI commands on the AP console.

Set up secure Wi-Fi profiles for Wave 2 and 11AX APs, allowing client authentication and traffic management using the AP CLI.

This configuration is performed on the AP console and does not apply to the controller CLI. Choose authentication and QoS settings best suited for your environment.

Before you begin

Ensure you have access to the AP console and any preshared keys or EAP profiles needed for authentication.

Procedure

---

1. Create an SSID profile and specify authentication (open, PSK, or EAP) and key management options.

Example

SSID profile with open authentication:

```
Device# configure ssid-profile test WRT s1 authentication open
```

SSID profile with PSK authentication:

```
Device# configure ssid-profile test WRT s1 authentication psk 1234
key-management dot11r
```

SSID profile with EAP authentication:

```
Device# configure ssid-profile test WRT s1 authentication eap profile test2
key-management dot11r
```

Choose the authentication protocol that matches your deployment requirements.

2. Attach the SSID profile to a radio interface.

Example

```
Device# configure dot11radio radio-interface mode wgb ssid-profile profle-name
```

Binds the SSID profile for wireless bridge operations.

3. Optional: Configure the DTIM period for the SSID profile.

Example

```
Device# configure ssid-profile profile-name ssid ssid-name dtim-period
ssid-name
```

Sets the delivery traffic indication message interval for broadcast/multicast traffic.

> **Note**
>
> Supported for wireless clients from Cisco IOS XE Cupertino 17.9.1 onwards.

4. Optional: Create a QoS profile.

Example

```
Device# configure qos profile qos-profile gold
```

Gold, silver, platinum, and bronze options are available to prioritize traffic.

5. Optional: Map the QoS profile to the SSID profile.

Example

```
Device# configure ssid-profile profile-name ssid ssid-name qos profile
qos-profile-name
```

Enables traffic prioritization for the specified SSID profile.

> **Note**
>
> Supported for wireless clients from Cisco IOS XE Cupertino 17.9.1 onwards.

6. Optional: Delete an SSID profile.

Example

```
Device# configure ssid-profile profile-name delete
```

7. Optional: Display summary of configured and connected SSIDs.

Example

```
Device# show wgb ssid
```

8. Optional: Display management, control, and data packet statistics for WGB SSIDs.

Example

```
Device# show wgb packet statistics
```

The SSID profile is configured for the AP and mapped to the desired radio interface. Wireless clients can now connect using the specified authentication mode, traffic is prioritized according to QoS settings, and you can monitor SSID and packet statistics as needed.

Configure the authentication server (CLI)

Configure the authentication server on the device using command-line interface (CLI) commands.

Set up a primary or secondary RADIUS authentication server for network access control using CLI.

RADIUS servers provide centralized authentication, authorization, and accounting (AAA) management for users connecting to the network.

Before you begin

Ensure you have the IP address, port number, and shared secret for the RADIUS server you are configuring.

Procedure

Configure a RADIUS authentication server with a primary or secondary role, IP version (IPv4 or IPv6), port, server address, and shared secret.

Example

```
Device# configure radius authentication primary | secondary add ipv4|ipv6
address radius-server-ip-address port radius-server-port-number secret
radius-secret
```

```
Device# configure radius authentication primary add ipv4 address 192.168.1.2
port 1812 secret Cisco123
```

a. **primary | secondary**: specifies if the RADIUS server is primary or secondary.

b. **ipv4 | ipv6**: identifies the IP address type.

c. **radius-server-ip-address**: is the management IP address of the RADIUS server.

d. **radius-server-port-number**: is typically 1812 for authentication.

e. **radius-secret**: is a shared secret between the device and the RADIUS server (case-sensitive).

The RADIUS authentication server is now configured. The device can authenticate users against the designated server.

Configure dot1X credential (CLI)

Use this procedure to configure a Dot1X credential using CLI commands on your device.

Configure or remove a Dot1X credential profile, and manage WGB client authentication using CLI commands.

Dot1X credentials are used to authenticate devices securely to the network. You can create or remove these credentials as required, and clear authenticated Workgroup Bridge (WGB) clients using dedicated commands.

Before you begin

Ensure the device is in global configuration mode before issuing these commands.

Procedure

1. Configure a dot1X credential profile with username and password.

   Example

   ```
   Device(config)# configure dot1x credential profile-name username username
   password password
   ```

2. Delete a dot1X credential profile.

   Example

   ```
   Device(config)# configure dot1x credential profile-name delete
   ```

3. Deauthenticate a WGB client by clearing a WGB client session based on MAC address or clear all WGB clients.

   Example

   ```
   Device# clear wgb client single mac-address
   ```

   To clear all WGB clients, use:

   ```
   Device# clear wgb client all
   ```

Dot1X credentials and WGB client sessions are managed according to your chosen operations.

Configure an EAP profile (CLI)

Configure an EAP profile using commands.

Use this procedure to create, manage, or delete EAP profiles on your device for authentication using the CLI.

An EAP profile defines the authentication method for wireless users. You can configure various methods and associate Trustpoints or credentials to strengthen security for wireless authentication.

Before you begin

Enable global configuration mode on the device and verify access to CLI.

Procedure

1. Configure an EAP profile and set the EAP method.

   Example

   ```
   Device# configure eap-profile profile-name method method-name
   ```

   Configures an EAP profile with the chosen method.

   The method-name can be:

   - **fast**
   - **peap**
   - **tls**

2. Associate a Trustpoint with the EAP profile. Use either the default or a specific name.

   Example

   EAP profile to Trustpoint with MIC Certificate:

   ```
   Device# configure eap-profile profile-name trustpoint default
   ```

   EAP profile to Trustpoint with CA Certificate:

   ```
   Device# configure eap-profile profile-name trustpoint name trustpoint-name
   ```

   Configures an EAP profile with a Trustpoint for authentication.

3. Attach a CA Trustpoint to the EAP profile.

   Example

   ```
   Device# configure eap-profile profile-name trustpoint {default | name
   trustpoint-name}
   ```

   Attaches the CA Trustpoint.

> **Note**
>
> With the default profile, WGB uses the internal MIC certificate for authentication.

4. Configure the 802.1X credential profile for the EAP profile.

   Example

   ```
   Device# configure eap-profile profile-name dot1x-credential profile-name
   ```

   Configures the 802.1X credential profile associated with the EAP profile.

5. Optional: Delete an EAP profile.

   Example

   ```
   Device# configure eap-profile profile-name delete
   ```

6. Optional: Display the WGB EAP dot1x credential profile summary.

   Example

   ```
   Device# show wgb eap dot1x credential profile
   ```

7. Optional: Display the EAP profile summary.

   Example

   ```
   Device# show wgb eap profile
   ```

8. Optional: Display all configured EAP and dot1x profiles.

   Example

   ```
   Device# show wgb eap profile all
   ```

The EAP profile is now configured as required. You can display or modify EAP profiles as needed to support different authentication strategies.

Configure manual-enrollment of a Trustpoint for WGB (CLI)

Use this procedure to configure manual enrollment of a Trustpoint for workgroup bridge (WGB) using CLI commands.

Manually configure and enroll a Trustpoint on a WGB to enable certificate-based security authentication via CLI.

A Trustpoint defines a Certification Authority (CA) used to issue certificates for device authentication. Manual enrollment is used when automatic enrollment via HTTP/SCEP is not available.

Before you begin

Ensure that the WGB configuration is complete and the required CA certificate is accessible. You must have administrator access to the CLI and all necessary server and certificate information.

Procedure

1. Configure a Trustpoint on the WGB and set the enrollment method to terminal.

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-name enrollment terminal
   ```

   Defines a new Trustpoint and specifies that certificate content will be entered manually.

2. Authenticate the Trustpoint and input the base 64 encoded CA certificate.

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-name authenticate
   ```

   Authenticates a Trustpoint by pasting the CA-signed certificate. End certificate input by typing **quit** on a new line.

3. Configure the private key size for the Trustpoint.

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-name key-size key-length
   ```

   Sets the RSA key pair length for the Trustpoint's private key.

4. Configure the subject name attributes for the Trustpoint's certificate.

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-name subject-name name
   2ltr-country-code state-name locality org-name org-unit email
   ```

   Example

   ```
   Device# configure crypto pki trustpoint
   ca-server-US subject-name test US CA abc cisco AP test@cisco.com
   ```

   Specifies certificate subject fields, matching your organization's naming policy.

5. Generate a Certificate Signing Request (CSR) on the WGB and enroll the Trustpoint.

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-name enroll
   ```

   Generates the private key and a CSR; provide the CSR output to your CA server for certificate signing.

6. Import the signed certificate from the CA back into the WGB Trustpoint.

Example

```
Device# configure crypto pki trustpoint ca-server-name import certificate
```

Installs the issued device certificate. End certificate input with the **quit** command on a new line.

7. Optional: Delete the Trustpoint if reconfiguration is needed.

Example

```
Device# configure crypto pki trustpoint ca-server-name delete
```

Removes a Trustpoint configuration from the device.

8. Optional: Display the Trustpoint summary.

Example

```
Device# show crypto pki trustpoint
```

Shows all Trustpoints configured on the device.

9. Optional: Display certificates associated with a Trustpoint.

Example

```
Device# show crypto pki trustpoint ca-server-name certificate
```

Inspect the contents of certificates for a specific Trustpoint.

Manual Trustpoint enrollment is now complete on the WGB. The device can use the certificates for secure authentication and establish connections as required for network operation.

Configure auto-enrollment of a Trustpoint for workgroup bridge (CLI)

Use this procedure to configure auto-enrollment of a Trustpoint for a workgroup bridge (WGB) using CLI commands.

Enable automatic certificate enrollment for secure communication by configuring a Trustpoint and CA enrollment in a WGB deployment.

Auto-enrollment of a Trustpoint streamlines certificate management for workgroup bridges by allowing automatic acquisition and renewal of CA certificates.

Before you begin

Ensure the device is running the appropriate WGB image and has connectivity to the CA server.

Procedure

---

1. Configure the Trustpoint and specify the CA server enrollment URL.

Example

```
Device# configure crypto pki trustpoint ca-server-name enrollment url
ca-server-url
```

Enrolls a Trustpoint in WGB using the CA server URL.

2. Authenticate the Trustpoint by fetching the CA certificate from the CA server.

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-name authenticate
   ```

   Fetches the CA certificate for Trustpoint authentication.

3. Set the private key size for the Trustpoint.

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-name key-size key-length
   ```

   Example

   ```
   Device# configure crypto pki trustpoint
   ca-server-Us key-size 60
   ```

4. Configure the subject name for the trustpoint.

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-name subject-name name
   2ltr-country-code state-name locality org-name org-unit email
   ```

   Example

   ```
   Device# configure crypto pki trustpoint
   ca-server-US subject-name test US CA abc cisco AP test@cisco.com
   ```

   Customize the subject name parameters as needed for your deployment.

5. Enroll the Trustpoint to request a signed certificate from the CA server.

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-name enroll
   ```

   Initiates the certificate enrollment request to the CA server.

6. Enable auto-enrollment and specify the renewal percentage threshold.

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-name auto-enroll enable
   renew-percentage
   ```

   Example

   ```
   Device# configure crypto pki trustpoint
   ca-server-US auto-enroll enable  10
   ```

Use **disable** to turn off auto-enroll if needed.

7. Optional: Delete the Trustpoint if it is no longer required.

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-name delete
   ```

   Removes the Trustpoint from the configuration.

8. Optional: Display the summary of Trustpoints.

   Example

   ```
   Device# show crypto pki trustpoint
   ```

   Lists all configured Trustpoints and their status.

9. Optional: Display the certificate for a specific Trustpoint.

   Example

   ```
   Device# show crypto pki trustpoint ca-server-name certificate
   ```

   Shows the certificate configured for the specified Trustpoint.

10. Optional: Display the PKI timer information.

    Example

    ```
    Device# show crypto pki timers
    ```

    Displays timer settings related to PKI operations.

The Trustpoint is now auto-enrolled and managed on the WGB, ensuring secure and automated certificate renewal.

Configure manual certificate enrollment using TFTP server (CLI)

Learn how to configure manual certificate enrollment using TFTP server commands.

Manual certificate enrolment allows devices to obtain certificates from a TFTP server, which is essential for secure communications.

Procedure

---

1. Specify the enrollment method to retrieve the CA certificate and client certificate for a Trustpoint in WGB.

   Example

   ```
   Device# configure crypto pki trustpoint
   ca-server-name enrollment addr/file-name
   ```

2. Authenticate the CA certificate from the specified TFTP server.

Example

```
Device# configure crypto pki trustpoint
ca-server-name authenticate
```

Retrieves the CA certificate and authenticates it from the specified TFTP server. If the file specification is included, the WGB will append the extension ".ca" to the specified filename.

3. Configure the private key size for the Trustpoint.

Example

```
Device# configure crypto pki trustpoint
ca-server-name key-size key-length
```

4. Set the subject name for the Trustpoint.

Example

```
Device# configure crypto pki trustpoint
ca-server-name subject-name test US CA abc cisco AP test@cisco.com
```

5. Generate a private key and certificate signing request (CSR).

Example

```
Device# configure crypto pki trustpoint
ca-server-name enroll
```

Generate a private key and Certificate Signing Request (CSR) and write the request out to the TFTP server. The filename to be written is appended with the extension ".req".

6. Import the signed certificate into WGB using TFTP at the console terminal, which retrieves the granted certificate.

Example

```
Device# configure crypto pki trustpoint
ca-server-name import certificate
```

The WGB will attempt to retrieve the granted certificate using TFTP using the same filename and the file name appended with ".crt" extension.

7. Optional: Display the Trustpoint summary.

Example

```
Device# show crypto pki trustpoint
```

8. Optional: Display the content of the certificates that are created for a Trustpoint.

Example

```
Device# show crypto pki trustpoint trustpoint-name certificate
```

Import the PKCS12 format certificates from the TFTP server (CLI)

Import the PKCS12 format certificates from the TFTP server using commands.

Importing PKCS12 format certificates is essential for establishing secure communications in network devices.

Procedure

1. Import the PKCS12 format certificate from the TFTP server.

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-name import pkcs12 tftp
   addr/file-name password pwd
   ```

   Example

   ```
   Device# configure crypto pki trustpoint ca-server-US import pkcs12
   tftp://10.8.0.6/all_cert.p12 password ******
   ```

2. Optional: Display the Trustpoint summary.

   Example

   ```
   Device# show crypto pki trustpoint
   ```

3. Optional: Display the content of the certificates created for a Trustpoint.

   Example

   ```
   Device# show crypto pki trustpoint trustpoint-name certificate
   ```

   Example

   ```
   Device# show crypto pki trustpoint ca-server-US certificate
   ```

Configure radio interface for workgroup bridges (CLI)

Configure radio interface for workgroup bridges using commands.

From the available two radio interfaces, before configuring WGB or UWGB mode on one radio interface, configure the other radio interface to root AP mode.

Procedure

1. Configure a radio interface as root AP.

   Example

   ```
   Device# configure dot11Radio radio-int mode root-ap
   ```

Example

```
Device# configure dot11Radio 0/3/0 mode root-ap
```

Maps a radio interface as root AP.

> **Note**
>
> When an active SSID or EAP profile is modified, you need to reassociate the profile to the radio interface for the updated profile to be active.

2. Configure the WLAN at the root AP mode radio.

Example

```
Device# configure dot11radio value wlan add ssid-profile-name ssid-number
```

Example

```
Device# configure dot11radio dotradiovalue wlan add ssid-profile-name
ssid-number
```

Enter the SSID profile name and SSID number between 1 and 16. Value can be 0 or 1.

3. Delete WLAN from the radio configuration.

Example

```
Device# configure dot11radio dotradiovalue wlan delete ssid-profile-name
```

Enter the SSID profile name. Value can be 0 or 1.

4. Configure a radio channel to broadcast the SSID.

Example

```
Device# configure dot11radio value channel channel-number width
```

Example

```
Device# configure dot11radio value channel channel-number width
```

The channel numbers are between 1 and 173. The channel width values are 20, 40, 80, and 160.

> **Note**
>
> - Only 20MHz channel width is supported on radio 0 (2.4-GHz band).
> - If radar is detected on a configured channel on radio 1, then the channel automatically changes to a non-DFS channel with a channel width of 20MHz. The administrator must reset the radio to bring it back to the configured channel.

5. Configure the periodic beacon interval in milliseconds.

   Example

   ```
   Device# configure dot11radio dotradiovalue beacon-period beacon-interval
   ```

   The value range is from 2 to 2000 milliseconds.

6. Map a radio interface to a WGB SSID profile.

   Example

   ```
   Device# configure dot11Radio 0/3/0 mode wgb ssid-profile ssid-profile-name
   ```

7. Map a radio interface to a UWGB SSID profile.

   Example

   ```
   Device# configure dot11Radio radio-int mode uwgb mac-addr ssid-profile
   ssid-profile-name
   ```

8. Configure a radio interface.

   Example

   ```
   Device# configure dot11Radio radio-int mode enable
   ```

   After configuring the uplink to the SSID profile, we recommend that you disable and enable the radio for the changes to be active.

9. Configure a radio antenna.

   Example

   ```
   Device# configure dot11Radio radio-int antenna a-antenna
   ```

10. Configure the radio interface encryption mode.

    Example

    ```
    Device# configure dot11Radio radio-int encryption mode ciphers aes-ccm
    ```

11. Configure the device channel rate.

Example

```
Device# configure wgb mobile rate basic 6 9 18 24 36 48 54
```

12. Configure the threshold duration and signal strength to trigger scanning.

Example

```
Device# configure wgb mobile period seconds thres-signal
```

13. Configure the static roaming channel.

Example

```
Device# configure wgb mobile station interface dot11Radio 0/3/0 scan
channel-number add
```

14. Optional: Delete the mobile channel.

Example

```
Device# configure wgb mobile station interface dot11Radio 0/3/0 scan
channel-number delete
```

15. Optional: Disable the mobile channel.

Example

```
Device# configure wgb mobile station interface dot11Radio 0/3/0 scan disable
```

16. Optional: Configure the beacon miss-count.

Example

```
Device# configure wgb beacon miss-count value
```

> **Note**
>
> When you set the beacon miss-count value to 10 or lower, then the beacon miss-count gets disabled. Set the value to 11 or higher to enable this function.

17. Optional: Display the Wi-Fi station statistics.

Example

```
Device# show wgb wifi wifi-interface stats
```

18. Optional: Display the radio antenna statistics.

Example

```
Device# show controllers dot11Radio radio-interface antenna
```

19. Optional: Display the mobile station channels scan configuration.

Example

```
Device# show wgb mobile scan channel
```

20. Optional: Display the configuration that is stored in the NV memory.

Example

```
Device# show configuration
```

21. Optional: Display the running configuration in the device.

Example

```
Device# show running-config
```

Configure workgroup bridge timeouts (CLI)

Configure timeouts for workgroup bridge features using CLI commands.

Use this task to configure various timeout values that control workgroup bridge (WGB) behavior, improving reliability and performance.

- Set specific timeout limits for association, authentication, EAP, DHCP response, and channel scan processes.

Timeout values affect how long the WGB waits for events before triggering corrective actions or failure notices.

This task applies to Cisco devices in environments where precise timeout management is needed for workgroup bridge operations and troubleshooting.

Before you begin

Ensure you have privileged EXEC access on the device.

- Verify device compatibility and active WGB configuration.

Procedure

1. Configure the WGB association response timeout.

Example

```
Device# configure wgb association response timeout response-millisecs
```

The valid range is 300 to 5000 milliseconds. The default value is 5000 milliseconds.

2. Configure the WGB authentication response timeout.

Example

```
Device# configure wgb authentication response timeout response-millisecs
```

The valid range is 300 to 5000 milliseconds. The default value is 5000 milliseconds.

3. Configure the Universal WGB client response timeout.

Example

```
Device# configure wgb uclient timeout timeout-secs
```

The valid range is 1 to 65535 seconds. The default value is 60 seconds.

4. Configure the WGB EAP timeout.

Example

```
Device# configure wgb eap timeout timeout-secs
```

The valid range is 2 to 60 seconds. The default value is 3 seconds.

5. Configure the WGB channel scan timeout.

Example

```
Device# configure wgb channel scan timeout {fast | slow | medium}
```

Select scan speed according to site requirements.

6. Configure the WGB DHCP response timeout.

Example

```
Device# configure wgb dhcp response timeout timeout-secs
```

The valid range is 1000 to 60000 milliseconds. The default value is 60 seconds.

7. Display the WGB association summary.

Example

```
Device# show wgb dot11 association
```

Display associated WGB clients and related statistics.

The configured timeout parameters apply immediately and affect how the WGB responds to events and failures.

For example, to set the WGB association response timeout to 4000 milliseconds:

```
Device# configure wgb association response timeout 4000
```

What to do next

Verify timeout settings by using display commands and monitoring WGB behavior.

- Adjust timeouts as needed for specific network or client stability.

Configure bridge forwarding for workgroup bridge (CLI)

Configure bridge forwarding for workgroup bridge using commands.

Before you begin

The Cisco Wave 2 and 11AX APs as Workgroup Bridge recognizes the Ethernet clients only when the traffic has the bridging tag.

We recommend setting the WGB bridge client timeout value to default value of 300 seconds, or less in environment where change is expected, such as:

- Ethernet cable is unplugged and plugged back.
- Endpoint is changed.
- Endpoint IP is changed (static to DHCP and vice versa).

If you need to retain the client entry in the WGB table for a longer duration, we recommend you increase the client WGB bridge timeout duration.

Procedure

1. Add a WGB client using the MAC address.

   Example

   ```
   Device# configure wgb bridge client add mac-address
   ```

   Example

   ```
   Device# configure wgb bridge client add F866.F267.7DFB
   ```

2. Configure the WGB bridge client timeout. Default timeout value is 300 seconds. The valid range is from 10 to 1000000 seconds.

   Example

   ```
   Device# configure wgb bridge client timeout timeout-secs
   ```

   Example

   ```
   Device# configure wgb bridge client timeout 400
   ```

3. Display the WGB wired clients over the bridge.

   Example

   ```
   Device# show wgb bridge
   ```

4. Display the WGB Gigabit wired clients over the bridge.

Example

```
Device# show wgb bridge wired gigabitEthernet interface
```

Example

```
Device# show wgb bridge wired gigabitEthernet 0/1
```

5. Display the WGB bridge radio interface summary.

Example

```
Device# show wgb bridge dot11Radio interface-number
```

Example

```
Device# show wgb bridge dot11Radio 0/3/1
```

Simplified WGB configuration

Describes the capability to configure workgroup bridge (WGB) settings across multiple Cisco access points simultaneously using configuration import methods.

Simplified WGB configuration is a network deployment capability that

- enables configuration of WGB in multiple Cisco access points (APs) simultaneously
- allows deployment of multiple WGBs in a network through configuration import, making them operational quicker
- eliminates the need to configure multiple Cisco APs using CLIs after logging into each one individually.

Configuration deployment methods

A network administrator can onboard Cisco APs using either of these methods:

- Upload the working configuration from an existing Cisco AP to a server and download it to the newly deployed Cisco APs.
- Send a sample configuration to all the Cisco APs in the deployment.

This feature is supported only on these Cisco APs:

- Cisco Aironet 1562 Access Points
- Cisco Aironet 2800 Access Points
- Cisco Aironet 3800 Access Points
- Cisco Catalyst 9105 Access Points
- Cisco Catalyst 9115 Access Points
- Cisco Catalyst 9120 Access Points
- Cisco Catalyst IW6300 Series Heavy Duty Access Points

For latest support information on various features in Cisco Wave 2 and 802.11ax (Wi-Fi 6) Access Points in Cisco IOS XE releases, see the Feature Matrix for Wave 2 and 802.11ax (Wi-Fi 6) Access Points document.

Configure multiple WGBs (CLI)

Configure multiple WGBs using commands.

Configure multiple WGBs efficiently by uploading and downloading configuration files via SFTP or TFTP.

Perform this procedure on the APs in WGB mode when you need to deploy the same configuration across multiple workgroup bridges or update their settings remotely.

Procedure

1. Enter privileged EXEC mode.

   Example

   ```
   Device# enable
   ```

2. Create upload configuration file and upload to the SFTP or TFTP server.

   Example

   ```
   Device# copy configuration upload {sftp:|tftp:} ip-address [directory]
   [file-name]
   ```

   Example

   ```
   Device# copy configuration upload sftp: 10.10.10.1 C:sample.txt
   ```

3. Download the configuration file and replace the old configuration in the AP.

   Example

   ```
   Device# copy configuration download {sftp:|tftp:} ip-address [directory]
   [file-name]
   ```

   Example

   ```
   Device# copy configuration download sftp: 10.10.10.1 C:sample.txt
   ```

   The device reboots and applies the new configuration when it restarts.

4. Display the WGB uplink information.

   Example

   ```
   Device# show wgb dot11 association
   ```

5. Display the AP software information.

Example

```
Device# show version
```

Multiple WGBs are configured with the uploaded configuration files, and you can verify their status using the show commands.

Verify WGB configuration

Lists the debug log output that confirms successful WGB configuration download and application.

After completing the configuration download and reboot of the AP, the WGB rejoins the network. Use the **show logging** command to list and verify the download events that are captured in the debug logs:

```
Device# show logging

Jan 13 18:19:17 kernel: [*01/13/2022 18:19:17.4880] WGB - Applying download
config...
Jan 13 18:19:18 download_config: configure clock timezone UTC
Jan 13 18:19:18 download_config: configure dot1x credential dot1x_profile
username wifiuser password U2FsdGVkX1+8PWmAOnFO8BXyk5EAphMy2PmhPPhWV0w=
Jan 13 18:19:18 download_config: configure eap-profile eap_profile method PEAP
Jan 13 18:19:18 download_config: configure eap-profile eap_profile
dot1x-credential dot1x_profile
Jan 13 18:19:18 chpasswd: password for user changed
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7260] chpasswd: password for
user changed
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7610]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7610]  Management user
configuration saved successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7610]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7650] Warning!!! Attach SSID
profile with the radio to use the new changes.
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7650]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7650]  Dot1x credential
configuration has been saved successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7650]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7740] Warning!!! Attach SSID
profile with the radio to use the new changes.
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7740]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7740]  EAP profile configuration
 has been saved successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7740]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7790] Warning!!! Attach SSID
profile with the radio to use the new changes.
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7790]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7790]  EAP profile configuration
 has been saved successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7790]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7830] Warning!!! Attach SSID
profile with the radio to use the new changes.
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7830]
Jan 13 18:19:18 download_config: configure ssid-profile psk ssid alpha_psk
authentication psk U2FsdGVkX18meBfFFeiC4sgkEmbGPNH/ul1dne6h/m8= key-management
 wpa2
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7930] Warning!!! Attach SSID
profile with the radio to use the new changes.
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7930]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7930]  EAP profile configuration
```

```
 has been saved successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.7930]
Jan 13 18:19:18 download_config: configure ssid-profile open ssid alpha_open
authentication open
Jan 13 18:19:18 download_config: configure ssid-profile openax ssid
alpha_open_ax authentication open
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.8650]  SSID-Profile dot1xpeap
has been saved successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.8650]
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.9270]  SSID-Profile psk has been
 saved successfully
Jan 13 18:19:18 kernel: [*01/13/2022 18:19:18.9270]
Jan 13 18:19:19 kernel: [*01/13/2022 18:19:19.0380]  SSID-Profile open has
been saved successfully
Jan 13 18:19:19 kernel: [*01/13/2022 18:19:19.0380]
Jan 13 18:19:19 kernel: [*01/13/2022 18:19:19.0380]  SSID-Profile openax has
been saved successfully
Jan 13 18:19:19 kernel: [*01/13/2022 18:19:19.0380]
Jan 13 18:19:22 download_config: configure wgb broadcast tagging disable
Jan 13 18:19:22 download_config: configure wgb packet retries 64 drop
Jan 13 18:19:22 kernel: [*01/13/2022 18:19:22.9710] Broadcast tagging 0
successfully
Jan 13 18:19:22 kernel: [*01/13/2022 18:19:22.9710]
Jan 13 18:19:23 download_config: configure dot11Radio 1 mode wgb ssid-profile
 open
Jan 13 18:19:23 download_config: configure dot11Radio 1 enable
Jan 13 18:19:23 download_config: configure ap address ipv6 disable
```

215 Peer-to-Peer Client Support

Topics:

- Peer-to-peer client support

Introduces peer-to-peer client support functionality and guides through configuration processes while outlining system restrictions and limitations.

Peer-to-peer client support

Describes granular control over how traffic is directed in wireless networks.

Peer-to-peer client support is a WLAN feature that

- provides granular control over how traffic is directed
- can be applied to individual WLANs with each client inheriting the peer-to-peer blocking setting of the WLAN to which it is associated
- allows traffic to be bridged locally within a device, dropped by a device, or forwarded to the upstream VLAN.

Peer-to-peer blocking support details

Peer-to-peer blocking is supported for clients that are associated with local and central switching WLANs.

These restrictions apply to peer-to-peer client support:

- Peer-to-peer blocking does not apply to multicast traffic.
- Peer-to-peer blocking is not enabled by default.
- In FlexConnect, peer-to-peer blocking configuration cannot be applied only to a particular FlexConnect AP or a subset of APs. It is applied to all the FlexConnect APs that broadcast the SSID.
- FlexConnect central switching clients supports peer-to-peer upstream-forward. However, this is not supported in the FlexConnect local switching. This is treated as peer-to-peer drop and client packets are dropped.

  FlexConnect central switching clients supports peer-to-peer blocking for clients associated with different APs. However, for FlexConnect local switching, this solution targets only clients connected to the same AP. FlexConnect ACLs can be used as a workaround for this limitation.

Configure peer-to-peer client support

Configure peer-to-peer client support to control how client traffic flows between devices on the same WLAN.

Enable peer-to-peer blocking policies to control communication between wireless clients on the same WLAN.

Peer-to-peer blocking allows you to control whether wireless clients can communicate directly with each other or must route traffic through the upstream network infrastructure. This feature is useful for security and traffic management in enterprise environments.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter WLAN configuration submode.

   Example

   ```
   Device(config)# wlan profile-name
   ```

   The *profile-name* is the profile name of the configured WLAN.

3. Configure peer-to-peer blocking parameters.

   Example

   ```
   Device(config-wlan)# peer-blocking allow-private-group | drop |
   forward-upstream
   ```

   Example

   ```
   Device(config-wlan)# peer-blocking drop
   ```

   The keywords are:

   - **allow-private-group**: Enables peer-to-peer blocking on the Allow Private Group action.
   - **drop**: Enables peer-to-peer blocking on the drop action.
   - **forward-upstream**: No action is taken and forwards packets to the upstream.

     Note

     The **forward-upstream** option is not supported for Flex local switching. Traffic is dropped even if this option is configured. Also, peer to peer blocking for local switching SSIDs are available only for the clients on the same AP.

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

5. Display the details of the selected WLAN.

   Example

   ```
   Device# show wlan id wlan-id
   ```

   Example

   ```
   Device# show wlan id 12
   ```

Peer-to-peer client support is now configured on the WLAN with the specified blocking behavior for client-to-client communication.

216 Deny Wireless Client Session Establishment Using Calendar Profiles

Topics:

- Denial of wireless client session establishment

Denial of wireless client session establishment

Describes a controller feature that stops client session establishment based on scheduled time periods to enable controlled network management without manual intervention.

Denial of wireless client session establishment is a controller feature that

- allows the controller to stop client session establishment based on a particular time
- helps control the network in efficient and controlled manner without any manual intervention, and
- supports daily, weekly, and monthly recurrence schedules through calendar profiles mapped to policy profiles.

Supported recurrence types

In Cisco Catalyst 9800 Series Wireless Controller , you can deny the wireless client session based on these recurrences:

- Daily
- Weekly
- Monthly

The Calendar Profiles created are then mapped to the policy profile. By attaching the calendar profile to a policy profile, you will be able to create different recurrences for the policy profile using different policy tag.

Note

You need to create separate Calendar Profile for Daily, Weekly, and Monthly sub-categories.

The following is the workflow for denial of wireless client session establishment feature:

- Create a calendar profile.
- Apply the calendar profile to a policy profile.

Note

A maximum of 100 calendar profile configuration and 5 calendar profile association to policy profile is supported.

Configuration limits and timing considerations:

- Feature activation: If you boot up your controller, the denial of client session establishment feature kicks in after a minute from the system boot up.
- Time change adaptation: If you change the system time after the calendar profile is associated to a policy profile, you can expect a maximum of 30 second delay to adapt to the new clock timings.

> **Note**
>
> You cannot use the **no action deny-client** command to disable action while associating the calendar profile to a policy profile.
>
> If you want to disable the action command, you need to disassociate the calendar profile from the policy profile, and re-configure again.

Guidelines and limitations

Guidelines and limitations

-
-

Guidelines and limitations

-
-

Configure daily calendar profile (CLI)

Configure daily calendar profile to set scheduled time periods for wireless operations.

Set up a calendar profile that defines daily recurring time periods for wireless network operations and power management.

Calendar profiles allow you to schedule specific time periods for wireless operations, such as controlling when certain wireless features are active. Daily calendar profiles apply the same time schedule every day.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a calendar profile.

   Example

   ```
   Device(config)# wireless profile calendar-profile name calendar-profile-name
   ```

   Example

   ```
   Device(config)# wireless profile calendar-profile name daily_calendar_profile
   ```

   The *calendar-profile-name* refers to the name of the calendar profile.

3. Configure start and end time for the calendar profile.

Example

```
Device(config-calendar-profile)# start start-time end end-time
```

Example

```
Device(config-calendar-profile)# start 09:00:00 end 17:00:00
```

The *start-time* is the start time for the calendar profile. You need to enter start time in **HH:MM:SS** format.

The *end-time* is the end time for the calendar profile. You need to enter end time in **HH:MM:SS** format.

4. Configure daily recurrences for the calendar profile.

Example

```
Device(config-calendar-profile)# recurrence daily
```

5. Return to privileged EXEC mode.

Example

```
Device(config-calendar-profile)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

> **Note**
>
> When the calendar profile kicks in, the AP power profile rules (for example, radio state and USB device state) that are defined for the Ethernet speed are not applied and continue to be as per the fixed power profile.

The daily calendar profile is configured and will apply the specified time schedule every day for wireless operations.

Configure weekly calendar profile (CLI)

Configure weekly calendar profile to define specific time windows and recurring schedules for network operations.

Create a weekly calendar profile that defines recurring time windows for scheduled network operations or policies.

Calendar profiles allow network administrators to schedule when certain policies or operations should be active. A weekly calendar profile provides recurring time windows that repeat on specified days of the week, enabling automated time-based network management.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a calendar profile.

   Example

   ```
   Device(config)# wireless profile calendar-profile name name
   ```

   Example

   ```
   Device(config)# wireless profile calendar-profile name weekly_calendar_profile
   ```

   The *name* refers to the name of the calendar profile.

3. Configure start and end time for the calendar profile.

   Example

   ```
   Device(config-calendar-profile)# start start-time end end-time
   ```

   Example

   ```
   Device(config-calendar-profile)# start 18:00:00 end 19:00:00
   ```

   The *start-time* is the start time for the calendar profile in **HH:MM:SS** format.

   The *end-time* is the end time for the calendar profile in **HH:MM:SS** format.

4. Configure weekly recurrences for the calendar profile.

   Example

   ```
   Device(config-calendar-profile)# recurrence weekly
   ```

5. Configure days when the weekly calendar needs to be active.

   Example

   ```
   Device(config-calendar-profile)# day {friday | monday | saturday | sunday |
   thursday | tuesday | wednesday}
   ```

   Example

   ```
   Device(config-calendar-profile)# day friday
   Device(config-calendar-profile)# day monday
   ```

   > **Note**
   >
   > You can configure multiple days using this command.

6. Return to privileged EXEC mode.

Example

```
Device(config-calendar-profile)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The weekly calendar profile is now configured and can be applied to network policies that require time-based scheduling and recurring weekly activation windows.

Configure monthly calendar profile (CLI)

Configure monthly calendar profile for scheduling network operations.

Create a monthly calendar profile to schedule recurring network operations or denial of service events on specific dates each month.

Monthly calendar profiles enable you to define time-based policies that recur on specific dates within a month. This is useful for scheduled maintenance windows or automated network policy enforcement.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a calendar profile.

   Example

   ```
   Device(config)# wireless profile calendar-profile name profile-name
   ```

   Example

   ```
   Device(config)# wireless profile calendar-profile name monthly_calendar_profile
   ```

   The *profile-name* refers to the name of the calendar profile.

3. Configure start and end time for the calendar profile.

   Example

   ```
   Device(config-calendar-profile)# start start-time end end-time
   ```

   Example

   ```
   Device(config-calendar-profile)# start 18:00:00 end 19:00:00
   ```

   The *start-time* is the start time for the calendar profile. You need to enter start time in **HH:MM:SS** format.

   The *end-time* is the end time for the calendar profile. You need to enter end time in **HH:MM:SS** format.

4. Configure monthly recurrences for the calendar profile.

Example

```
Device(config-calendar-profile)# recurrence monthly
```

5. Configure a date for the calendar profile.

Example

```
Device(config-calendar-profile)# date date-value
```

Example

```
Device(config-calendar-profile)# date 25
```

> **Note**
>
> If the requirement is to perform denial of service in certain timing, such as, 2,10, and 25 of every month, all three days need to be configured using the **date** command. There is no range for date. You need to configure the dates as per your requirement.

6. Return to privileged EXEC mode.

Example

```
Device(config-calendar-profile)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The monthly calendar profile is configured and can be applied to schedule recurring network operations on the specified dates and times each month.

Map daily calendar profile to policy profile (CLI)

Configure mapping of a daily calendar profile to a policy profile to control client access based on time schedules.

Enable time-based access control by associating a calendar profile with a policy profile to manage client session establishment during specific time intervals.

Calendar profiles define time-based restrictions that can be applied to policy profiles. When mapped together, they allow administrators to automatically deny or allow client connections based on scheduled time slots, providing granular control over network access.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Create policy profile for the WLAN.

   Example

   ```
   Device(config)# wireless profile policy profile-name
   ```

   Example

   ```
   Device(config)# wireless profile policy default-policy-profile
   ```

   The *profile-name* is the profile name of the policy profile.

3. Map a calendar profile to a policy profile.

   Example

   ```
   Device(config-wireless-policy)# calender-profile name calendar-profile-name
   ```

   Example

   ```
   Device(config-wireless-policy)# calender-profile name daily_calendar_profile
   ```

   The *calendar-profile-name* is the name of the calendar profile name created in the daily calendar profile configuration task.

   > **Note**
   >
   > You need to disable Policy Profile before associating a calendar profile to a policy profile. Run this command:
   >
   > ```
   > Device(config-wireless-policy)# shutdown
   > ```

4. Configure deny client session establishment during calendar profile interval.

   Example

   ```
   Device(config-policy-profile-calender)# action deny-client
   ```

   > **Note**
   >
   > Client associations are denied daily between timeslot 9:00:00 to 17:00:00. For start and end time details, see the daily calendar profile configuration task.

5. Return to privileged EXEC mode.

Example

```
Device(config-policy-profile-calender)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The daily calendar profile is successfully mapped to the policy profile, enabling time-based client access control according to the configured schedule.

Map weekly calendar profile to policy profile (CLI)

Configure a weekly calendar profile mapping to a policy profile to control client session establishment during specific time periods.

Control wireless client access by mapping time-based calendar profiles to policy profiles, enabling scheduled denial of client sessions.

Weekly calendar profiles allow administrators to define specific time periods when client associations should be denied. This configuration is useful for implementing access control policies based on business hours or maintenance windows.

Before you begin

Ensure that a weekly calendar profile has been created before mapping it to a policy profile.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create or enter the policy profile configuration mode.

   Example

   ```
   Device(config)# wireless profile policy profile-name
   ```

   Example

   ```
   Device(config)# wireless profile policy default-policy-profile
   ```

3. Map a calendar profile to the policy profile.

   Example

   ```
   Device(config-wireless-policy)# calender-profile name calendar-profile-name
   ```

   Example

   ```
   Device(config-wireless-policy)# calender-profile name weekly_calendar_profile
   ```

> **Note**
>
> You need to disable Policy Profile before associating a calendar profile to a policy profile. Use this command:
>
> ```
> Device(config-wireless-policy)# shutdown
> ```

4. Configure the action to deny client session establishment during calendar profile intervals.

   Example

   ```
   Device(config-policy-profile-calender)# action deny-client
   ```

   > **Note**
   >
   > Client associations are denied daily between timeslot 9:00:00 to 17:00:00. For start and end time details, see the weekly calendar profile configuration task.
   >
   > On Monday and Tuesday, clients are denied between 17:30:00 and 19:00:00 besides regular time 9:00:00 to 17:00:00.
   >
   > On 25th of every month, clients are denied between 18:00:00 and 19:00:00 besides regular time 9:00:00 to 17:00:00.

5. Exit to privileged EXEC mode.

   Example

   ```
   Device(config-policy-profile-calender)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The weekly calendar profile is now mapped to the policy profile, and client session establishment will be denied during the configured time periods defined in the calendar profile.

Map monthly calendar profile to policy profile (CLI)

Configure policy profile calendar mapping to control client access based on monthly calendar schedules.

Control client session establishment by mapping calendar-based time restrictions to wireless policy profiles.

Monthly calendar profiles define time-based access restrictions that can be applied to wireless policy profiles. This allows administrators to automatically deny client associations during specified time periods based on daily, weekly, or monthly schedules.

Before you begin

A monthly calendar profile must be created before it can be mapped to a policy profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create or enter policy profile configuration mode.

   Example

   ```
   Device(config)# wireless profile policy profile-name
   ```

   Example

   ```
   Device(config)# wireless profile policy default-policy-profile
   ```

   The *profile-name* is the profile name of the policy profile.

3. Map a calendar profile to the policy profile.

   Example

   ```
   Device(config-wireless-policy)# calender-profile name calendar-profile-name
   ```

   Example

   ```
   Device(config-wireless-policy)# calender-profile name monthly_calendar_profile
   ```

   The *calendar-profile-name* is the name of the calendar profile name created in the monthly calendar profile configuration task.

4. Configure deny client action for the calendar profile interval.

   Example

   ```
   Device(config-policy-profile-calender)# action deny-client
   ```

   Note

   Every day client associations are denied between timeslot 9:00:00 to 17:00:00. For start and end time details, see the monthly calendar profile configuration task.

   On Monday and Tuesday, clients are denied between 17:30:00 and 19:00:00 besides regular time 9:00:00 to 17:00:00.

   On 25th of every month, clients are denied between 18:00:00 and 19:00:00 besides regular time 9:00:00 to 17:00:00.

5. Return to privileged EXEC mode.

Example

```
Device(config-policy-profile-calender)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

The monthly calendar profile is now mapped to the policy profile and will automatically deny client sessions during the configured time intervals.

Verify calendar profile configuration

Lists commands to verify calendar profile configuration.

Verify calendar profile configuration using show commands to view profile summaries and detailed information for specific profiles.

To view the summary of calendar profiles, use this command:

```
Device# show wireless profile calendar-profile summary
Number of Calendar Profiles: 3

Profile-Name
--------------------------------
monthly_25_profile
weekly_mon_profile
daily_calendar_profile
```

To view the calendar profile details for a specific profile name, use this command:

```
Device# show wireless profile calendar-profile detailed daily_calendar_profile
Calendar profiles                          : daily_calendar_profile
-------------------------------------------------------------------
Recurrence                                 : DAILY
Start Time                                 : 09:00:00
End Time                                   : 17:00:00
```

Verify policy profile configuration

Provides commands to verify policy profile configuration.

Verify policy profile configuration using specific show commands to view detailed parameters and calendar profile information.

To view the detailed parameters for a specific policy profile, use this command:

```
Device# show wireless profile policy detailed default-policy-profile
Tunnel Profile
  Profile Name                     : Not Configured
Calendar Profile
  Profile Name                     : monthly_25_profile
  Wlan Enable                      : Not Configured
  Client Block                     : Client Block Configured
  ------------------------------------------------
  Profile Name                     : weekly_mon_profile
  Wlan Enable                      : Not Configured
  Client Block                     : Client Block Configured
  ------------------------------------------------
```

```
  Profile Name                        : daily_calendar_profile
  Wlan Enable                         : Not Configured
  Client Block                        : Client Block Configured
  ----------------------------------------------------
Fabric Profile
  Profile Name                        : Not Configured
```

To view the configured calendar profile information under policy profile, use this command:

```
Device# show wireless profile policy all
Tunnel Profile
Profile Name : Not Configured
Calendar Profile
Profile Name : daily_calendar_profile
Wlan Enable : Not Configured
Client Block : Client Block Configured
----------------------------------------------------
Profile Name : weekly_calendar_profile
Wlan Enable : Not Configured
Client Block : Client Block Configured
----------------------------------------------------
Fabric Profile
Profile Name : Not Configured
```

Note

The anchor priority is always displayed as local. Priorities can be assigned on the foreign controller.

217 Ethernet over GRE

Topics:

- Ethernet over GRE

Introduces Ethernet over Generic Routing Encapsulation tunnels and guides through configuring tunnel gateways, domains, profiles, and wireless policy associations for secure network connectivity.

Ethernet over GRE

Describes Ethernet over GRE aggregation solution for grouping Wi-Fi traffic from hotspots.

Ethernet over GRE (EoGRE) is an aggregation solution that

- groups Wi-Fi traffic from hotspots
- enables customer premises equipment (CPE) devices to bridge Ethernet traffic coming from an end-host and encapsulate the traffic in Ethernet packets over an IP Generic Routing Encapsulation (GRE) tunnel, and
- forwards end-host traffic and initiates subscriber sessions when IP GRE tunnels are terminated on a service provider's broadband network gateway.

Client IPv6 traffic

Client IPv6 traffic is supported on IPv4 EoGRE tunnels. A maximum of eight different client IPv6 addresses are supported per client. Wireless controllers send all the client IPv6 addresses that they have learned to the accounting server using the accounting update message. All RADIUS or accounting messages exchanged between controllers and tunnel gateways or RADIUS servers are outside the EoGRE tunnel.

To enable EoGRE for a WLAN, the wireless policy profile should be mapped to a tunnel profile, which may contain:

- AAA override: Allows you to bypass rule filtering for a client.
- Gateway RADIUS proxy: Allows forwarding of AAA requests to tunnel gateways.
- Tunnel rules: Defines the domain to use for each realm. They also define VLAN tagging for the client traffic towards tunnel gateways.
- DHCP option 82: Provides a set of predefined fields.

EoGRE deployment with multiple tunnel gateways

The wireless controller sends keepalive pings to the primary and secondary tunnel gateways and keeps track of the missed pings. When a certain threshold level is reached for the missed pings, switchover is performed and the secondary tunnel is marked as active. This switchover deauthenticates all the clients to enable them to rejoin the access points (APs). When the primary tunnel come back online, all the client traffic are reverted to the primary tunnel. However, this behavior depends on the type of redundancy.

Load balancing in EtherChannels

Load balancing of tunneled traffic over Etherchannels works by hashing the source or destination IP addresses or MAC addresses of the tunnel endpoint pair. Because the number of tunnels is very limited when compared to clients (each tunnel carries traffic for many clients), the spreading effect of hashing is highly reduced and optimal utilization of Etherchannel links can be hard to achieve.

Using the EoGRE configuration model, you can use the *tunnel source* option of each tunnel interface to adjust the load-balancing parameters and spread tunnels across multiple links.

You can use different source interfaces on each tunnel for load balancing based on the source or destination IP address. For that choose the source interface IP address in such a way that traffic flows take different links for each src-dest IP pair. This is an example with four ports:

```
Client traffic on Tunnel1 – Src IP: 40.143.0.72  Dest IP: 40.253.0.2
Client traffic on Tunnel2 – Src IP: 40.146.0.94  Dest IP: 40.253.0.6
Client traffic on Tunnel3 – Src IP: 40.147.0.74  Dest IP: 40.253.0.10
```

Use the **show platform software port-channel link-select interface port-channel 4 ipv4** *src_ip dest_ip* command to determine the link that a particular flow will take.

EoGRE configuration

Describes EoGRE configuration methods and deployment options.

EoGRE configuration is a network deployment approach that

- supports two different deployment modes: Central-Switching and Flex or Local-Switching
- uses tunnel gateways, tunnel domains, and tunnel profiles to manage client traffic, and
- enables secure tunneling through policy profiles mapped to WLANs using policy tags.

EoGRE deployment modes

The EoGRE solution can be deployed in two different ways:

- Central-Switching: EoGRE tunnels connect the controller to the tunnel gateways.
- Flex or Local-Switching: EoGRE tunnels are initiated on the APs and terminated on the tunnel gateways.

To configure EoGRE, perform these tasks:

1. Create a set of tunnel gateways.
2. Create a set of tunnel domains.
3. Create a tunnel profile with rules that define how to match clients to domains.
4. Create a policy profile and attach the tunnel profile to it.
5. Map the policy profile to WLANs using policy tags.

> **Note**
>
> The EoGRE tunnel fallback to the secondary tunnel is triggered after the *max-skip-count* ping fails in the last measurement window. Based on the starting and ending instance of the measurement window, the fall-back may take more time than the duration that is configured.

EoGRE supports multiple authentication methods:

Table 191: EoGRE authentication methods

Method Name	First Supported Release	Mode
PSK	17.2.1	Local/Flex (central authentication)
Open	16.12.1	Local/Flex (central authentication)
LWA	16.12.1	Local/Flex (central authentication)
Dot1x	16.12.1	Local/Flex (central authentication)
CWA	16.12.1	Local/Flex (central authentication)

Create a tunnel gateway (CLI)

Configure a tunnel gateway to establish network connectivity between remote sites.

Establish a tunnel gateway for secure communication between network segments.

In the controller, a tunnel gateway is modeled as a tunnel interface.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a tunnel interface and enter interface configuration mode.

   Example

   ```
   Device(config)# interface tunnel tunnel-number
   ```

   Example

   ```
   Device(config)# interface tunnel 21
   ```

3. Set the source address of the tunnel interface.

   Example

   ```
   Device(config-if)# tunnel source source-interface
   ```

   Example

   ```
   Device(config-if)# tunnel source 22
   ```

   The source interface can be VLAN, Gigabit Ethernet or loopback.

4. Set the destination address of the tunnel.

   Example

   ```
   Device(config-if)# tunnel destination tunnel-address
   ```

   Example

   ```
   Device(config-if)# tunnel destination 10.11.12.13
   ```

5. Set the encapsulation mode of the tunnel to Ethernet over GRE IPv4 or Ethernet over GRE IPv6.

   Example

   ```
   Device(config-if)# tunnel mode ethernet gre {ipv4 | ipv6} p2p
   ```

Example

```
Device(config-if)# tunnel mode ethernet gre ipv4 p2p
```

The tunnel gateway is configured and ready to establish secure connections between network segments.

Configure the tunnel gateway (GUI)

Configure the tunnel gateway using GUI.

Configure the tunnel gateway to establish network tunneling connections through the graphical user interface.

Use the GUI to set up tunnel gateway parameters including tunnel identification, destination addressing, source interface selection, and AAA proxy configuration when needed.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **EoGRE**.
2. Click the **Gateways** tab.

   The **Add Gateway** window is displayed.
3. In the **Tunnel ID** field, specify the tunnel ID.
4. In the **Destination address (IPv4/IPv6)** field, specify the IPv4 or IPv6 address.
5. From the **Source Interface** drop-down list, select an interface.
6. In the **AAA Proxy** section, slide the **AAA Proxy** slider to **Enabled**. When AA Proxy is enabled, complete these steps:
   a) From the **ENCRYPTION Type** drop-down list, select either **UNENCRYPTED** or **AES ENCRYPTION**.
   b) In the **Key Phrase** field, specify the key phrase.
7. Click **Apply to Device**.

The tunnel gateway is configured with the specified parameters and applied to the device.

Configure a tunnel domain (CLI)

Configure a tunnel domain.

This task creates a tunnel domain to group tunnels for redundancy purposes, ensuring High Availability by designating primary and secondary tunnels with specific failover behavior.

> **Note**
>
> Tunnel domains are a redundancy grouping of tunnels. This configuration procedure specifies a primary and a secondary tunnel, along with a redundancy model.

At any point of time, only one tunnel will be active per domain. The primary tunnel will be active when tunnel domain is setup. When the primary tunnel goes down (heartbeats are skipped), a switchover to the secondary tunnel is performed.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure EoGRE redundancy domain. **tunnel eogre domain** *domain*

   Example

   ```
   Device(config)# tunnel eogre domain domain
   ```

   ```
   Device(config)# tunnel eogre domain dom1
   ```

3. Configure the primary tunnel.

   Example

   ```
   Device(config-eogre-domain)# primary tunnel primary-tunnel_intf
   ```

   ```
   Device(config-eogre-domain)# primary tunnel 21
   ```

4. Configure the secondary tunnel.

   Example

   ```
   Device(config-eogre-domain)# secondary tunnel secondary-tunnel_intf
   ```

   ```
   Device(config-eogre-domain)# secondary tunnel 22
   ```

5. Set the redundancy model as revertive.

   Example

   ```
   Device(config-eogre-domain)# redundancy revertive
   ```

   When redundancy is set to revertive and the primary tunnel goes down, a switchover to secondary tunnel is performed. When the primary tunnel comes back up, a switchover to the primary tunnel is performed, because the primary tunnel has priority over the secondary tunnel.

   When redundancy is not set to revertive, tunnels will have the same priority, and a switchover to the primary tunnel is not performed if the active tunnel is the secondary tunnel and the primary tunnel comes back up.

The tunnel domain is configured with designated primary and secondary tunnels. The primary tunnel will be active initially, and automatic failover to the secondary tunnel will occur if the primary tunnel becomes unavailable. If revertive redundancy is configured, the system will switch back to the primary tunnel when it becomes available again.

Configure tunnel domain (GUI)

Configure tunnel domain using GUI.

Configure the tunnel domain to establish EoGRE connectivity with primary and secondary tunnel gateways.

Use this procedure to set up tunnel domains through the graphical user interface, including gateway selection and redundancy options.

Follow these steps to configure the tunnel domain:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **EoGRE**.
2. Click the **Domains** tab.

   The **Add Domain** window is displayed.
3. In the **Name** field, specify the domain name.

   The name can be ASCII characters from 32 to 126, without leading and trailing spaces.
4. From the **Primary Tunnel Gateway** drop-down list, choose an option.
5. From the **Secondary Tunnel Gateway** drop-down list, choose an option.
6. Slide the **Status** button to **Enabled**, to activate the domain status.
7. Slide the **Revertive Redundancy** button to **Enabled**, to activate revertive redundancy.
8. Click **Apply to Device**.

The tunnel domain is configured with the specified settings and applied to the device.

Configure EoGRE global parameters CLI)

Configure EoGRE global parameters to establish tunnel communication settings and AAA proxy configuration.

Establish EoGRE tunnel parameters including heartbeat intervals, skip counts, source interfaces, and AAA proxy keys for proper tunnel operation.

EoGRE (Ethernet over Generic Routing Encapsulation) global parameters control tunnel behavior across the network. These settings ensure reliable tunnel communication and proper failover handling.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Set EoGRE tunnel heartbeat periodic interval.

   Example

   ```
   Device(config)# tunnel eogre heartbeat interval interval-value
   ```

   Example

   ```
   Device(config)# tunnel eogre heartbeat interval 600
   ```

3. Set the maximum number of tolerable dropped heartbeats.

   Example

   ```
   Device(config)# tunnel eogre heartbeat max-skip-count skip-count
   ```

   Example

   ```
   Device(config)# tunnel eogre heartbeat max-skip-count 7
   ```

   After reaching the maximum number of heartbeats that can be dropped, the tunnel is declared as down and a switchover is performed.

4. Set the tunnel EoGRE source interface.

   Example

   ```
   Device(config)# tunnel eogre source loopback tunnel-source
   ```

   Example

   ```
   Device(config)# tunnel eogre source loopback 12
   ```

5. Optional: Configure AAA proxy RADIUS key for the AAA proxy setup.

   Example

   ```
   Device(config)# tunnel eogre interface tunnel tunnel-intf aaa proxy key key
   key-name
   ```

   Example

   ```
   Device(config)# tunnel eogre interface tunnel 21 aaa proxy key 0 mykey
   ```

   Note

   When the tunnel gateway is behaving as the AAA proxy server, only this step is required for the configuration.

EoGRE global parameters are configured with the specified heartbeat interval, maximum skip count, source interface, and optional AAA proxy key settings.

Configure EoGRE global parameters (GUI)

Configure EoGRE global parameters using GUI.

The EoGRE global parameters define the network-wide settings for Ethernet over Generic Routing Encapsulation tunnels, including heartbeat intervals and interface assignments.

Use this task to establish the fundamental EoGRE configuration parameters that will apply across your network infrastructure.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **EoGRE**.

   The EoGRE **Global Config** tab is displayed.
2. In the **Heartbeat Interval (seconds)** field, specify an appropriate timer value for heartbeat interval. The valid range is from 60 to 600 seconds.
3. In the **Max Heartbeat Skip Count** field, specify the maximum heartbeat skip count. The valid range is from 3 to 10.
4. From the **Interface Name** drop-down list, choose an interface name.
5. Click **Apply**.

The EoGRE global parameters are configured and applied to your network infrastructure.

Configure a tunnel profile (CLI)

Configure a tunnel profile using commands.

Create and configure a tunnel profile to manage client traffic routing and authentication through tunneled connections.

Tunnel profiles define how client traffic is handled when using tunneled connections, including DHCP options, AAA overrides, and RADIUS proxy settings. This configuration is essential for managing client authentication and traffic routing in wireless environments.

Before you begin

Ensure that you define the destination VLAN on the controller. If you do not define the VLAN, clients will not be able to connect.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Configure a WLAN policy profile.

   Example

   ```
   Device(config)# wireless profile policy profile-policy-name
   ```

   Example

   ```
   Device(config)# wireless profile policy eogre_policy
   ```
3. Create a tunnel profile.

   Example

   ```
   Device(config-wireless-policy)# tunnel-profile tunnel-profile-name
   ```

Example

```
Device(config-wireless-policy)# tunnel-profile tunnel1
```

4. Return to global configuration mode.

   Example

   ```
   Device(config-wireless-policy)# exit
   ```

5. Configure a wireless tunnel profile.

   Example

   ```
   Device(config)# wireless profile tunnel tunnel-profile-name
   ```

   Example

   ```
   Device(config)# wireless profile tunnel wl-tunnel-1
   ```

6. Activate DHCP Option 82 for the tunneled clients.

   Example

   ```
   Device(config-tunnel-profile)# dhcp-opt82 enable
   ```

7. Configure Remote ID options.

   Example

   ```
   Device(config-tunnel-profile)# dhcp-opt82 remote-id remote-id
   ```

   Example

   ```
   Device(config-tunnel-profile)# dhcp-opt82 remote-id vlan
   ```

   Choose from the comma-separated list of options such as **ap-mac**, **ap-ethmac**, **ap-name**, **ap-group-name**, **flex-group-name**, **ap-location**, **VLAN**, **ssid-name**, **ssid-type**, and **client-mac**.

8. Enable AAA policy override.

   Example

   ```
   Device(config-tunnel-profile)# aaa-override
   ```

9. Enable the gateway RADIUS proxy.

   Example

   ```
   Device(config-tunnel-profile)# gateway-radius-proxy
   ```

10. Enable the gateway accounting RADIUS proxy.

Example

```
Device(config-tunnel-profile)# gateway-accounting-radius-proxy
```

11. Create a rule to choose a domain, using the realm filter, for client Network Access Identifier (NAI), tunneling domain name, and destination VLAN.

Example

```
Device(config-tunnel-profile)# rule priority realm-filter realm domain
domain-name vlan vlan-id
```

Example

```
Device(config-tunnel-profile)# rule 12 realm-filter realm domain dom1 vlan 5
```

The tunnel profile is now configured with the specified settings for DHCP options, AAA overrides, RADIUS proxy, and filtering rules, enabling proper client traffic management through tunneled connections.

Configure the tunnel profile (GUI)

Configure a tunnel profile using GUI.

Configure a tunnel profile to define the parameters and settings for establishing tunnel connections in the EoGRE environment.

Use this procedure to create and configure tunnel profiles that manage tunnel connections and their associated settings through the graphical user interface.

Follow these steps to configure the tunnel profile:

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **EoGRE**.
2. Click the **Tunnel Profiles** tab.
3. Click the **Add** button.

   The **Add Tunnel Profile** window is displayed.

4. Click the **General** tab and complete these steps:
   a) In the **Name** field, specify the tunnel profile name. The name can be ASCII characters from 32 to 126, without leading and trailing spaces.
   b) In the **Status** field, slide the button to change the status to **Enabled**.
   c) In the **Central Forwarding** field, slide the button to **Enabled**, to enable the feature.
   d) In the **DHCP Option-82** section, change the **Status** field and the **ASCII** field to **Enabled**, as per requirement.
   e) In the **Delimiter** field, specify the delimiter.
   f) From the **Circuit ID Available Services** list, select an available services and click the > sign to add the services to the assigned list.
   g) From the **Remote ID Available Services** list, select an available services and click the > sign to add the services to the assigned list.
   h) In the **AAA** section, choose an appropriate status for the **Radius Proxy** field, the **Accounting Proxy** field, and the **Override** field.

5. Click the **Rules** tab, and complete these steps:
   a) Click the **Add Rules** button.
   b) In the **Priority** field, specify the priority of the rule from a range of 1 to 100.
   c) In the **Realm** field, specify a realm.
   d) From the **Domain** drop-down list, choose a domain.
   e) In the **VLAN ID** field, specify the VLAN ID that ranges from 1 to 4094.
   f) Click **Save**.
6. Click **Apply to Device**.

The tunnel profile is successfully configured and applied to the device with the specified settings for tunnel connections.

Associate WLAN to a wireless policy profile (CLI)

Configure the association of a WLAN to a wireless policy profile.

Associate a WLAN with a wireless policy profile to apply specific policies and configurations to wireless network traffic.

When managing wireless networks, you need to associate WLANs with policy profiles to ensure proper traffic handling and security configurations are applied to wireless clients.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a policy tag and enter policy tag configuration mode.

   Example

   ```
   Device(config)# wireless tag policy policy-tag-name
   ```

   Example

   ```
   Device(config)# wireless tag policy eogre_tag
   ```

3. Map an EoGRE policy profile to a WLAN profile.

   Example

   ```
   Device(config-policy-tag)# wlan wlan-name policy profile-policy-name
   ```

   Example

   ```
   Device(config-policy-tag)# wlan eogre_open_eogre policy eogre_policy
   ```

4. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

Example

```
Device(config-policy-tag)# end
```

The WLAN is now associated with the specified wireless policy profile, and the configured policies will be applied to wireless traffic on that WLAN.

Attach a policy tag and a site tag to an AP

Configure an access point with policy and site tags to define its network behavior and location-based settings.

Associate policy and site tags with an AP to apply specific network policies and location-based configurations.

Policy tags define network behaviors and WLAN mappings for APs, while site tags specify location-based settings like DNS servers and local switching parameters. Both tags must be attached to APs to ensure proper network operation.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an AP and enter AP profile configuration mode.

   Example

   ```
   Device(config)# ap mac-address
   ```

   Example

   ```
   Device(config)# ap 80E8.6FD4.0BB0
   ```

3. Map the policy tag to the AP.

   Example

   ```
   Device(config-ap-tag)# policy-tag policy-tag-name
   ```

   Example

   ```
   Device(config-ap-tag)# policy-tag eogre_tag
   ```

4. Map a site tag to the AP.

   Example

   ```
   Device(config-ap-tag)# site-tag site-tag-name
   ```

Example

```
Device(config-ap-tag)# site-tag sp-flex-site
```

5. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

Example

```
Device(config-ap-tag)# end
```

The AP is now configured with the specified policy tag and site tag, enabling it to apply the associated network policies and location-based settings.

Verify EoGRE tunnel configuration

Verify the EoGRE tunnel configuration.

The `show tunnel eogre` command displays the EoGRE clients, domains, gateways, global-configuration, and manager information in the local mode.

To display the EoGRE domain summary in the local mode, use this command:

```
Device# show tunnel eogre domain summary
Domain Name          Primary GW   Secondary GW      Active GW      Redundancy
-------------------------------------------------------------------------------
domain1             Tunnel1      Tunnel2           Tunnel1        Non-Revertive
eogre_domain        Tunnel1      Tunnel2           Tunnel1        Non-Revertive
```

To display the details of an EoGRE domain in the local mode, use this command:

```
Device# show tunnel eogre domain detailed domain-name
 Domain Name    : eogre_domain
 Primary GW     : Tunnel1
 Secondary GW   : Tunnel2
 Active GW      : Tunnel1
 Redundancy     : Non-Revertive
```

To view the EoGRE tunnel gateway summary and statistics in the local mode, use this command:

```
Device# show tunnel eogre gateway summary
Name               Type  Address                                    AdminState
 State  Clients
-------------------------------------------------------------------------------------
Tunnel1            IPv4  9.51.1.11                                  Up
 Up     0
Tunnel2            IPv4  9.51.1.12                                  Up
 Down   0
Tunnel10           IPv6  fd09:9:8:21::90                            Down
 Down   0
Tunnel11           IPv4  9.51.1.11                                  Up
 Up     0
Tunnel12           IPv6  fd09:9:8:21::90                            Up
 Down   0
Tunnel100          IPv4  9.51.1.100                                 Up
```

```
 Down   0
```

To view the details of an EoGRE tunnel gateway in the local mode, use this command:

```
Device# show tunnel eogre gateway detailed gateway-name
Gateway : Tunnel1
 Mode   : IPv4
 IP     : 9.51.1.11
 Source : Vlan51 / 9.51.1.1
 State  : Up
 SLA ID : 56
 MTU    : 1480
 Up Time: 4 minutes 45 seconds
 Clients
  Total Number of Wireless Clients      : 0
 Traffic
  Total Number of Received Packets      : 0
  Total Number of Received Bytes        : 0
  Total Number of Transmitted Packets   : 0
  Total Number of Transmitted Bytes     : 0
 Keepalives
  Total Number of Lost Keepalives       : 0
  Total Number of Received Keepalives   : 5
  Total Number of Transmitted Keepalives: 5
  Windows                               : 1
  Transmitted Keepalives in last window : 2
  Received Keepalives in last window    : 2
```

To view the client summary of EoGRE in the local mode, use this command:

```
Device# show tunnel eogre client summary
Client MAC      AP MAC          Domain                                Tunnel
   VLAN  Local
-------------------------------------------------------------------------------
74da.3828.88b0  80e8.6fd4.9520  eogre_domain                          N/A
   2121   No
```

To view the details of an EoGRE global configuration in the local mode, use this command:

```
Device# show tunnel eogre global-configuration
Heartbeat interval       : 60
Max Heartbeat skip count : 3
Source Interface         : (none)
```

To view the details of the global tunnel manager statistics in the local mode, use this command:

```
Device# show tunnel eogre manager stats global
Tunnel Global Statistics
Last Updated                    : 02/18/2019 23:50:35
EoGRE Objects
  Gateways                      : 6
  Domains                       : 2
EoGRE Flex Objects
  AP Gateways                   : 2
  AP Domains                    : 1
  AP Gateways HA inconsistencies : 0
  AP Domains HA inconsistencies  : 0
```

```
Config events
  IOS Tunnel updates              : 806
  IOS Domain updates              : 88
  Global updates                  : 48
  Tunnel Profile updates          : 120
  Tunnel Rule updates             : 16
  AAA proxy key updates           : 0
AP events
  Flex AP Join                    : 1
  Flex AP Leave                   : 0
  Local AP Join                   : 0
  Local AP leave                  : 0
  Tunnel status (rx)              : 4
  Domain status (rx)              : 1
  IAPP stats msg (rx)             : 3
  Client count (rx)               : 6
  VAP Payload msg (tx)            : 4
  Domain config (tx)              : 1
  Global config (tx)              : 1
  Client delete (tx)              : 1
  Client delete per domain (tx)   : 3
  DHCP option 82 (tx)             : 4
Client events
  Add-mobile                      : 2
  Run-State                       : 3
  Delete                          : 1
  Cleanup                         : 0
  Join                            : 2
  Plumb                           : 0
  Join Errors                     : 0
  HandOff                         : 0
  MsPayload                       : 2
  FT Recover                      : 0
  Zombie GW counter increase      : 0
  Zombie GW counter decrease      : 0
  Tunnel Profile reset            : 88
  Client deauth                   : 0
  HA reconciliation               : 0
Client Join Events
  Generic Error                   : 0
  MSPayload Fail                  : 0
  Invalid VLAN                    : 0
  Invalid Domain                  : 0
  No GWs in Domain                : 0
  Domain Shut                     : 0
  Invalid GWs                     : 0
  GWs Down                        : 0
  Rule Match Error                : 0
  AAA-override                    : 0
  Flex No Active GW               : 0
  Open Auth join attempt          : 2
  Dot1x join attempt              : 2
  Mobility join attempt           : 0
  Tunnel Profile not valid        : 2
  Tunnel Profile valid            : 2
  No rule match                   : 0
  Rule match                      : 2
  AAA proxy                       : 0
  AAA proxy accounting            : 0
  AAA eogre attributes            : 0
  Has aaa override                : 0
  Error in handoff payload        : 0
```

```
  Handoff AAA override           : 0
  Handoff no AAA override        : 0
  Handoff payload received       : 0
  Handoff payload sent           : 0
SNMP Traps
  Client                         : 0
  Tunnel                         : 2
  Domain                         : 0
IPC
  IOSd TX messages               : 0
Zombie Client
  Entries                        : 0
```

To view the tunnel manager statistics of a specific process instance in the local mode, use this command:

```
Device# show tunnel eogre manager stats instance instance-number
Tunnel Manager statistics for process instance : 0
Last Updated                     : 02/18/2019 23:50:35
EoGRE Objects
  Gateways                       : 6
  Domains                        : 2
EoGRE Flex Objects
  AP Gateways                    : 2
  AP Domains                     : 1
  AP Gateways HA inconsistencies : 0
  AP Domains HA inconsistencies  : 0
Config events
  IOS Tunnel updates             : 102
  IOS Domain updates             : 11
  Global updates                 : 6
  Tunnel Profile updates         : 15
  Tunnel Rule updates            : 2
  AAA proxy key updates          : 0
AP events
  Flex AP Join                   : 1
  Flex AP Leave                  : 0
  Local AP Join                  : 0
  Local AP leave                 : 0
  Tunnel status (rx)             : 4
  Domain status (rx)             : 1
  IAPP stats msg (rx)            : 3
  Client count (rx)              : 6
  VAP Payload msg (tx)           : 4
  Domain config (tx)             : 1
  Global config (tx)             : 1
  Client delete (tx)             : 1
  Client delete per domain (tx)  : 3
  DHCP option 82 (tx)            : 4
Client events
  Add-mobile                     : 2
  Run-State                      : 3
  Delete                         : 1
  Cleanup                        : 0
  Join                           : 2
  Plumb                          : 0
  Join Errors                    : 0
  HandOff                        : 0
  MsPayload                      : 2
  FT Recover                     : 0
  Zombie GW counter increase     : 0
  Zombie GW counter decrease     : 0
```

```
  Tunnel Profile reset           : 11
  Client deauth                  : 0
  HA reconciliation              : 0
Client Join Events
  Generic Error                  : 0
  MSPayload Fail                 : 0
  Invalid VLAN                   : 0
  Invalid Domain                 : 0
  No GWs in Domain               : 0
  Domain Shut                    : 0
  Invalid GWs                    : 0
  GWs Down                       : 0
  Rule Match Error               : 0
  AAA-override                   : 0
  Flex No Active GW              : 0
  Open Auth join attempt         : 2
  Dot1x join attempt             : 2
  Mobility join attempt          : 0
  Tunnel Profile not valid       : 2
  Tunnel Profile valid           : 2
  No rule match                  : 0
  Rule match                     : 2
  AAA proxy                      : 0
  AAA proxy accounting           : 0
  AAA eogre attributes           : 0
  Has aaa override               : 0
  Error in handoff payload       : 0
  Handoff AAA override           : 0
  Handoff no AAA override        : 0
  Handoff payload received       : 0
  Handoff payload sent           : 0
SNMP Traps
  Client                         : 0
  Tunnel                         : 2
  Domain                         : 0
IPC
  IOSd TX messages               : 0
Zombie Client
  Entries                        : 0
```

The `show ap tunnel eogre` command displays the tunnel domain information, EoGRE events, and the tunnel gateway status on the APs, in the flex mode.

To view the summary information of an EoGRE tunnel gateway in the flex mode, use this command:

```
Device# show ap tunnel eogre domain summary
AP MAC          Domain                              Active Gateway
-------------------------------------------------------------------------------
80e8.6fd4.9520  eogre_domain                        Tunnel1
```

To view the wireless tunnel profile summary, use this command:

```
Device# show wireless profile tunnel summary
Profile Name                     AAA-Override AAA-Proxy DHCP Opt82 Enabled
-------------------------------- ------------ --------- ---------- --------
eogre_tunnel                               No        No        Yes      Yes
eogre_tunnel_set                           No        No        Yes       No
eogre_tunnel_snmp                          No        No         No       No
```

To view a wireless tunnel profile's details, use this command:

```
Device# show wireless profile tunnel detailed profile-name
Profile Name : eogre_tunnel
Status : Enabled
 AAA-Proxy/Accounting-Proxy: Disabled / Disabled
 AAA-Override : Disabled
 DHCP Option82 : Enabled
  Circuit-ID : ap-mac,ap-ethmac,ap-location,vlan
  Remote-ID : ssid-name,ssid-type,client-mac,ap-name
Tunnel Rules
Priority Realm                Vlan Domain (Status/Primary GW/Secondary GW)
-------- -------------------- ---- ---------------------------------------------
1        *                    2121 eogre_domain (Enabled/Tunnel1/Tunnel2)
```

To view detailed information about an EoGRE tunnel domain's status, use this command:

```
Device# show ap tunnel eogre domain detailed
Domain     : eogre_domain
AP MAC     : 80e8.6fd4.9520
Active GW  : Tunnel1
```

To view the EoGRE events on an AP, use this command:

```
Device# show ap tunnel eogre events
AP 80e8.6fd4.9520  Event history
Timestamp               #Times   Event                RC Context
----------------------- -------- -------------------- --
----------------------------------------
02/18/2019 23:50:26.341 6        IAPP_STATS            0 GW Tunnel2 uptime:0s

02/18/2019 23:49:40.222 2        CLIENT_JOIN           0 74da.3828.88b0,
(eogre_domain/2121)
02/18/2019 23:48:43.549 1        CLIENT_LEAVE          0 74da.3828.88b0,
(eogre_domain/2121)
02/18/2019 23:47:33.127 1        DOMAIN_STATUS         0 eogre_domain Active
GW: Tunnel1
02/18/2019 23:47:33.124 4        AP_TUNNEL_STATUS      0 Tunnel2 Dn

02/18/2019 23:47:33.124 1        MSG_CLIENT_DEL        0 GW Tunnel2 (IP:
9.51.1.12)
02/18/2019 23:47:33.124 2        TUNNEL_ADD            0 GW Tunnel2

02/18/2019 23:47:33.120 3        MSG_CLIENT_DEL_PD     0 GW Tunnel1 (IP:
9.51.1.11)
02/18/2019 23:47:31.763 2       AP_DOMAIN_PUSH       0 Delete:eogre_domain_set,
 0 GWs
02/18/2019 23:47:31.753 4       AP_VAP_PUSH          0 profile:'eogre_tunnel',
 wlan:pyats_eogre
```

To view the summary information of the EoGRE tunnel gateway, use this command:

```
Device# show ap tunnel eogre gateway summary
AP MAC          Gateway          Type  IP
  State  Clients
-------------------------------------------------------------------------------------
80e8.6fd4.9520  Tunnel1          IPv4  9.51.1.11
  Up     1
80e8.6fd4.9520  Tunnel2          IPv4  9.51.1.12
```

```
  Down   0
```

To view detailed information about an EoGRE tunnel gateway, use this command:

```
Device# show ap tunnel eogre gateway detailed gateway-name
Gateway : Tunnel1
 Mode   : IPv4
 IP     : 9.51.1.11
 State  : Up
 MTU    : 1476
 Up Time: 14 hours 25 minutes 2 seconds
 AP MAC : 80e8.6fd4.9520
 Clients
  Total Number of Wireless Clients      : 1
 Traffic
  Total Number of Received Packets      : 6
  Total Number of Received Bytes        : 2643
  Total Number of Transmitted Packets   : 94
  Total Number of Transmitted Bytes     : 20629
  Total Number of Lost Keepalive        : 3
```

To view summary information about the EoGRE tunnel gateway status, use this command:

```
Device# show ap tunnel eogre domain summary
AP MAC          Domain                          Active Gateway
-------------------------------------------------------------------------------
80e8.6fd4.9520  eogre_domain                    Tunnel1
```

To view information about EoGRE events on an AP, use this command:

```
Device# show ap name ap-name tunnel eogre events
AP 80e8.6fd4.9520  Event history
Timestamp               #Times   Event               RC Context
----------------------- -------- ------------------- --
----------------------------------------
02/18/2019 23:50:26.341 6        IAPP_STATS           0 GW Tunnel2 uptime:0s

02/18/2019 23:49:40.222 2        CLIENT_JOIN          0 74da.3828.88b0,
(eogre_domain/2121)
02/18/2019 23:48:43.549 1        CLIENT_LEAVE         0 74da.3828.88b0,
(eogre_domain/2121)
02/18/2019 23:47:33.127 1        DOMAIN_STATUS        0 eogre_domain Active
GW: Tunnel1
02/18/2019 23:47:33.124 4        AP_TUNNEL_STATUS     0 Tunnel2 Dn

02/18/2019 23:47:33.124 1        MSG_CLIENT_DEL       0 GW Tunnel2 (IP:
9.51.1.12)
02/18/2019 23:47:33.124 2        TUNNEL_ADD           0 GW Tunnel2

02/18/2019 23:47:33.120 3        MSG_CLIENT_DEL_PD    0 GW Tunnel1 (IP:
9.51.1.11)
02/18/2019 23:47:31.763 2       AP_DOMAIN_PUSH      0 Delete:eogre_domain_set,
 0 GWs
02/18/2019 23:47:31.753 4       AP_VAP_PUSH         0 profile:'eogre_tunnel',
 wlan:pyats_eogre
```

To view the summary information about EoGRE tunnel domain's status on an AP, use this command:

```
Device# show ap name ap-name tunnel eogre domain summary
AP MAC          Domain                          Active Gateway
-------------------------------------------------------------------------------
80e8.6fd4.9520  eogre_domain
```

To view the detailed information about EoGRE tunnel domain on an AP, use this command:

```
Device# show ap name ap-name tunnel eogre domain detailed
Domain Name    : eogre_domain
 Primary GW    : Tunnel1
 Secondary GW  : Tunnel2
 Active GW     : Tunnel1
 Redundancy    : Non-Revertive
 AdminState    : Up
```

To view the summary information about EoGRE tunnel gateways on an AP, use this command:

```
Device# show ap name ap-name tunnel eogre gateway summary
AP MAC          Gateway          Type  IP
  State  Clients
-------------------------------------------------------------------------------------------
80e8.6fd4.9520  Tunnel1          IPv4  9.51.1.11
  Up     1
80e8.6fd4.9520  Tunnel2          IPv4  9.51.1.12
  Down   0
```

To view detailed information about an EoGRE tunnel gateway's status on an AP, use this command:

```
Device# show ap name ap-name tunnel eogre gateway detailed gateway-name
Gateway : Tunnel2
 Mode   : IPv4
 IP     : 9.51.1.12
 State  : Down
 MTU    : 0
 AP MAC : 80e8.6fd4.9520
 Clients
  Total Number of Wireless Clients      : 0
 Traffic
  Total Number of Received Packets      : 0
  Total Number of Received Bytes        : 0
  Total Number of Transmitted Packets   : 0
  Total Number of Transmitted Bytes     : 0
  Total Number of Lost Keepalive        : 151
```

218 Wireless Guest Access

Topics:

- Wireless guest access
- Configure guest access using different security methods

Guides the configuration and management of wireless guest access, including mobility tunnels, load balancing, and guest anchor setups for secure visitor connectivity.

Wireless guest access

Explains the Wireless Guest Access feature that provides internet access to guests in a secure and accountable manner.

Wireless guest access is a network security feature that

- provides internet access to guests in a secure and accountable manner
- uses the enterprise's existing wireless and wired infrastructure to the maximum extent, and
- reduces the cost and complexity of building a physical overlay network.

Wireless guest access architecture and components

The Wireless Guest Access solution comprises of two controllers - a Guest Foreign and a Guest Anchor. An administrator can limit bandwidth and shape the guest traffic to avoid impacting the performance of the internal network.

Note

- When a client joins through a capwap tunnel from an AP, the RADIUS NAS-Port-Type is set as "wireless 802.11". Here, Point of Attachment (PoA) and Point of Presence (PoP) is the same.
- When a client joins through a mobility tunnel, the RADIUS NAS-Port-Type is set as "virtual". Here, PoA is the Foreign controller and PoP is the Anchor controller as the client is anchored. For information on the standard types, see the following link:

  https://www.iana.org/assignments/radius-types/radius-types.xhtml#radius-types-13

Wireless Guest Access feature comprises these functions:

- Guest Anchor controller is the point of presence for a client.
- Guest Anchor Controller provides internal security by forwarding the traffic from a guest client to a Cisco Wireless Controller in the demilitarized zone (DMZ) network through the anchor controller.
- Guest Foreign controller is the point of attachment of the client.
- Guest Foreign Controller is a dedicated guest WLAN or SSID and is implemented throughout the campus wireless network wherever guest access is required. A WLAN with mobility anchor (guest controller) configured on it identifies the guest WLAN.
- Guest traffic segregation implements Layer 2 or Layer 3 techniques across the campus network to restrict the locations where guests are allowed.
- Guest user-level QoS is used for rate limiting and shaping, although it is widely implemented to restrict the bandwidth usage for a guest user.
- Access control involves using embedded access control functionality within the campus network, or implementing an external platform to control guest access to the Internet from the enterprise network.
- Authentication and authorization of guests that are based on variables, including date, duration, and bandwidth.
- An audit mechanism to track who is currently using, or has used, the network.
- A wider coverage is provided by including areas such as lobbies and other common areas that are otherwise not wired for network connectivity.
- The need for designated guest access areas or rooms is removed.

> **Note**
>
> To use IRCM with AireOS in your network, contact Cisco TAC for assistance.

This table shows controller support for guest access functions.

Table 192: Supported controllers

Controller Name	Supported as Guest Anchor	Supported as Guest Foreign
Cisco Catalyst 9800-40 Wireless Controller	Yes	Yes
Cisco Catalyst 9800-80 Wireless Controller	Yes	Yes
Cisco Catalyst 9800-CL Wireless Controller	Yes	Yes
Cisco Catalyst 9800-L Wireless Controller	Yes	Yes
Cisco Catalyst 9800 Embedded Wireless Controller for Switch	No	No
Cisco Catalyst 9800 Embedded Wireless Controller on Cisco Catalyst 9100 Series APs	No	No

This is a list of features supported by Cisco Guest Access:

- Sleeping Clients
- FQDN
- AVC (AP upstream and downstream)
- Native Profiling
- Open Authentication
- OpenDNS
- Supported Security Methods:
  - MAB Central Web Authentication (CWA)
  - Local Web Authentication (LWA)
  - LWA on MAB Failure
  - 802.1x + CWA
  - 802.1x
  - PSK
  - 802.1x + LWA
  - PSK + CWA

- PSK + LWA
- iPSK + CWA
- MAB Failure + PSK
- MAB Failure + OWE
- MAB Failure + SAE

- SSID QoS Upstream and Downstream (Foreign)
- AP/ Client SSO
- Static IP Roaming
- Client IPv6
- Roaming across controllers
- RADIUS Accounting

> **Note**
>
> In a guest access scenario, accounting is always performed at the foreign controller for all authentication methods.

- QoS: Client-Level Rate Limiting
- Guest Anchor Load Balancing
- Workgroup Bridges (WGB)

> **Note**
>
> To enable the controller to support multiple VLANs from a WGB, use **wgb vlan** command.

Foreign map

Explains the foreign map feature that supports guest access using policy profile and WLAN profile configuration models.

A foreign map is a guest access feature that

- supports guest access using Policy Profile and WLAN Profile configuration models in the controller
- is achieved with policy profile and WLAN profile config model, and
- configures two different WLAN profiles on two Guest Foreigns where seamless roaming is not allowed between them.

Foreign map configuration

Foreign Map support in Cisco Catalyst 9800 Series Wireless Controller is achieved with the policy profile and WLAN profile configuration model.

Foreign map commands

- Guest Foreign commands:
  - **Foreign1: wlanProf1 PolicyProf1**

- **Foreign2: wlanProf2 PolicyProf2**

- Guest Anchor commands:
  - **wlanProf1, wlanProf2**
  - **PolicyProf1: Vlan100 - subnet1**
  - **PolicyProf2: Vlan200 - subnet2**

Foreign map roaming

Configure two different WLAN profiles on the two Guest Foreigns and seamless roaming is not allowed between them. This is expected configuration. However, seamless roaming is allowed if the same WLAN profile is configured on two Guest Foreigns, but it prevents Foreign Map feature from working.

Wireless Guest Access: Use Cases

The wireless guest access feature can be used to meet different requirements. Some of the possibilities are shared here.

Scenario One: Providing Secured Network Access During Company Merger

This feature can be configured to provide employees of **company A** who are visiting **company B** to access company A resources on company B network securely.

Scenario Two: Shared Services over Existing Setup

Using this feature, you can provide multiple services using multiple vendors piggy backing on the existing network. A company can provide services on an SSID which is anchored on the existing controller. This is while the existing service continues to serve over the same controller and network.

Guidelines for wireless guest access

Outlines guidelines and limitations for wireless guest access.

Match the security profiles under WLAN on both Guest Foreign, and Guest Anchor.

- Match the policy profile attributes such as NAC and AAA Override on both Guest Foreign, and Guest Anchor controllers.
- On Export Anchor, the WLAN profile name and Policy profile name is chosen when a client joins at runtime and the same should match with the Guest Foreign controller.

Recommendation: troubleshooting IPv6

Consider these workarounds when a guest export client cannot get a routable IPv6 address through SLAAC or cannot pass traffic when the IPv6 address is learned through DHCPv6.

When a guest export client cannot get a routable IPv6 address through SLAAC or cannot pass traffic when the IPv6 address is learned through DHCPv6, you can use these workarounds:

- On IPv6 Routers: You can work around the RA multicast to unicast conversion by modifying behavior on the IPv6 gateway. Depending on the product, this may be the default behavior or may require configuration.
- On Cisco IPv6 Routers: Configure unicast RA depending on your platform.
- On non-Cisco IPv6 Routers: If non-Cisco network devices do not support configuration command to enable solicited unicast RA then a work around does not exist.

For Cisco IPv6 Routers:

- Cisco Nexus platform: Has solicited unicast RA enabled by default to help with wireless deployment.
- Cisco IOS-XE platform: Use the following configuration command to turn on unicast RA to help with wireless deployment:

  ipv6 nd ra solicited unicast

Load balancing among multiple guest controllers

Describes load balancing among multiple guest controllers.

Load balancing among multiple guest controllers is a network configuration feature that

- distributes large guest client volumes across up to 72 controllers for a single export foreign guest WLAN configuration
- supports priority-based anchor configuration with primary anchors (priority 1,3) and backup anchors for failure scenarios, and
- automatically handles failover by disconnecting clients from failed primary anchors and redirecting them to secondary anchors.

Configuration and failover behavior

To configure mobility guest controllers, use **mobility anchor** *ip address*.

You can specify primary anchors with priority (1,3) and choose another anchor as backup in case of failure.

In a multi-anchor scenario, when the primary anchor goes down, the clients get disconnected from the primary anchor and joins the secondary anchor.

When the highest priority anchor goes down before the keep-alive timeout completes, the export anchor request for new clients fails, and Foreign selects the next highest priority Anchor in the list to export the clients.

Configure mobility tunnel for guest access (GUI)

Configure mobility tunnel for guest access using GUI.

Enable secure guest network access by establishing a mobility tunnel that allows traffic to traverse between network segments while maintaining security policies.

Use this procedure when setting up guest network access that requires mobility tunneling to route guest traffic through designated mobility anchors in your wireless infrastructure.

Procedure

1. Choose **Configure** > **Tags and Profiles** > **WLANs**.
2. In the **Wireless Networks** area, click the relevant WLAN or RLAN and click **Mobility Anchor**.
3. In the **Wireless Network Details** section, choose a device from the **Switch IP Address** drop-down list.
4. Click **Apply**.

The mobility tunnel for guest access is configured and the selected switch is designated as the mobility anchor for the wireless network.

Configure mobility tunnel for guest access (CLI)

Configure mobility tunnel for guest access using commands.

Configure a mobility tunnel to enable guest access across mobility groups.

Follow the procedure given below to configure a mobility tunnel. This allows guest clients to roam between access points in different mobility groups while maintaining their network session.

Procedure

1. Configure a mobility group.

   Example

   ```
   Device(config)# wireless mobility group name group-name
   ```

   Example

   ```
   Device(config)# wireless mobility group name mtunnelgrp
   ```

2. Configure a mobility MAC address.

   Example

   ```
   Device(config)# wireless mobility mac-address mac-address
   ```

   Example

   ```
   Device(config)# wireless mobility mac-address 0d:4c:da:3a:f2:21
   ```

3. Configure a mobility peer.

   Example

   ```
   Device(config)# wireless mobility group member mac-address mac-address ip
   ip-address group group-name
   ```

   Example

   ```
   Device(config)# wireless mobility group member mac-address df:07:a1:a7:a8:55
    ip 206.223.123.2 group mtgrp
   ```

The mobility tunnel is now configured, allowing guest clients to seamlessly roam between controllers in the mobility group.

Configure guest access policy (GUI)

Configure guest access policy using GUI.

Configure guest access policies to manage how guest users connect to and use your wireless network.

Guest access policies define the network access parameters and security settings for temporary users who need network connectivity without full user credentials.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.

2. Click **Add**.
3. In the **General** tab, enter the **Name** and enable the **Central Switching** toggle button.
4. In the **Access Policies** tab, under the **VLAN** settings, choose the vlans from the **VLAN/VLAN Group** drop-down list.
5. In the **Mobility** tab, under the **Mobility Anchors** settings, check the **Export Anchor** check box.
6. In the **Advanced** tab, under the **WLAN Timeout** settings, enter the **Idle Timeout (sec)**.
7. Click **Apply to Device**.

The guest access policy is configured and applied to the device, allowing guest users to access the network according to the defined parameters.

Configure guest access policy (CLI)

Configure guest access policy using commands.

Create and configure a guest access profile policy to enable guest networking on wireless clients.

Follow the procedure given below to create and configure the guest access profile policy. Alternately, you may use the existing default policy profile after configuring the mobility anchor to that policy.

You can only configure anchors which are peers. Ensure that the IP address that is used is a mobility peer and is included in the mobility group. The system shows an invalid anchor IP address error message when any other IP address is used.

To delete the mobility group, ensure that the mobility peer which is also a mobility anchor is removed from the policy profile.

Note

- No payload is sent to Guest Foreign to display the VLAN.
- To avoid a client exclusion from occurring due to VLAN, Cisco Catalyst 9800 Series Controllers need to define VLAN along with the associated name being pushed from ISE.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the policy profile and enter wireless profile configuration mode.

   Example

   ```
   Device(config)# wireless profile policy wlan-policy-profile
   ```

   Example

   ```
   Device(config)# wireless profile policy guest-test-policy
   ```

> **Note**
>
> - You can use the **default-policy-profile** to configure the profile policy.

3. Shut down the policy if it exists before configuring the anchor.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

4. Optional: Enable central switching.

   Example

   ```
   Device(config-wireless-policy)# central switching
   ```

5. Configure Guest Foreign or Guest Anchor.
   - Configure Guest Foreign with anchor IP address
   - Configure Guest Anchor

   Example

   For Guest Foreign:

   ```
   Device(config-wireless-policy)# mobility anchor anchor-ip-address
   ```

   For Guest Anchor:

   ```
   Device(config-wireless-policy)# mobility anchor
   ```

   Example

   For Guest Foreign:

   ```
   Device(config-wireless-policy)# mobility anchor 19.0.2.1
   ```

   For Guest Anchor:

   ```
   Device(config-wireless-policy)# mobility anchor
   ```

6. Optional: Configure duration of idle timeout, in seconds.

   Example

   ```
   Device (config-wireless-policy)# idle-timeout timeout
   ```

   Example

   ```
   Device (config-wireless-policy)# idle-timeout 1000
   ```

7. Configure VLAN name or VLAN ID.

   Example

   ```
   Device(config-wireless-policy)# vlan vlan-id
   ```

   Example

   ```
   Device(config-wireless-policy)# vlan 2
   ```

   > **Note**
   >
   > VLAN is optional for a Guest Foreign controller.

8. Enable policy profile.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

9. Exit the configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

10. Optional: Display the configured profiles.

    Example

    ```
    Device# show wireless profile policy summary
    ```

11. Optional: Display detailed information of a policy profile.

    Example

    ```
    Device# show wireless profile policy detailed policy-profile-name
    ```

    Example

    ```
    Device# show wireless profile policy detailed guest-test-policy
    ```

The guest access policy is configured and enabled, allowing guest clients to connect using the specified mobility anchor and VLAN settings.

View guest access debug information (CLI)

Lists commands to view guest access debug information.

View guest access debug information using commands.

- To display client level detailed information about mobility state and the anchor IP address, use this command:

  show wireless client mac-add *mac-address* **detail**

- To display the client mobility statistics, use this command:

  show wireless client mac-address *mac-address* **mobility statistics**

- To display client level roam history for an active client in sub-domain, use this command:

  show wireless client mac-address *mac-address* **mobility history**

- To display detailed parameters of a given profile policy, use this command:

  show wireless profile policy detailed *policy-name*

- To display the global level summary for all mobility messages, use this command:

  show wireless mobility summary

- To display the statistics for the Mobility manager, use this command:

  show wireless stats mobility

Verify wireless guest access enablement

Provides verification that wireless guest access is enabled.

To check if wireless guest access is enabled, run this command.

```
Device# show platform hardware chassis  active  qfp feature sw client vlan all

------------------------------------------------------------
Vlan : 666
Learning Enabled : true
DHCPSN Enabled : true
Non IP Multicast Enabled : false
Broadcast Enabled : false
Wireless Passive Client Enabled : false
Guest-Lan Enabled : true
MTU : 65535
Input UIDB : 65503
Output UIDB : 65497
Flood List : 0XB8658A0
```

Configure guest access using different security methods

Describes how to configure guest access using various security methods and provides detailed information about different configuration options and implementation approaches.

These sections provide information about:

Open authentication

Describes how to configure guest access using open authentication by setting up the WLAN profile, with no tag requirements unless Application Visibility and Control (AVC) is enabled.

To configure the guest access with open authentication, follow the steps:

1. Configure the WLAN Profile

2. Configure access policy profile

> **Note**
>
> No tag is required unless AVC is enabled.

Configure a WLAN profile for guest access with open authentication (GUI)

Configure a WLAN profile that enables guest access using open authentication through the graphical user interface.

Create a WLAN profile that allows guest users to connect to the wireless network without requiring authentication credentials, providing open access for temporary or visitor use.

Guest access with open authentication is commonly used in public areas, visitor networks, or temporary access scenarios where security requirements are minimal and ease of access is prioritized.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**, the **SSID** and the **WLAN ID**. Choose the radio policy from the **Radio Policy** drop-down list. Enable or disable the **Status** and **Broadcast SSID** toggle buttons.
4. Choose **Security** > **Layer2** tab. Uncheck the **WPA Policy**, **WPA2 Policy**, **AES** and **802.1x** check boxes.
5. Click **Apply to Device**.

The WLAN profile is created and configured for guest access with open authentication. Guest users can now connect to the wireless network without providing authentication credentials.

Configure a WLAN profile for guest access with open authentication (CLI)

Configure a WLAN profile with open authentication to provide unrestricted network access for guest users.

Establish a guest access WLAN that allows users to connect without authentication credentials.

Open authentication WLANs are commonly used for guest networks where ease of access is prioritized over security. This configuration removes all security barriers, making it suitable for public or temporary access scenarios.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN and SSID.

   Example

   ```
   Device(config)# wlan profile-name wlan-id ssid-name
   ```

Example

```
Device(config)# wlan mywlan 34 mywlan-ssid
```

3. Disable WPA security.

Example

```
Device(config-wlan)# no security wpa
```

4. Disable security AKM for dot1x.

Example

```
Device(config-wlan)# no security wpa akm dot1x
```

5. Disable WPA2 security.

Example

```
Device(config-wlan)# no security wpa wpa2
```

6. Disable WPA2 ciphers for AES.

Example

```
Device(config-wlan)# no security wpa wpa2 ciphers aes
```

7. Save the configuration.

Example

```
Device(config-wlan)# no shutdown
```

The WLAN profile is now configured with open authentication, allowing guest users to connect without any security credentials.

Configure a policy profile (CLI)

Configure a WLAN policy profile to define network connectivity and mobility settings for wireless clients.

Create a policy profile that establishes WLAN connectivity rules and mobility configurations for wireless network access.

Policy profiles define how wireless clients connect to and interact with the network, including VLAN assignments, mobility settings, and switching behavior. Configure these profiles to control client access and network segmentation.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

Example

```
Device(config)# wireless profile policy wlan-policy-profile
```

Example

```
Device(config)# wireless profile policy open_it
```

3. Configure Guest Foreign or Guest Anchor. Choose the first option to configure a Guest Foreign or second option to configure a Guest Anchor:
   - **mobility anchor** *anchor-ip-address*
   - **mobility anchor**

Example

For Guest Foreign:

```
Device (config-wireless-policy)# mobility anchor anchor-ip-address
```

For Guest Anchor:

```
Device (config-wireless-policy)# mobility anchor
```

Example

For Guest Foreign:

```
Device (config-wireless-policy)# mobility anchor 19.0.2.1
```

For Guest Anchor:

```
Device (config-wireless-policy)# mobility anchor
```

4. Enable Central switching.

Example

```
Device(config-wireless-policy)# central switching
```

5. Configure a VLAN name or VLAN ID.

Example

```
Device(config-wireless-policy)# vlan id
```

Example

```
Device(config-wireless-policy)# vlan 16
```

> **Note**
>
> VLAN is optional for a Guest Foreign controller.

6. Enable the policy profile.

Example

```
Device(config-wireless-policy)# no shutdown
```

The policy profile is now configured and enabled, ready to be applied to WLAN configurations for wireless client connectivity.

Local web authentication

Describes the configuration steps for Local Web Authentication (LWA), including setting up parameter maps, WLAN profiles, policy profiles, and AAA servers to enable web-based authentication for wireless clients.

To configure LWA, follow these steps:

1. Configure the Parameter Map.
2. Configure the WLAN Profile.
3. Applying Policy Profile on a WLAN
4. Configure the AAA Server.

Configure a parameter map (GUI)

Configure a parameter map for web authentication through the graphical user interface.

Configure a parameter map to define web authentication settings including connection limits and timeout values.

Parameter maps are used to configure web authentication parameters that control how users authenticate through the web interface. This configuration is performed through the device's graphical user interface.

Procedure

---

1. Choose **Configuration** > **Security** > **Web Auth**.
2. Click **Add**.
3. Enter the **Parameter-map name**, **Maximum HTTP connections**,**Init-State Timeout(secs)** and choose **webauth** in the **Type** drop-down list.
4. Click **Apply to Device**.

The parameter map is configured and applied to the device with the specified web authentication settings.

Configure a parameter map (CLI)

Configure a parameter map to define web authentication settings and timeout parameters for client authentication.

Define web authentication parameters and timeout settings to control client authentication behavior.

Parameter maps are used to configure global web authentication settings that control how clients authenticate and how long they remain in various authentication states.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a parameter map and enter parameter-map WEBAUTH configuration mode.

   Example

   ```
   Device(config)# parameter-map type webauth global
   ```

3. Configure the WEBAUTH type parameter.

   Example

   ```
   Device(config-params-parameter-map)# type webauth
   ```

4. Configure the WEBAUTH timeout in seconds.

   Example

   ```
   Device(config-params-parameter-map)# timeout init-state sec timeout-seconds
   ```

   Example

   ```
   Device(config-params-parameter-map)# timeout init-state sec 3600
   ```

   Valid range for time is from 60 to 3932100 seconds.

5. Configure a virtual IP address.

   Example

   ```
   Device(config-params-parameter-map)# virtual-ip ipv4 virtual-IP-address
   ```

   Example

   ```
   Device(config-params-parameter-map)# virtual-ip ipv4 209.165.201.1
   ```

The parameter map is configured with web authentication settings, timeout values, and virtual IP address for client authentication.

Configure a WLAN profile for guest access with local web authentication (GUI)

Configure a WLAN profile to enable guest access using local web authentication through the GUI interface.

Configure a WLAN profile to enable guest access with local web authentication, allowing temporary network access for visitors while maintaining security controls.

Use this procedure when you need to provide internet access to guests through a web-based authentication portal. This configuration enables controlled access for users who do not have permanent network credentials.

Follow these steps to configure a WLAN profile for guest access with local web authentication:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click on the **WLAN** name.
3. Choose **Security** > **Layer3**.
4. Check the **Web Policy** check box.
5. Choose a parameter map from the **Web Auth Parameter Map** drop-down list.
6. Choose an authentication list from the **Authentication List** drop-down list.
7. Click **Update & Apply to Device**.

The WLAN profile is configured with local web authentication for guest access. Guest users connecting to this WLAN will be redirected to a web authentication portal where they can obtain network access.

Configure a WLAN profile for guest access with local web AUTHENTICATION (CLI)

Configure local web AUTHENTICATION for a WLAN profile to enable guest access through the command-line interface.

This task configures a WLAN profile with local web AUTHENTICATION to provide secure guest access to the wireless network.

Local web AUTHENTICATION allows guest users to access the network through a web-based AUTHENTICATION portal. This configuration is typically used in guest access scenarios where users need to authenticate via a web interface before gaining network access.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN and SSID.

   Example

   ```
   Device# Device(config)# wlan mywlan 38 mywlan-ssid1
   ```

3. Enable web authentication for a WLAN.

Example

```
Device(config-wlan)# security web-auth
```

4. Configure the default parameter map.

Example

```
Device(config-wlan)# security web-auth parameter-map default
```

> **Note**
>
> When **security web-auth** is enabled, you get to map the **default authentication-list** and global **parameter-map**. This is applicable for authentication-list and parameter-map that are not explicitly mentioned.

5. Configure the global parameter map.

Example

```
Device(config-wlan)# security web-auth parameter-map global
```

6. Set the authentication list for IEEE 802.1x.

Example

```
Device(config-wlan)# security web-auth authentication-list lwa-authentication
```

The WLAN profile is configured with local web AUTHENTICATION for guest access. Guest users will now be prompted to authenticate through a web interface when connecting to the wireless network.

Configure an AAA server for local web authentication (GUI)

Configure AAA server settings for local web authentication through the GUI interface.

This task allows you to set up AAA server configuration for local web authentication using the graphical user interface.

Use this procedure when you need to configure authentication and authorization settings for local web authentication through the device's web interface.

Procedure

---

1. Choose **Configuration** > **Security** > **AAA** > **AAA Advanced** > **Global Config**.
2. Choose the options from the **Local Authentication**, **Authentication Method List**, **Local Authorization** and **Authorization Method List** drop-down lists.
3. Enable or Disable the **Radius Server Load Balance** using toggle button.
4. Check the **Interim Update** check box.
5. Click **Apply**.

The AAA server configuration for local web authentication is now configured with your selected settings.

Configure an AAA server for local web authentication (CLI)

Configure an AAA server for local web authentication using CLI commands.

Configure Authentication, Authorization, and Accounting (AAA) server settings to enable local web authentication on the device.

Use this procedure when you need to set up local authentication and authorization for web-based user access on your device.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. **aaa authentication login** *lwa-authentication* **local**

   Example

   ```
   Device(config)#aaa authentication login lwa-authentication local
   ```

   Defines the authentication method at login.

3. **aaa authorization network default local if-authenticated**

   Example

   ```
   Device(config)#aaa authorization network default local if-authenticated
   ```

   Sets the authorization method to local if the user has authenticated.

The AAA server is configured for local web authentication. Users can now authenticate locally when accessing the device through web interface.

Configure global settings (CLI)

Configure global system settings including user authentication and HTTP server parameters.

Establish basic global configuration settings for system access and HTTP server functionality.

Global configuration sets system-wide parameters that affect device operation and user access methods.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Set the clear text password for the user.

Example

```
Device(config)# username name password 0 clear-text-password
```

Example

```
Device(config)# username base password 0 pass1
```

3. Enable the HTTP server.

Example

```
Device(config)# ip http server
```

4. Set the HTTP server authentication method to local.

Example

```
Device(config)# ip http authentication local
```

> **Note**
>
> You will get the admin access rights regardless of the user privilege, if the **ip http authentication local** is disabled and username is the same as enable password.

Global configuration settings are now applied, enabling HTTP server access with local authentication and user password configuration.

Central web authentication

Describes a security method that redirects unauthenticated users to a centralized web portal for credential verification before granting network access, commonly used in guest networks and BYOD environments.

Configure WLAN profile for guest access with central web authentication (GUI)

Configure a WLAN profile for guest access with central web authentication using the GUI interface.

Configure a WLAN profile that enables guest access using central web authentication to provide secure network access for guest users.

Guest access with central web authentication allows visitors to access the network through a web-based authentication portal while maintaining security controls through MAC filtering and authorization lists.

Before you begin

Follow these steps to configure a WLAN profile for guest access with central web authentication:

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.

2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**, the **SSID**, and the **WLAN ID**.
4. To enable the WLAN, set **Status** as **Enabled**.
5. From the **Radio Policy** drop-down list, select the radio policy.
6. To enable the **Broadcast SSID**, set the status as **Enabled**.
7. Choose **Security** > **Layer2** tab. Uncheck the **WPA Policy**, **WPA2 Policy**, **AES** and **802.1x** check boxes.
8. Check the **MAC Filtering** check box to enable the feature. With MAC Filtering enabled, choose the Authorization list from the **Authorization List** drop-down list.
9. Click **Apply to Device**.

The WLAN profile is configured for guest access with central web authentication. The profile is now available for guest users to connect through the web authentication portal.

Configure a WLAN profile for guest access with central web authentication (CLI)

Configure a WLAN profile to enable guest access using central web authentication for secure network access.

Set up a WLAN profile that allows guest users to access the network through central web authentication, providing controlled access while maintaining security.

Guest access with central web authentication is used when you need to provide network access to temporary users while maintaining control and security through web-based authentication portals.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN and SSID.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id ssid-name
   ```

   Example

   ```
   Device(config)# wlan mywlan 38 mywlan-ssid1
   ```

3. Enable MAB authentication for the remote RADIUS server.

   Example

   ```
   Device(config-wlan)# mac-filtering remote-authorization-list-name
   ```

   Example

   ```
   Device(config-wlan)# mac-filtering auth-list
   ```

4. Disable WPA security.

   Example

   ```
   Device(config-wlan)# no security wpa
   ```

5. Disable security AKM for dot1x.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

6. Disable WPA2 security.

   Example

   ```
   Device(config-wlan)# no security wpa wpa2
   ```

7. Disable WPA2 ciphers for AES.

   Example

   ```
   Device(config-wlan)# no security wpa wpa2 ciphers aes
   ```

8. Save the configuration and activate the WLAN.

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

The WLAN profile is configured for guest access with central web authentication, allowing guest users to connect and authenticate through the web portal.

Configure AAA server (GUI)

Configure RADIUS server groups and servers for AAA authentication using the GUI interface.

Configure RADIUS server groups and individual RADIUS servers to enable AAA authentication for network access control and user authentication.

AAA (Authentication, Authorization, and Accounting) servers provide centralized authentication services for network devices. RADIUS servers must be properly configured with server groups to ensure reliable authentication services.

Procedure

---

1. Choose **Configuration** > **Security** > **AAA** > **Servers/Groups** > **RADIUS** > **Server Groups**.
2. Click the RADIUS server group.
3. From the **MAC-Delimiter** drop-down list, choose an option.
4. From the **MAC-Filtering** drop-down list, choose an option.
5. Enter the **Dead-Time (mins)**.
6. From the **Available Servers** on the left, move the servers you need to **Assigned Servers** on the right.

7. Click **Update & Apply to Device**.
8. Choose **Configuration** > **Security** > **AAA** > **Servers/Groups** > **RADIUS** > **Servers**.
9. Click the RADIUS server.
10. Enter the **IPv4/IPv6 Server Address**, **Auth Port**, **Acct Port**, **Server Timeout (seconds)** and **Retry Count**.
11. Check or uncheck the **PAC Key** checkbox and choose the Key Type from the **Key Type** drop-down list. Enter the **Key** and **Confirm Key**.
12. Enable or disable the **Support for CoA** toggle button.
13. Click **Update & Apply to Device**.

The RADIUS server groups and servers are configured and applied to the device. AAA authentication is now available using the configured RADIUS servers.

Configure AAA server (CLI)

Configure AAA server settings using the command line interface for authentication and authorization services.

Set up AAA server configuration to enable authentication and authorization for network access control.

Configure AAA server for Guest Foreign only. This configuration establishes RADIUS server groups and authorization methods for network access control.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Set the authorization method to local.

   Example

   ```
   Device(config)# aaa authorization network authorization-list local group
   server-group-name
   ```

   Example

   ```
   Device(config)# aaa authorization network cwa local group ise
   ```

3. Configure RADIUS server group definition.

   Example

   ```
   Device(config)# aaa group server radius server-group-name
   ```

   Example

   ```
   Device(config)# aaa group server radius ise
   ```

> **Note**
>
> *server-group-name* refers to the server group name. The valid range is from 1 to 32 alphanumeric characters.

4. Configure the RADIUS server name.

   Example

   ```
   Device(config-sg-radius)# server name radius-server-name
   ```

   Example

   ```
   Device(config-sg-radius)# server name ise1
   ```

5. Set the MAC address as the password.

   Example

   ```
   Device(config-sg-radius)# subscriber mac-filtering security-mode mac
   ```

6. Set the MAC address delimiter to colon.

   Example

   ```
   Device(config-sg-radius)# mac-delimiter colon
   ```

7. Save the configuration, exit configuration mode, and return to privileged EXEC mode.

   Example

   ```
   Device(config-sg-radius)# end
   ```

8. Set the RADIUS server name.

   Example

   ```
   Device(config)# radius server name
   ```

   Example

   ```
   Device(config)# radius server ISE1
   ```

9. Configure the RADIUS server IP address authentication and accounting ports.

   Example

   ```
   Device(config-radius-server)# address ipv4 radius-server-ipaddress auth-port
    port-number acct-port port-number
   ```

Example

```
Device(config-radius-server)# address ipv4 209.165.201.1 auth-port 1635
acct-port 33
```

The AAA server configuration is now complete with RADIUS server group settings, authentication methods, and MAC filtering parameters established for Guest Foreign access control.

Configure 802.1x with local web authentication

Configure 802.1x authentication combined with local web authentication for enhanced WLAN security.

Configure dual authentication mechanism using 802.1x and local web authentication for comprehensive WLAN security.

This configuration combines 802.1x network access control with local web authentication, providing layered security for wireless clients. This is typically used when both network-level and user-level authentication are required.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN and SSID.

   Example

   ```
   Device(config)# wlan wlan-profile wlan-id ssid
   ```

   Example

   ```
   Device(config)# wlan testwprofile 22 ssid-3
   ```

3. Configure 802.1X for the WLAN.

   Example

   ```
   Device(config-wlan)# security dot1x authentication-list
   authentication-list-name
   ```

   Example

   ```
   Device(config-wlan)# security dot1x authentication-list default
   ```

4. Enable authentication list for web authentication security on the WLAN.

   Example

   ```
   Device(config-wlan)# security web-auth authentication-list
   authentication-list-name
   ```

Example

```
Device(config-wlan)# security web-auth authentication-list default
```

5. Configure the global parameter map for web authentication.

Example

```
Device(config-wlan)# security web-auth parameter-map global
```

6. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

The WLAN is configured with both 802.1x and local web authentication, providing dual-layer security for wireless clients.

Configure local web authentication with PSK protocol (CLI)

Configure local web authentication with PSK protocol to enable secure access to the wireless network through web-based authentication combined with pre-shared key security.

Enable secure wireless network access using both PSK encryption and web-based authentication for enhanced security.

Local web authentication with PSK protocol combines pre-shared key security with web-based user authentication, providing a dual-layer security approach for wireless networks where traditional enterprise authentication methods are not required.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the WLAN and SSID.

Example

```
Device(config)# wlan wlan-profile wlan-id ssid
```

Example

```
Device(config)# wlan psksec-profile 22 ssid-4
```

3. Disable WPA security.

Example

```
Device(config-wlan)# no security wpa
```

4. Disable WPA2 security.

Example

```
Device(config-wlan)# no security wpa wpa2
```

5. Disable security AKM for dot1x.

Example

```
Device(config-wlan)# no security wpa akm dot1x
```

6. Enable the security type as PSK.

Example

```
Device(config-wlan)# security wpa akm psk
```

7. Configure the PSK shared key.

Example

```
Device(config-wlan)# security wpa akm psk set-key {ascii | hex} key
```

Example

```
Device(config-wlan)# security wpa akm psk set-key asci 0
```

8. Enable the web authentication for the WLAN.

Example

```
Device(config-wlan)# security web-auth
```

9. Enable authentication list for the WLAN.

Example

```
Device(config-wlan)# security web-auth authentication-list default
```

10. Configure the global parameter map.

Example

```
Device(config-wlan)# security web-auth parameter-map global
```

The WLAN is now configured with local web authentication using PSK protocol, requiring users to authenticate through a web portal while maintaining PSK encryption for wireless traffic.

Central web authentication with PSK protocol

Describes the configuration process for Central Web Authentication using Pre-Shared Key protocol, including steps to configure WLAN profiles and apply policy profiles to wireless networks.

To configure the CWA with PSK security protocol, follow the steps:

1. Configure the WLAN Profile.
2. Applying Policy Profile on a WLAN

Configure WLAN profile for central web authentication with PSK protocol (CLI)

Configure a WLAN profile to support central web authentication using pre-shared key (PSK) protocol for wireless clients.

Set up central web authentication with PSK to provide secure wireless access with web-based client authentication.

Central web authentication with PSK allows wireless clients to connect using a shared key and then authenticate through a web portal. This configuration is useful in guest networks or environments requiring web-based user authentication while maintaining PSK connectivity.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN and SSID.

   Example

   ```
   Device(config)# wlan wlan-profile wlan-id ssid
   ```

   Example

   ```
   Device(config)# wlan cwasec-profile 27 ssid-5
   ```

3. Disable WPA security.

   Example

   ```
   Device(config-wlan)# no security wpa
   ```

4. Disable WPA2 security.

   Example

   ```
   Device(config-wlan)# no security wpa wpa2
   ```

5. Disable security AKM for dot1x.

Example

```
Device(config-wlan)# no security wpa akm dot1x
```

6. Enable the security type as PSK.

Example

```
Device(config-wlan)# security wpa psk
```

7. Configure the PSK shared key.

Example

```
Device(config-wlan)# security wpa psk set-key {ascii | hex} key
```

Example

```
Device(config-wlan)# security wpa psk set-key asci 0
```

8. Enable MAC filtering for PSK web authentication.

Example

```
Device(config-wlan)# mac-filtering authorization-list-name
```

Example

```
Device(config-wlan)# mac-filtering cwa-list
```

The WLAN profile is configured for central web authentication with PSK protocol, allowing clients to connect using the shared key and authenticate via web portal.

Central web authentication with iPSK protocol

Describes how to configure Central Web Authentication (CWA) with individual Pre-Shared Key (iPSK) protocol by setting up the WLAN profile and applying the appropriate policy profile to the wireless LAN.

To configure the CWA with iPSK security protocol, follow the steps:

1. Configure the WLAN Profile.

Configure WLAN profile for central web authentication with iPSK protocol

Configure a WLAN profile that uses central web authentication with iPSK protocol for secure guest access.

This task configures a WLAN profile that enables central web authentication with iPSK (Identity Pre-Shared Key) protocol, providing secure guest access while maintaining centralized authentication control.

Use this configuration when you need to provide guest wireless access with pre-shared key authentication combined with central web authentication. This approach allows for individualized keys per device while maintaining centralized authentication oversight.

Procedure

1. Configure guest WLAN.

   Example

   ```
   config# wlan guest-wlan-name wlan-id ssid
   ```

   ```
   config# wlan ipsk-cwa-profile 28 ssid-6
   ```

2. Disable security AKM for 802.1x.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

3. Configure the PSK AKM shared key.

   Example

   ```
   Device(config-wlan)# security wpa akm psk set-key {asci | hex} key
   ```

   ```
   Device(config-wlan)# security wpa akm psk set-key asci 0
   ```

4. Enable MAC filtering for iPSK authentication.

   Example

   ```
   Device(config-wlan)# mac-filtering authorization_list_name
   ```

   ```
   Device(config-wlan)# mac-filtering cwa-list
   ```

The WLAN profile is configured for central web authentication with iPSK protocol. Guest devices can now connect using pre-shared keys while being subject to centralized web authentication control and MAC filtering.

Configure web authentication on MAC address bypass failure (GUI)

Configure Web Authentication to handle MAC address bypass failures through the WLAN profile settings.

This task configures Web Authentication to handle situations where MAC address filtering fails, providing an alternative authentication method for network access.

When MAC filtering is enabled but a device's MAC address is not in the authorization list, you can configure the system to fall back to Web Authentication instead of denying access completely.

Procedure

1. Click **Configuration** > **Tags and Profiles** > **WLANs**.
2. Click **Add** to add a new WLAN Profile or click the one you want to edit.
3. In the **Edit WLAN** window, complete the following steps:

a) Choose **Security** > **Layer2** and check the **MAC Filtering** check box to enable MAC filtering.
b) From the **Authorization List** drop-down list, select a value.
c) Choose the **Layer3** tab.
d) Click **Show Advanced Settings** and check the **On MAC Filter Failure** checkbox.

Web Authentication is now configured to activate when MAC address filtering fails, allowing devices not on the authorization list to authenticate through the web interface.

Configure web authentication on MAC address bypass failure (CLI)

Configure authentication to fall back to web authentication if a client cannot authenticate using MAC filter while trying to connect to a WLAN.

Enable web authentication fallback when MAC filter authentication fails to avoid client disassociations due to MAC filter authentication failures.

You can configure authentication to fall back to web authentication, if a client cannot authenticate using MAC filter (Local or RADIUS), while trying to connect to a WLAN. To enable this feature, configure both MAC filtering and Web Authentication on the device. This can also avoid disassociations that happen only because of MAC filter authentication failure.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure WLAN policy profile and enter the wireless policy configuration mode.

   Example

   ```
   Device(config)# wireless profile policy policy-name
   ```

   Example

   ```
   Device(config)# wireless profile policy cwa
   ```

3. Enable Central switching.

   Example

   ```
   Device(config-wireless-policy)# central switching
   ```

4. Configure Guest Foreign or Guest Anchor.
   - mobility anchor *anchor-ip-address*
   - mobility anchor

Example

For Guests Foreign:

```
Device (config-wireless-policy)# mobility anchor anchor-ip-address
```

For Guest Anchor:

```
Device (config-wireless-policy)# mobility anchor
```

Example

For Guests Foreign:

```
Device (config-wireless-policy)# mobility anchor 19.0.2.1
```

For Guest Anchor:

```
Device (config-wireless-policy)# mobility anchor
```

5. Configure a VLAN name or VLAN ID.

Example

```
Device(config-wireless-policy)# vlan name
```

Example

```
Device(config-wireless-policy)# vlan 16
```

> **Note**
>
> VLAN is optional for a Guest Foreign controller.

6. Enable the policy profile.

Example

```
Device(config-wireless-policy)# no shutdown
```

7. Configure guest WLAN.

Example

```
config# wlan guest-wlan-name wlan-id ssid
```

Example

```
config# wlan test-wlan-guest 10 wlan-ssid
```

8. Configure MAC filtering support on WLAN.

Example

```
config-wlan# mac-filtering mac-auth-listname authorization-override
```

9. Enable web authentication.

   Example

   ```
   config-wlan# security web-auth
   ```

10. Enable web authentication if MAC filter authentication fails.

    Example

    ```
    config-wlan# security web-auth on-macfilter-failure
    ```

Web authentication fallback is now configured for MAC filter authentication failures, preventing client disconnections when MAC authentication fails.

Configure WLAN for web authentication on MAC authentication bypass failure with pre-shared key (CLI)

Enable web authentication for clients who fail MAC authentication bypass on a WLAN, using a pre-shared key, through the CLI.

Use this configuration to ensure clients who do not pass MAC authentication bypass are redirected to a web authentication portal with a secure pre-shared key.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN.

   Example

   ```
   Device(config)# wlan wlan-name wland-id SSID-name
   ```

   - **wlan-name** is the name of the configured WLAN.
   - **wlan-id** is the WLAN identifier. The range is one to 512.
   - **SSID-name** is the SSID name which can have up to 32 alphanumeric characters.

   If you have already created and configured the WLAN, use the **wlan** *wlan-name* command.

3. Set the MAC filtering parameters.

Example

```
Device(config-wlan)# mac-filtering test-auth-list
```

4. Configure the PSK AKM shared key using the command.

Example

```
Device(config-wlan)# ecurity wpa psk set-key {ascii | hex} key password
```

```
Device(config-wlan)# security wpa psk set-key ascii 0 PASSWORD
```

5. Disable the security AKM for 802.1X.

Example

```
Device(config-wlan)# no security wpa akm dot1x
```

6. Configure PSK support.

Example

```
Device(config-wlan)# security wpa akm psk
```

7. Enable the authentication list for 802.1X security.

Example

```
Device(config-wlan)# security web-auth authentication-list
authenticate-list-name
```

8. Enable the authorization list for 802.1X security.

Example

```
Device(config-wlan)# security web-auth authorization-list authorize-list-name
```

9. Enable web authentication on MAC filter failure.

Example

```
Device(config-wlan)# security web-auth on-macfilter-failure
```

10. Configure the parameter map.

Example

```
Device(config-wlan)# security web-auth parameter-map parameter-map-name
```

> **Note**
>
> If a parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.

11. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

The WLAN is configured to perform web authentication for clients failing MAC authentication bypass, securing access with a pre-shared key.

Configure WLAN with web authentication for MAB failures using a pre-shared key (CLI)

Enable web authentication for clients who fail MAC authentication bypass (MAB) on a WLAN, using a pre-shared key through the CLI.

Use this configuration to ensure that clients who do not pass MAC authentication bypass are redirected to a web authentication portal protected by a secure pre-shared key.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the WLAN.

Example

```
Device(config)# wlan wlan-name wland-id SSID-name
```

- **wlan-name** is the name of the configured WLAN.
- **wlan-id** is the WLAN identifier. The range is one to 512.
- **SSID-name** is the SSID name which can have up to 32 alphanumeric characters.

If you have already created and configured the WLAN, use the **wlan** *wlan-name* command.

3. Set the MAC filtering parameters.

Example

```
Device(config-wlan)# mac-filtering auth-list-name
```

4. Disable security AKM for dot1x.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

5. Enable WPA3 support.

   Example

   ```
   Device(config-wlan)# security wpa wpa3
   ```

6. Enable WPA3 OWE support.

   Example

   ```
   Device(config-wlan)# security wpa akm owe
   ```

7. Enable authentication list for 802.1X security.

   Example

   ```
   Device(config-wlan)# security web-auth authentication-list
   authenticate-list-name
   ```

8. Enable authorization list for 802.1X security.

   Example

   ```
   Device(config-wlan)# security web-auth authorization-list authorize-list-name
   ```

9. Enable web authentication on MAC filter failure.

   Example

   ```
   Device(config-wlan)# security web-auth on-macfilter-failure
   ```

10. Configure the parameter map.

    Example

    ```
    Device(config-wlan)# security web-auth parameter-map parameter-map-name
    ```

    If a parameter map is not associated with a WLAN, the system uses the configuration from the global parameter map.

11. Enable the WLAN.

    Example

    ```
    Device(config-wlan)# no shutdown
    ```

The WLAN now performs web authentication for clients who fail MAC authentication bypass and secures access with a pre-shared key.

Configure WLAN for web authentication on MAB failure with SAE (CLI)

Configure a WLAN using the CLI. If MAC Authentication Bypass (MAB) fails, users are directed to web authentication with Simultaneous Authentication of Equals (SAE) enabled.

Use this task when you require a WLAN that supports MAC authentication. If MAC authentication fails, the controller falls back to web authentication with SAE.

This setup strengthens security by ensuring that only authorized users can access the network. It also supports modern authentication protocols.

Before you begin

Prepare the names of the authentication and authorization lists and the parameter map, if needed.

Procedure

1. Configure the WLAN.

   Example

   ```
   Device(config)# wlan wlan-name wland-id SSID-name
   ```

   - **wlan-name** is the name of the configured WLAN.
   - **wlan-id** is the WLAN identifier. The range is one to 512.
   - **SSID-name** is the SSID name which can have up to 32 alphanumeric characters.

   If you have already created and configured the WLAN, use the **wlan** *wlan-name* command.

2. Set the MAC filtering parameters.

   Example

   ```
   Device(config-wlan)# mac-filtering test-auth-list
   ```

3. Disable security AKM for 802.1X.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

4. Enable WPA3 support.

   Example

   ```
   Device(config-wlan)# security wpa wpa3
   ```

5. Enable AKM SAE support.

Example

```
Device(config-wlan)# security wpa akm sae
```

6. Enable authentication list for 802.1X security.

   Example

   ```
   Device(config-wlan)# security web-auth authentication-list
   authenticate-list-name
   ```

7. Enable authorization list for 802.1X security.

   Example

   ```
   Device(config-wlan)# security web-auth authorization-list authorize-list-name
   ```

8. Enable web authentication on MAC filter failure.

   Example

   ```
   Device(config-wlan)# security web-auth on-macfilter-failure
   ```

9. Configure the parameter map.

   Example

   ```
   Device(config-wlan)# security web-auth parameter-map parameter-map-name
   ```

   If a parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.

The WLAN is configured to use MAC authentication. If MAC authentication fails, users are redirected to web authentication, with WPA3 and SAE enabled.

Configure WLAN for web authentication on MAC authentication failure with Dot1x (CLI)

Configure a WLAN to use web authentication as a fallback mechanism when MAC filtering fails, while maintaining 802.1x as the primary authentication method.

Enable web authentication fallback for clients that fail MAC filtering on a WLAN configured with 802.1x authentication.

This configuration allows clients to authenticate via a web portal when they cannot pass MAC filtering, providing a secondary authentication path while maintaining 802.1x as the primary security mechanism.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create or enter WLAN configuration submode.

Example

```
Device(config)# wlan profile-name wlan-id ssid-name
```

Example

```
Device(config)# wlan wlan-test 3 ssid-test
```

- *profile-name*: Profile name of the configured WLAN.
- *WLAN-id*: Wireless LAN identifier. Range is from 1 to 512.
- *SSID_Name*: SSID that can contain 32 alphanumeric characters.

> **Note**
>
> If you have already configured a WLAN, enter the profile name of the configured WLAN in the command (**WLAN** *profile-name*) and continue with the rest of the configuration steps.

3. Configure MAC filtering parameters.

Example

```
Device(config-wlan)# mac-filtering auth-list-name
```

Example

```
Device(config-wlan)# mac-filtering test-auth-list
```

4. Configure 802.1x authentication.

Example

```
Device(config-wlan)# security dot1x authentication-list
dot1x-authentication-list
```

5. Enable the web authentication list.

Example

```
Device(config-wlan)# security web-auth authentication-list
authenticate-list-name
```

Example

```
Device(config-wlan)# security web-auth authentication-list default
```

6. Enable web authentication on MAC filter failure.

Example

```
Device(config-wlan)# security web-auth on-macfilter-failure
```

7. Configure the web authentication parameter map.

Example

```
Device(config-wlan)# security web-auth parameter-map parameter-map-name
```

Example

```
Device(config-wlan)# security web-auth parameter-map WLAN1_MAP
```

> **Note**
>
> If a parameter map is not associated with a WLAN, the configuration is considered from the global parameter map.

8. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

The WLAN is configured to use 802.1x as the primary authentication method, with web authentication as a fallback for clients that fail MAC filtering.

219 Wired Guest Access

Topics:

- Wired guest access

Guides the configuration of wired guest access through foreign and anchor controllers, detailing authentication methods and session management for secure guest connectivity.

Wired guest access

Describes a feature that allows guest users in an enterprise network to connect to a designated guest access network.

Wired guest access is a network feature that

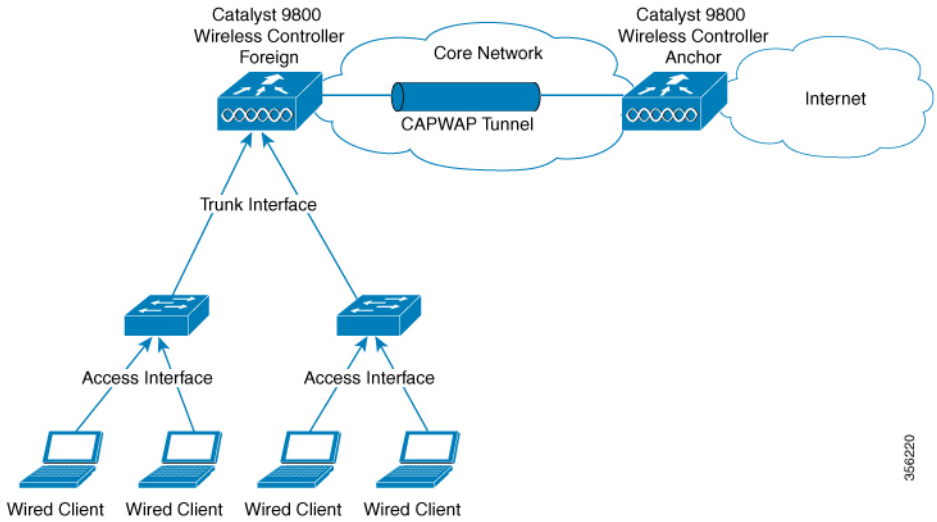
- enables guest users of an enterprise network that supports both wired and wireless access to connect to the guest access network
- allows wired guest clients to connect from designated and configured wired Ethernet ports for guest access after they complete the configured authentication methods, and
- directs wired session guests to a wireless guest controller in a demilitarized zone (DMZ) through a Control And Provisioning of Wireless Access Points (CAPWAP) tunnel.

Wired guest access configuration and architecture

Wired guest access can be configured in a dual-controller configuration that uses both an anchor controller and a foreign controller. A dual-controller configuration isolates wired guest access traffic from the enterprise user traffic.

The wired session guests are provided open or web-authenticated access from the wireless controller.

Figure 38: Guest access architecture



IPv6 Router Advertisement forwarding for wired guests

- **Guest Anchor Controller**: Guest anchor controller forwards the RA packets, from the receiving VLAN, to all the foreign controllers using the mobility data tunnel. The RA packets are tagged with the anchor VLAN to ensure the message is forwarded to the correct clients using the foreign controller data path.
- **Guest Foreign Controller**: Guest foreign controller forwards the received RAs from the guest anchor to the wired ports on which the wired guest clients are connected. To forward the RAs to the intended clients, the guest foreign controller keeps a track of the wired guest clients-per interface, access VLANs, and anchor VLANs.

Wired clients get the IPv6 based connectivity when they receive the IPv6 Router Advertisement (RA) message. The IPv6 router sends these RA messages and it contains information such as IPv6 prefix and router link-local address.

These RA messages are sent as Unicast or Multicast messages. The Unicast RA messages are routed as same as the client directed traffic. The Multicast RA messages are forwarded to all the clients present in the intended VLAN. RA message forwarding is enabled by default and requires no specific configuration.

Supported features

Supported features include:

- Cisco Catalyst 9800 Series Wireless Controllers-Anchor
- Cisco AireOS Wireless Controllers-Anchor
- Cisco Catalyst 9800 Series Wireless Controllers-Foreign
- Cisco AireOS Wireless Controllers-Foreign
- Dual controller solution (foreign + anchor) and access switch
- Trunk Ports
- Open Authentication
- Local Web Authentication
- Scale max 2k clients and 5 guest-LANs (5 VLANs max)
- Client IPv6 support
- Idle Timeout and Session Timeout
- Accounting on Foreign
- Manageability (SNMP/Yang/WebUI)
- QoS Rate-Limiting and MQC Policies (Upstream at foreign, Upstream, and Downstream at the anchor)
- QoS support with AireOS Anchor setup
- Stateful Switch Over (SSO)
- Port Channel support on Anchor and Foreign with no restrictions to the controller's role.
- Access Port on Foreign
- Cisco Umbrella (not supported in AireOS Anchor)
- ACL support at anchor
- Fully Qualified Domain Name (FQDN) URL filtering is supported at Anchor controller.
- IP theft detection
- Sleeping Client

Local web authentication options:

- Local Web Authentication (web consent)

> **Note**
>
> In AireOS, this is referred to as **web pass-through**.

- Local Web Authentication + ISE (External Web Authentication).
- LWA (local web authentication), with a username and a password.
- Web consent (LWA + consent), that is with a username, a password and the check box of acceptance.

To configure Web Authentication, see Web-based Authentication section of the Cisco Catalyst 9800 Series Wireless Controller Software Configuration Guide.

Note

Statistics computation not supported.

Note

QoS rate-limiting supports bps rate-limiting, pps rate-limiting is not supported.

Restrictions for wired guest access

Outlines the configuration and deployment restrictions for wired guest access implementation.

- A maximum of five guest LANs are supported on the foreign controller.
- A maximum of 2000 clients per foreign are supported.
- No Multicast or Broadcast support.
- You can map only one wired VLAN to a guest LAN.
- You can map only one guest LAN to one policy profile.
- Every guest LAN has a unique name and this name cannot be shared with RLAN or WLAN.
- Ensure that the Anchor VLAN ID and the wired VLAN ID configured on the Foreign controller is not the same.
- QoS is not supported on VLAN and on physical interfaces of the controller.

Configure access switch for wired guest client (CLI)

Configure access switch for wired guest client connectivity.

Set up an access switch to provide network connectivity for wired guest clients through VLAN assignment and port configuration.

Wired guest clients require dedicated VLAN assignment on access switches to segregate guest traffic from corporate network traffic and provide appropriate network access controls.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create the VLAN ID.

Example

```
Device(config)# vlan vlan-id
```

Example

```
Device(config)# vlan 200
```

3. Return to configuration mode.

Example

```
Device(config)# exit
```

4. Enter the interface to be added to the VLAN.

Example

```
Device(config)# interface GigabitEthernetinterface-number
```

Example

```
Device(config)# interface GigabitEthernet1/0/1
```

5. Assign the port to a VLAN.

Example

```
Device(config-if)# switchport access vlan vlan-id
```

Example

```
Device(config-if)# switchport access vlan 200
```

The valid VLAN IDs range is from 1 to 4094.

6. Define the VLAN membership mode for the port.

Example

```
Device(config-if)# switchport mode access
```

7. Disable CDP on the interface.

Example

```
Device(config-if)# no cdp enable
```

8. Save the configuration and exit to privileged EXEC mode.

Example

```
Device(config-if)# end
```

The access switch is configured with the appropriate VLAN assignment and port settings to provide network connectivity for wired guest clients.

Configure access switch for foreign controller (CLI)

Configure access switch for foreign controller.

Configure an access switch to support communication with a foreign controller by setting up appropriate VLAN and trunk configurations.

This configuration is needed when setting up an access switch to work with a foreign controller in a wireless network deployment. The switch must be configured with the proper VLAN and trunk settings to enable communication.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create the VLAN ID.

   Example

   ```
   Device(config)# vlan vlan-id
   ```

   Example

   ```
   Device(config)# vlan 200
   ```

3. Return to configuration mode.

   Example

   ```
   Device(config)# exit
   ```

4. Enter the interface to be added to the VLAN.

   Example

   ```
   Device(config)# interface GigabitEthernetinterface-number
   ```

   Example

   ```
   Device(config)# interface GigabitEthernet1/0/2
   ```

5. Assign the allowed VLAN ID to the port when it is in trunking mode.

   Example

   ```
   Device(config-if)# switchport trunk allowed vlan vlan-id
   ```

   Example

   ```
   Device(config-if)# switchport trunk allowed vlan 200
   ```

6. Set the trunking mode to trunk unconditionally.

   Example

   ```
   Device(config-if)# switchport mode trunk
   ```

7. Save the configuration and exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config-if)# end
   ```

The access switch is now configured with the appropriate VLAN and trunk settings to support communication with a foreign controller.

Configure foreign controller with open authentication (GUI)

Configure foreign controller with open authentication using GUI.

Configure a foreign controller with open authentication to enable wireless guest access through the GUI interface.

Foreign controllers allow guest clients to connect through external wireless controllers while maintaining centralized policy management. This configuration uses open authentication without web authentication requirements.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click on a **Policy Name**.
3. Go to the **Mobility** tab.
4. In the **Mobility Anchors** section, check the **Export Anchor** check box.
5. Click **Apply to Device**.
6. Choose **Configuration** > **Wireless** > **Guest LAN** > **Guest LAN Configuration**
7. Click **Add**.
8. In the **General** tab, enter the **Profile Name**, **Guest LAN ID**, **Client Association Limit**.
9. Choose the desired mode from the **mDNS Mode** drop-down list.
10. Enable or disable the **Status** and **Wired VLAN Status** toggle button.
11. In the **Security** tab, disable the **Web Auth** toggle button.
12. Click **Apply to Device**.
13. Choose **Configuration** > **Wireless** > **Guest LAN** > **Guest LAN Map Configuration**

14. Click **Add Map**.
15. In the Add Guest LAN Map window, enter the **Guest LAN Map**.
16. Click **Apply to Device**.
17. Click **Add**.
18. Choose the values from the **Profile Name** and **Policy Name** drop-down lists.
19. Click **Save**.

The foreign controller is configured with open authentication. Guest clients can now connect through the configured guest LAN without requiring web authentication.

Configure foreign controller with open authentication (CLI)

Configure foreign controller with open authentication using commands.

Set up a foreign controller with open authentication to enable guest access through mobility anchor configuration.

Foreign controllers are used in mobility deployments where guest traffic is tunneled to a designated anchor controller. This configuration establishes the necessary mobility anchor relationship and guest LAN settings for open authentication.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile.

   Example

   ```
   Device(config)# wireless profile policy wlan-policy-profile-name
   ```

   Example

   ```
   Device(config)# wireless profile policy testpro-1
   ```

3. Configure the mobility anchor and set its priority.

   Example

   ```
   Device(config-wireless-policy)# mobility anchor non-local-mobility-cntlr-ip
   priority priority
   ```

   Example

   ```
   Device(config-wireless-policy)# mobility anchor 192.168.201.111 priority 1
   ```

4. Enable the configuration.

Example

```
Device(config-wireless-policy)# no shutdown
```

5. Return to configuration mode.

Example

```
Device(config-wireless-policy)# exit
```

6. Configure guest LAN profile with a wired VLAN.

Example

```
Device(config)# guest-lan profile-name guest-profile-name guest-lan-id
wired-vlan wired-vlan-id
```

Example

```
Device(config)# guest-lan profile-name gstpro-1 1 wired-vlan 25
```

> **Note**
>
> Configure the wired VLAN only for the Guest Foreign controller.

7. Disable web-authentication.

Example

```
Device(config-guest-lan)# no security web-auth
```

8. Enable the guest LAN.

Example

```
Device(config-guest-lan)# no shutdown
```

9. Return to configuration mode.

Example

```
Device(config-guest-lan)# exit
```

10. Configure a guest LAN map.

Example

```
Device(config)# wireless guest LAN map gst-map-name
```

Example

```
Device(config)# wireless guest LAN map gstmap-1
```

11. Attach a guest LAN map to the policy profile.

Example

```
Device(config-guest-lan-map)# guest-lan guest-profile-name policy
wlan-policy-profile-name
```

Example

```
Device(config-guest-lan-map)# guest-lan gstpro-1 policy testpro-1
```

12. Return to configuration mode.

Example

```
Device(config-guest-lan-map)# exit
```

The foreign controller is now configured with open authentication, and guest traffic will be tunneled to the designated mobility anchor controller.

Configure foreign controller with local web authentication (GUI)

Configure foreign controller with local web authentication using GUI.

Configure a foreign controller to enable local web authentication for guest access through the graphical user interface.

Local web authentication allows guest users to authenticate through a web portal when connecting to the wireless network. This configuration involves setting up mobility anchors, guest LAN configuration, and guest LAN mapping to enable the authentication process.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Select a **Policy Name**.
3. Go to the **Mobility** tab.
4. In the **Mobility Anchors** section, check the **Export Anchor** check box.
5. Click **Update & Apply to Device**.
6. Choose **Configuration** > **Wireless** > **Guest LAN** > **Guest LAN Configuration**
7. Click **Add**.
8. In the **General** tab, enter the **Profile Name**, **Guest LAN ID**, **Client Association Limit**.
9. Choose the desired mode from the **mDNS Mode** drop-down list.
10. Enable or disable the **Status** and **Wired VLAN Status** using toggle button.
11. Go to the **Security** tab.
12. Enable the **Web Auth** using toggle button.
13. Choose the values from the **Web Auth Parameter Map**, **Authentication List** and **Authorization List** drop-down lists.

14. Click **Apply to Device**.
15. Choose **Configuration** > **Wireless** > **Guest LAN** > **Guest LAN Map Configuration**
16. Click **Add Map**.
17. In the Add Guest LAN Map window, enter the **Guest LAN Map**.
18. Click **Apply to Device**.
19. Click **Add**.
20. Choose the values from the **Profile Name** and **Policy Name** drop-down lists.
21. Click **Save**.

The foreign controller is successfully configured with local web authentication. Guest users can now authenticate through the web portal when connecting to the wireless network.

Configure foreign controller with local WEB authentication (CLI)

Configure foreign controller with local WEB authentication using commands.

Enable foreign controller functionality with local WEB authentication to provide secure guest access through mobility anchoring.

Foreign controllers work with anchor controllers in mobility scenarios to provide seamless roaming and secure guest access. This configuration establishes the necessary policy profiles and guest LAN mappings for local WEB authentication.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile.

   Example

   ```
   Device(config)# wireless profile policy wlan-policy-profile-name
   ```

   Example

   ```
   Device(config)# wireless profile policy testpro-1
   ```

3. Configure the mobility anchor and set its priority.

   Example

   ```
   Device(config-wireless-policy)# mobility anchor non-local-mobility-cntlr-ip
   priority priority
   ```

   Example

   ```
   Device(config-wireless-policy)# mobility anchor 192.168.201.111 priority 1
   ```

4. Enable the configuration.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

5. Return to configuration mode.

   Example

   ```
   Device(config-wireless-policy)# exit
   ```

6. Configure guest LAN profile with a wired VLAN.

   Example

   ```
   Device(config)# guest-lan profile-name guest-profile-name guest-lan-id
   wired-vlan wired-vlan-id
   ```

   Example

   ```
   Device(config)# guest-lan profile-name gstpro-2 3 wired-vlan 26
   ```

7. Enable WEB-authentication.

   Example

   ```
   Device(config-guest-lan)# security web-auth
   ```

8. Configure the authentication list for a IEEE 802.1x network.

   Example

   ```
   Device(config-guest-lan)# security web-auth authentication-list auth-list-name
   ```

   Example

   ```
   Device(config-guest-lan)# security web-auth authentication-list default
   ```

9. Configure the security WEB-auth parameter map.

   Example

   ```
   Device(config-guest-lan)# security web-auth parameter-map parameter-map-name
   ```

   Example

   ```
   Device(config-guest-lan)# security web-auth parameter-map global
   ```

10. Enable the guest LAN.

Example

```
Device(config-guest-lan)# no shutdown
```

11. Return to configuration mode.

Example

```
Device(config-guest-lan)# exit
```

12. Configure a guest LAN map.

Example

```
Device(config)# wireless guest-lan map gst-map-name
```

Example

```
Device(config)# wireless guest-lan map gstmap-2
```

13. Attach a guest LAN map to the policy profile.

Example

```
Device(config-guest-lan-map)# guest-lan guest-lan-profile-name policy
policy-profile-name
```

Example

```
Device(config-guest-lan-map)# guest-lan gstpro-2 policy testpro-1
```

14. Return to configuration mode.

Example

```
Device(config-guest-lan-map)# exit
```

The foreign controller is now configured with local WEB authentication, enabling secure guest access through the mobility anchor setup and guest LAN mapping.

What to do next
For more information about Local WEB Authentication, see https://www.cisco.com/c/en/us/td/docs/wireless/controller/9800/config-guide/b_wl_16_10_cg/wireless-web-authentication.html

Configure anchor controller with open authentication (GUI)

Configure anchor controller with open authentication using GUI.

This task configures an anchor controller with open authentication to enable guest access and mobility management for wireless clients.

Use this procedure when you need to set up mobility anchoring for guest users without requiring web authentication. This configuration creates a policy profile with mobility anchor settings and a corresponding guest LAN for open authentication.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click **Add**.
3. In the **General** tab, enter the **Name**.
4. Go to the **Access Policies** tab.
5. Under the **VLAN** settings, choose the vlans from the **VLAN/VLAN Group** drop-down list.
6. Go to the **Mobility** tab.
7. Under the **Mobility Anchors** settings, check the **Export Anchor** check box.
8. Click **Apply to Device**.
9. Choose **Configuration** > **Wireless** > **Guest LAN**.
10. Click **Add**.
11. In the **General** tab, enter the **Profile Name**, the **Guest LAN ID** and the **Client Association Limit**.
12. In the **Security** tab, under the **Layer3** settings, disable the **Web Auth** toggle button.
13. Click **Apply to Device**.

The anchor controller is now configured with open authentication. The policy profile with mobility anchor settings and the guest LAN are created and applied to the device.

Configure anchor controller with open authentication (CLI)

Configure anchor controller with open authentication using commands.

Enable anchor controller functionality with open authentication for guest access scenarios.

Anchor controllers provide centralized guest access management by terminating guest traffic at a designated controller while maintaining security isolation from the main network.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile.

   Example

   ```
   Device(config)# wireless profile policy wlan-policy-profile-name
   ```

   Example

   ```
   Device(config)# wireless profile policy testpro-2
   ```

3. Configure the mobility anchor.

Example

```
Device(config-wireless-policy)# mobility anchor
```

4. Configure a VLAN name or a VLAN ID.

Example

```
Device(config-wireless-policy)# vlan vlan-id
```

Example

```
Device(config-wireless-policy)# vlan 29
```

5. Enable the configuration.

Example

```
Device(config-wireless-policy)# no shutdown
```

6. Return to configuration mode.

Example

```
Device(config-wireless-policy)# exit
```

7. Configure the guest LAN profile with a wired VLAN.

Example

```
Device(config)# guest-lan profile-name guest-profile-name guest-lan-id
```

Example

```
Device(config)# guest-lan profile-name testpro-2 1
```

8. Configure the maximum client connections per guest LAN.

Example

```
Device(config-guest-lan)# client association limit guest-lan-client-limit
```

The valid range is from 1 to 2000.

9. Disable web authentication.

Example

```
Device(config-guest-lan)# no security web-auth
```

10. Enable the guest LAN.

Example

```
Device(config-guest-lan)# no shutdown
```

11. Return to configuration mode.

Example

```
Device(config-guest-lan)# exit
```

The anchor controller is now configured with open authentication, allowing guest clients to access the network through the designated guest LAN profile with the specified VLAN assignment.

Configure anchor controller with local web authentication (GUI)

Configure anchor controller with local web authentication using GUI.

Configure an anchor controller with local web authentication to establish mobility anchoring for guest access control.

Use this procedure when you need to set up an anchor controller that handles local web authentication for guest users in a wireless network environment.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click **Add**.
3. In the **General** tab, enter the **Name**.
4. Go to the **Access Policies** tab.
5. Under the **VLAN** settings, choose the vlans from the **VLAN/VLAN Group** drop-down list.
6. Go to the **Mobility** tab.
7. Under the **Mobility Anchors** settings, check the **Export Anchor** check box.
8. Click **Apply to Device**.
9. Choose **Configuration** > **Wireless** > **Guest LAN**.
10. Click **Add**.
11. In the **General** tab, enter the **Profile Name**, the **Guest LAN ID** and the **Client Association Limit**.
12. In the **Security** tab, under the **Layer3** settings, enable the **Web Auth** toggle button. Choose the Parameter map from the **Web Auth Parameter Map** drop-down list and the authentication list from the **Authentication List** drop-down list.
13. Click **Apply to Device**.

The anchor controller is configured with local web authentication. The policy profile and guest LAN configuration are applied to the device and ready to handle guest user authentication.

Configure anchor controller with local web authentication (CLI)

Configure anchor controller with local web authentication using commands.

Establish an anchor controller configuration with local web authentication to manage client connectivity and security.

Anchor controllers provide centralized authentication and policy enforcement for wireless clients across multiple wireless LAN controllers. This configuration enables local web authentication for guest access scenarios.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile.

   Example

   ```
   Device(config)# wireless profile policy wlan-policy-profile-name
   ```

   Example

   ```
   Device(config)# wireless profile policy testpro-2
   ```

3. Configure the mobility anchor.

   Example

   ```
   Device(config-wireless-policy)# mobility anchor
   ```

4. Configure a VLAN name or a VLAN ID.

   Example

   ```
   Device(config-wireless-policy)# vlan vlan-id
   ```

   Example

   ```
   Device(config-wireless-policy)# vlan 30
   ```

5. Enable the configuration.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

6. Return to configuration mode.

   Example

   ```
   Device(config-wireless-policy)# exit
   ```

7. Configure a guest LAN profile with a wired VLAN.

   Example

   ```
   Device(config)# guest-lan profile-name guest-profile-name guest-lan-id
   ```

Example

```
Device(config)# guest-lan profile-name testpro-2 1
```

8. Configure the maximum client connections per guest LAN.

Example

```
Device(config-guest-lan)# client association limit guest-lan-client-limit
```

The valid range is between 1 and 2000.

9. Configure web authentication.

Example

```
Device(config-guest-lan)# security web-auth
```

10. Configure the security web-auth parameter map.

Example

```
Device(config-guest-lan)# security web-auth parameter-map parameter-map-name
```

Example

```
Device(config-guest-lan)# security web-auth parameter-map testmap-1
```

11. Configure the authentication list for the IEEE 802.1x network.

Example

```
Device(config-guest-lan)# security web-auth authentication-list
authentication-list-name
```

Example

```
Device(config-guest-lan)# security web-auth authentication-list testlwa-1
```

12. Enable the guest-LAN.

Example

```
Device(config-guest-lan)# no shutdown
```

13. Return to configuration mode.

Example

```
Device(config-guest-lan)# exit
```

The anchor controller is now configured with local web authentication, enabling secure guest access through the designated guest LAN profile with web-based authentication.

Configure session timeout for a profile policy (CLI)

Configure session timeout for a profile policy.

Configure session timeout values for wired guests to control the duration of their network sessions.

Session timeout for a wired guest is set to infinite by default. Use this procedure to configure specific timeout values for wired guests.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN policy profile.

   Example

   ```
   Device(config)# wireless profile policy wlan-policy-profile-name
   ```

   Example

   ```
   Device(config)# wireless profile policy testpol-1
   ```

3. Enable the client session timeout on the guest LAN.

   Example

   ```
   Device(config-wireless-policy)# guest-lan enable-session-timeout
   ```

4. Configure the client session timeout in seconds.

   Example

   ```
   Device(config-wireless-policy)# session-timeout timeout-duration
   ```

   Example

   ```
   Device(config-wireless-policy)# session-timeout 1000
   ```

   The valid range is from 0 to 86400 seconds.

Session timeout is now configured for the profile policy, controlling how long wired guest sessions remain active before automatic disconnection.

Configure global settings (GUI)

Configure global settings including user administration and HTTP/HTTPS access using the GUI.

Configure global system settings to establish user accounts and manage HTTP/HTTPS access for device management.

Global configuration includes setting up user administration and configuring HTTP/HTTPS access settings for secure device management.

Procedure

1. Choose **Administration** > **User Administration**.
2. Click **Add**.
3. Enter the **Username**, **Password** and **Confirm Password**.
4. Choose the desired value from the **Policy** and **Privilege** drop-down lists.
5. Click **Apply to Device**.
6. Choose **Administration** > **Management** > **HTTP/HTTPS/Netconf**.
7. In the **HTTP/HTTPS Access Configuration** settings, enable or disable the **HTTP Access**, **HTTPS Access** and **Personal Identity Verification** toggle buttons.
8. Enter the **HTTP Port** and **HTTPS Port**.
9. Click **Apply**.

The global configuration is applied with the new user account created and HTTP/HTTPS access settings configured according to your specifications.

Verify wired guest configurations

Provides commands to verify wired guest configurations.

This reference provides commands to verify wired guest configurations and view guest-LAN details and client information.

To validate the wireless configuration, use this command:

```
Device# wireless config validate

Wireless Management Trustpoint Name: 'WLC-29c_WLC_TP'
        Trustpoint certificate type is WLC-SSC
    Wireless management trustpoint config is valid

Jan 22 07:49:15.371: %CONFIG_VALIDATOR_MESSAGE-5-EWLC_GEN_ERR: Chassis 1 R0/0:
 wncmgrd:
Error in No record found for VLAN 9, needed by Guest-LAN open-wired
```

To display the summary of all Guest-LANs, use this command:

```
Device# show guest-lan summary

Number of Guest LANs: 1

GLAN  GLAN Profile Name                  Status
-----------------------------------------------
1     wired_guest_open                   UP
```

To view the detailed output of all Guest-LANs, use this command:

```
Device# show guest-lan all

Guest-LAN Profile Name     : open
================================================
Guest-LAN ID                                  : 1
Wired-Vlan                                    : 200
Status                                        : Enabled
Number of Active Clients                      : 1
Max Associated Clients                        : 2000
Security
    WebAuth                                   : Enabled
    Webauth Parameter Map                     : global
    Webauth Authentication List               : LWA-AUTHENTICATION
    Webauth Authorization List                : LWA-AUTHENTICATION
```

To view the guest-LAN configuration by ID, use this command:

```
Device# show guest-lan id 1
Guest-LAN Profile Name     : open
================================================
Guest-LAN ID                                  : 1
Wired-Vlan                                    : 200
Status                                        : Enabled
Number of Active Clients                      : 1
Max Associated Clients                        : 2000
Security
    WebAuth                                   : Enabled
    Webauth Parameter Map                     : global
    Webauth Authentication List               : LWA-AUTHENTICATION
    Webauth Authorization List                : LWA-AUTHENTICATION
```

To view the guest-LAN configuration by profile name, use this command:

```
Device# show guest-lan name open

Guest-LAN Profile Name     : open
================================================
Guest-LAN ID                                  : 1
Wired-Vlan                                    : 200
Status                                        : Enabled
Number of Active Clients                      : 1
Max Associated Clients                        : 2000
Security
    WebAuth                                   : Enabled
    Webauth Parameter Map                     : global
    Webauth Authentication List               : LWA-AUTHENTICATION
    Webauth Authorization List                : LWA-AUTHENTICATION
```

To view the guest-LAN map summary, use this command:

```
Device# show wireless guest-lan-map summary

Number of Guest-Lan Maps: 2

WLAN Profile Name                 Policy Name
------------------------------------------------------------------------
```

```
open_wired_guest                  open_wired_guest
lwa_wired_guest                   lwa_wired_guest
```

To view the active clients, use this command:

```
Device# show wireless client summary

Number of Local Clients: 1

MAC Address    AP Name                                          Type ID   State
             Protocol Method     Role
-------------------------------------------------------------------------------
000a.bd15.0001 N/A                                              GLAN 1    Run
             802.3    Web Auth   Export Foreign
```

To view the detailed information about a client by MAC address, use this command:

```
Device# show wireless client mac-address 3383.0000.0001 detail

Client MAC Address : 3383.0000.0001
Client IPv4 Address : 155.165.152.151
Client Username: N/A
AP MAC Address:  N/A
AP slot : N/A
Client State : Associated
Policy Profile : guestlan_lwa
Flex Profile : N/A
Guest Lan:
  GLAN Id: 2
  GLAN Name: guestlan_lwa
  Wired VLAN: 312
Wireless LAN Network Name (SSID) : N/A
BSSID : N/A
Connected For : 128 seconds
Protocol : 802.3
Channel : N/A
Client IIF-ID : 0xa0000002
Association Id : 0
Authentication Algorithm : Open System
Session Timeout : 1800 sec (Timer not running)
Session Warning Time : Timer not running
Input Policy Name  : clsilver
Input Policy State : Installed
Input Policy Source : AAA Policy
Output Policy Name  : None
Output Policy State : None
Output Policy Source : None
WMM Support : Disabled
Fastlane Support : Disabled
Power Save : OFF
AAA QoS Rate Limit Parameters:
  QoS Average Data Rate Upstream             : 0 (kbps)
  QoS Realtime Average Data Rate Upstream    : 0 (kbps)
  QoS Burst Data Rate Upstream               : 0 (kbps)
  QoS Realtime Burst Data Rate Upstream      : 0 (kbps)
  QoS Average Data Rate Downstream           : 0 (kbps)
  QoS Realtime Average Data Rate Downstream  : 0 (kbps)
  QoS Burst Data Rate Downstream             : 0 (kbps)
  QoS Realtime Burst Data Rate Downstream    : 0 (kbps)
```

```
Mobility:
  Anchor IP Address           : 101.0.0.1
  Point of Attachment         : 0x00000008
  Point of Presence           : 0xA0000001
  AuthC status                : Enabled
  Move Count                  : 0
  Mobility Role               : Export Foreign
  Mobility Roam Type          : L3 Requested
  Mobility Complete Timestamp : 05/07/2019 22:31:45 UTC
Client Join Time:
  Join Time Of Client : 05/07/2019 22:31:42 UTC
Policy Manager State: Run
Last Policy Manager State : IP Learn Complete
Client Entry Create Time : 125 seconds
Policy Type : N/A
Encryption Cipher : N/A
Encrypted Traffic Analytics : No
Protected Management Frame - 802.11w : No
EAP Type : Not Applicable
VLAN : default
Multicast VLAN : 0
Access VLAN : 153
Anchor VLAN : 155
WFD capable : No
Managed WFD capable : No
Cross Connection capable : No
Support Concurrent Operation : No
Session Manager:
  Point of Attachment : TenGigabitEthernet0/0/0
  IIF ID             : 0x00000008
  Authorized         : TRUE
  Session timeout    : 1800
  Common Session ID: 00000000000000CB946C8BA3
  Acct Session ID  : 0x00000000
  Last Tried Aaa Server Details:
   Server IP :
  Auth Method Status List
   Method : Web Auth
    Webauth State    : Authz
    Webauth Method   : Webauth
  Local Policies:
   Service Template : wlan_svc_guestlan_lwa_local (priority 254)
    VLAN             : 153
    Absolute-Timer   : 1800
  Server Policies:
    QOS Level        : 0
  Resultant Policies:
    VLAN Name         : VLAN0153
    QOS Level        : 0
    VLAN             : 153
    Absolute-Timer   : 1800
DNS Snooped IPv4 Addresses : None
DNS Snooped IPv6 Addresses : None
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 0
Fast BSS Transition Details :
  Reassociation Timeout : 0
```

```
11v BSS Transition : Not implemented
11v DMS Capable : No
QoS Map Capable : No
FlexConnect Data Switching : N/A
FlexConnect Dhcp Status : N/A
FlexConnect Authentication : N/A
FlexConnect Central Association : N/A
Client Statistics:
  Number of Bytes Received : 0
  Number of Bytes Sent : 0
  Number of Packets Received : 8
  Number of Packets Sent : 0
  Number of Policy Errors : 0
  Radio Signal Strength Indicator : 0 dBm
  Signal to Noise Ratio : 0 dB
  Idle time : 0 seconds
  Last idle time update : 05/07/2019 22:32:27
  Last statistics update : 05/07/2019 22:32:27
Fabric status : Disabled
Client Scan Reports
Assisted Roaming Neighbor List
Nearby AP Statistics:
EoGRE : Pending Classification
```

Wired guest access use cases

Describes use cases for wired guest access.

Wired guest access use cases are network scenarios that utilize wired guest access functionality to meet specific operational requirements beyond standard guest network access.

Equipment software update scenario

Video streaming scenario

This feature can be configured to allow the wired port to connect to the manufacture or vendor website for equipment maintenance, software, or firmware updates.

This feature can be configured to allow devices that are connected to a wired port to stream video to visitor information screens.

220 Data Rate Selection at SSID Level

Topics:

- Legacy data rate selection at SSID level

Legacy data rate selection at SSID level

This section explains legacy data rate selection at SSID level.

Legacy data rate selection at SSID level is a wireless configuration feature that:

- allows the configuration of 802.11a/b/g (legacy) data rates per WLAN or Virtual AP (VAP)
- specifies the minimum bit rate to be used by each radio for a specific SSID, and
- ensures that the AP does not advertise rates lower than the configured minimum, even if those rates are enabled in the RF profile.

Previously, you managed legacy data rate configurations at the global or RF profile level. This feature provides granular control at the SSID level, which is useful for deployments requiring specific performance thresholds for different user groups or applications.

Bit rate defaults and ranges

The values to configure a legacy date rate in RF profile configuration depends on the frequency band:

- For 2.4 GHz, the default is 1 Mbps. The valid range is 1 to 54 Mbps.
- For 5 GHz, the default is 6 Mbps. The valid range is 6 to 54 Mbps.

> **Note**
>
> This configuration does not apply to the 6 GHz radio band.

Feature history for legacy data rate selection at SSID level

This section explains the feature history details of the legacy data rate selection at SSID level feature.

- This table provides release and related information for the feature explained in this module.
- This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 193: Feature history for legacy data rate selection at SSID level

Feature Name	Release Information	Feature Description
Legacy data rate selection at SSID level	Cisco IOS XE 26.1.1	SSID-level legacy data rate selection enables the customization of legacy 802.11a/b/g speeds for individual WLANs or Virtual APs. This feature enforces a minimum bit rate for each radio, ensuring the AP does not advertise any speeds below that threshold, even if those lower rates are permitted in the general RF profile.

Guidelines for legacy data rate selection

This section explains the guidelines for legacy data rate selection

Requirement: Radio band applicability

Ensure that minimum data rate configurations apply only to the 2.4 GHz and 5 GHz bands. This feature does not support or impact the 6 GHz radio band.

RF profile precedence

The minimum data rate configured at the SSID level acts as a threshold. The AP does not advertise legacy rates lower than this threshold, even if those rates are marked as supported or mandatory in the RF profile.

Mandatory rate alignment

At least one mandatory rate in the RF profile must be higher than the minimum data rate set for the SSID. If no mandatory rate is higher than the SSID minimum data rate, the configuration fails and an error appears.

Configure the minimum data rate (GUI)

Use this procedure to configure the minimum data rate using the GUI.

Set a minimum legacy bit rate threshold for a specific WLAN to control which data rates the AP advertises.

Use this task to ensure that clients can only connect using data rates above a specific threshold for a SSID. This setting overrides broader RF profile settings.

Before you begin

Make sure you have created and can access the WLAN. Determine the minimum bit rate that meets your use case to improve cell efficiency.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click the **WLAN Name** you want to configure, or click **Add** to create a new one.
3. In the **Edit WLAN** or **Add WLAN** window, click the **Advanced** tab.
4. Locate the **Minimum Data Rates At SSID Level** section.
5. From the **2.4 GHz Data Rate** and **5 GHz Data Rate** drop-down lists, select the desired minimum bit rate.
6. Click **Update & Apply to Device**.

The WLAN is updated with the new minimum data rate thresholds. The APs broadcasting this SSID now advertise legacy rates at or above the minimum values you set.

Configure the minimum data rate (CLI)

Use this procedure to configure the minimum data rate using commands.

Configure the minimum data rate for 2.4 GHz and 5 GHz radios on your WLAN using commands.

Minimum data rate settings control which devices connect to a WLAN and at what speeds. These settings help remove support for slower rates and improve throughput.

Before you begin

Make sure you can access the CLI on your wireless controller. Know your WLAN profile name, SSID, and preferred data rate values for 2.4 GHz and 5 GHz radios.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the WLAN network and enter the WLAN profile name.

   Example

   ```
   Device(config)# wlan (default-wlan-profile)
   ```

   You can use up to 32 alphanumeric characters for the WLAN profile name.

3. Create WLAN identifier and enter the SSID (network) name.

   Example

   ```
   Device(config)# wlan (default-wlan-profile) <1-4096> SSID50
   ```

   The valid WLAN identifier range is 1 to 4096. You can use up to 32 alphanumeric characters for the SSID name.

4. Configure 802.11 parameters.

   Example

   ```
   Device(config-wlan)# dot11
   ```

5. Configure 802.11a and 802.11b parameters and data rates.

   Example

   ```
   Device(config-wlan)# dot11 5ghz data-rate
   ```

   ```
   Device(config-wlan)# dot11 24ghz data-rate
   ```

6. Configure minimum data rate and set the minimum data rate for 2.4 GHz and 5 GHz radios.

   Example

   ```
   Device(config-wlan)# dot11 24ghz data-rate min RATE_9M
   ```

   ```
   Device(config-wlan)# dot11 5ghz data-rate min RATE_12M
   ```

   The minimum data rate for 2.4 GHz radio is 9 Mbps. The minimum data rate for 5 GHz radio is 12 Mbps.

7. Exit the global configuration mode.

Example

```
Device(config-wlan)# end
```

Verify RF profile and radio slot status

This section provides how to verify RF profile and radio slot status details.

Use this command to verify if the RF profile is assigned to the radio and to also check the radio slot status.

- **show ap rf-profile summary**

To verify if the RF profile is assigned to the radio and to also check the radio slot status, enter this command from the console:

```
Device# show ap rf-profile summary

Number of RF-profiles: 17
RF Profile Name                  Band     Description
  State
------------------------------------------------------------------------------------
t                                2.4 GHz
  Down
test                             2.4 GHz
  Down
test1                            5 GHz
  Down
rf-test                          2.4 GHz
  Down
test_5gh                         5 GHz
  Down
test_24gh                        2.4 GHz
  Up
test_5ghz                        5 GHz
  Up
test_24ghz                       2.4 GHz
  Up
5ghz-rf-policy                   5 GHz
  Down
default-rf-profile               2.4 GHz
  Down
default-rf-profile-6ghz          6 GHz    default rfprofile for 6GHz radio
  Up
Low_Client_Density_rf_5gh        5 GHz    pre configured Low Client Density
rf Up
High_Client_Density_rf_5gh       5 GHz    pre configured High Client Density
r Up
Low_Client_Density_rf_24gh       2.4 GHz  pre configured Low Client Density
rf Up
High_Client_Density_rf_24gh      2.4 GHz  pre configured High Client Density
r Up
Typical_Client_Density_rf_5gh    5 GHz    pre configured Typical Density rfpro
 Up
Typical_Client_Density_rf_24gh   2.4 GHz  pre configured Typical Client Densit
 Up
```

Verify RF profile configuration

This section provides how to verify RF profile configuration details.

Use this command to verify RF profile configuration.

- **show wireless profile radio detailed <profile-name>**

To verify RF profile configuration, enter this command from the console:

```
Device# show wireless profile radio detailed

WORD  radio-profile name

Device# show wireless profile radio detailed default-radio-profile

Radio Profile name                                : default-radio-profile
Description                                       : Preconfigured default radio
 profile
Beam-Selection                                    : Not configured
Number of antenna to be enabled                   : 0
Mesh Backhaul                                     : Enabled
Mesh Designated Downlink                          : Disabled
DTIM period                                       : 1

URWB
Role                                            : Not Configured
Channel                                           : Not Configured
Channel Width                                     : 20 MHz
AES                                               : Fixed Key
Key Control Rotation Timeout                      : 15
Channel List                                      : Not Configured
Point-to-Multi-Point
RSSI Threshold                                  : 0
Auto Scan                                       : Disabled
Cluster ID                                      : CiscoURWB
Tower ID                                        : Not Configured
Packet Retries                                    : 32
Maximum Link Distance (km)                        : 3
Dot11ax Maximum MCS Index                         : MCS9

Wireless Active Testing (WAT) Configuration
WAT Admin State (Radio Selection)                 : Disabled
```

221 User Defined Network

Topics:

- User defined networks

Introduces User Defined Network capabilities, allowing administrators to create customized network topologies and configurations tailored to specific organizational requirements.

User defined networks

Describes a solution that provides secure and remote on-boarding of devices in shared service environments.

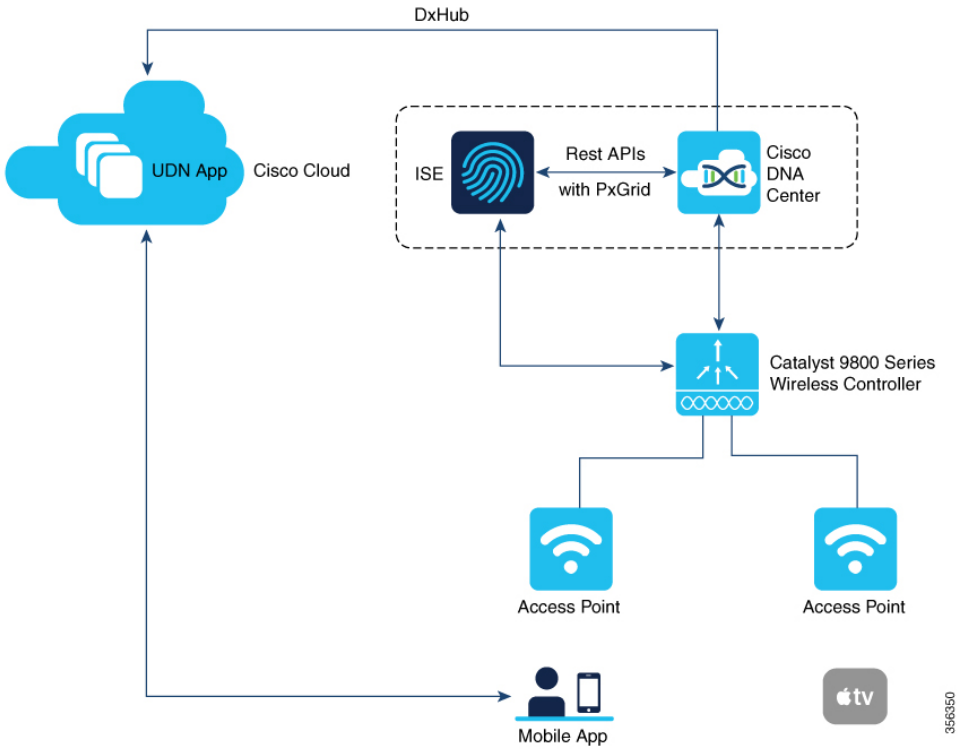
A user defined network (UDN) is a solution that

- provides secure and remote on-boarding of devices in shared service environments like dormitory rooms, resident halls, class rooms and auditoriums
- allows users to securely use Simple Discovery Protocols (SDP) like Apple Bonjour and mDNS-based protocols (Air Play, Air Print, Screen Cast, Print, and so on.), and UPnP based protocols to interact and share information with only their registered devices in a shared environment, and
- enables the users to share their devices and resources with friends and roommates securely.

UDN solution features

The UDN solution provides an easy way to create a virtual segment that allows user to create a private segment to add their devices. Traffic (unicast, non-Layer 3 multicast, or broadcast) to these devices can be seen only by other devices and users in the private segment. This feature also eliminates the security concern where users knowingly or unknowingly take control of devices that belong to other users in a shared environment. As of now, the UDN is supported only in local mode.

Figure 39: User defined network topology



The User Defined Network solution workflow includes the following steps:

1. User Defined Network is enabled on the controller, using policy profile, and the policy configuration is pushed to all the WLANs on a site.
2. User Defined Network association is automatically generated by the UDN cloud service and is inherited by all the devices belonging to an user.

3. Users can add or modify devices to the User Defined Network assigned to them by using a web portal or a mobile application. Users can also add devices to another User Defined Network, if they are invited to join that User Defined Network.
4. The controller is updated with the client or resource information assigned to the User Defined Network.

> **Note**
>
> Cisco Identity Services Engine (ISE) policy infrastructure is not used to update User Defined Network information. Whenever, there is a change in the User Defined Network, the ISE updates the controller with an explicit or a separate Change of Authorization (CoA) containing only the change of the User Defined Network ID.

Restrictions for User Defined Network

Outlines restrictions and limitations for User Defined Network feature implementation and deployment.

Consider these restrictions when implementing User Defined Network:

- A user can be associated to only one UDN. Roaming across controllers is not supported.
- This feature is not applicable for Cisco Mobility Express and Cisco AireOS platforms. Hence, IRCM is not supported.
- This feature is supported only in local mode on the Wave 2 access points and Cisco Catalyst 9100 series access points.
- This feature is supported only for centrally switched SSIDs.
- This feature is not supported for Flex mode APs, Fabric SSIDs, Guest Anchor scenario, Layer 2 and Layer 3 roaming.
- Layer 3 multicast (except SSDP/UPnP) containment using UDN is not supported, L3 multicast will continue to work as it is today.
- It is not recommended to enable user-defined **drop unicast** option in GUI if UDN is disabled.
- It is recommended to disable UDN from policy profile in SSIDs when integrating UDN+ solution with Cisco exclusive partner Splash Access.

  > **Note**
  >
  > The UDN+ simplifies the solution and brings the same unique user experience for both Meraki and Cisco Catalyst 9800 Series Wireless Controller based deployments.

- For user defined network plus deployments, do not configure **peer-blocking drop** or **peer-blocking allow-private-group** on the WLAN used for UDN. These settings can prevent UDN from functioning.

  For user defined network plus deployments, do not configure **user-defined-network drop-unicast** in the wireless policy profile. This setting can prevent UDN from functioning, even when peer blocking is not configured.

Configure a user defined network (CLI)

Configure a user defined network to enable site-based policy enforcement across WLANs with mDNS filtering based on UDN-ID.

Deploy user defined network configuration to enforce policies across all clients or devices in a network for a site.

The User Defined Network configuration is site based and is added as part of a policy profile. When applied, the policy is enforced to all the clients or devices in a network for a site, across WLANs. When enabled, the policy profile also enforces the filtering of mDNS queries based on the UDN-ID.

Before you begin

- RADIUS server should be configured for the UDN solution to work.
- Configure **aaa-override** in the policy profile.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Create a policy profile.

   Example

   ```
   Device(config)# wireless profile policy profile-name
   ```

   ```
   Device(config)# wireless profile policy policy-wpn
   ```

   profile-name is the profile name of the policy profile.

3. Enable user defined private-network.

   Example

   ```
   Device(config-wireless-policy)# user-defined-network
   ```

4. Set action to drop unicast traffic.

   Example

   ```
   Device(config-wireless-policy)# user-defined-network drop-unicast
   ```

   By default, unicast traffic is allowed across UDN.

   > ⚠ **Caution**
   >
   > If you enable **user-defined-network drop-unicast**, it can disrupt UDN operation and affect client connectivity.

5. Exit to global configuration mode.

Example

```
Device(config-wireless-policy)# exit
```

6. Configure a remote LAN policy profile.

Example

```
Device(config)# ap remote-lan-policy policy-name policy-name
```

```
Device(config)# ap remote-lan-policy policy-name policy-wpn
```

7. Enable user defined private-network for remote LAN policy.

Example

```
Device(config-remote-lan-policy)# user-defined-network
```

8. Set action to drop unicast traffic for remote LAN policy.

Example

```
Device(config-remote-lan-policy)# user-defined-network drop-unicast
```

The user defined network is configured with both wireless policy profile and remote LAN policy, enabling site-based enforcement and mDNS filtering based on UDN-ID across all network clients.

Configure a user defined network (GUI)

Configure a user defined network using GUI.

Configure a user defined network to enable a user personal network and control unicast traffic behavior.

Use this procedure when you need to enable user defined networks and configure unicast traffic handling through the GUI interface.

Procedure

---

1. Choose **Configuration > Tags & Profiles > Policy**.
2. In the **Policy Profile** window, select a policy profile.
3. In the **Edit Policy Profile** window, click the **Advanced** tab.
4. In the **User Defined Network** section, check the **Status** check box to enable a user personal network.
5. Check the **Drop Unicast** check box to set the action to Drop Unicast traffic.

   By default, unicast traffic is not contained.

The user defined network is configured with the specified unicast traffic handling settings.

Verify user defined network configuration

Lists commands to verify user defined network configuration status and client information.

Use the commands in this reference to verify the status of the User Defined Network (UDN) feature, view client UDN information, check UDN payloads, and examine mDNS service instances for private networks.

To view the status of the UDN feature (either enabled or disabled) and also information about the drop unicast flag, use this command:

```
Device# show wireless profile policy detailed default-policy-profile

User Defined (Private) Network                : Enabled
User Defined (Private) Network Unicast Drop   : Enabled
```

To view the name of the UDN to which the client belongs, use this command:

```
Device# show wireless client mac-address 00:0d:ed:dd:35:80 detailed

User Defined (Private) Network : Enabled
User Defined (Private) Network Drop Unicast : Enabled
               Private group name: upn*group*7
               Private group id : 7777
               Private group owner: 1
               Private group name: upn*group*7
               Private group id : 7777
               Private group owner:
```

To view the UDN payload sent from an AP to the controller, use this command:

```
Device# show wireless stats client detail | inc udn

Total udn payloads sent                           : 1
```

When mDNS gateway is enabled on the controller, the mDNS services are automatically filtered based on the user private network ID for all the clients on the WLANs where user private network is enabled.

To view the service instances of a private network, use this command:

```
Device# show mdns-sd cache udn 7777 detail

Name: _services._dns-sd._udp.local
  Type: PTR
  TTL: 4500
  WLAN: 2
  WLAN Name: mdns-psk
  VLAN: 16
  Client MAC: f4f9.51e2.a6a6
  AP Ethernet MAC: 002a.1087.d68a
  Remaining-Time: 4486
  Site-Tag: default-site-tag
  mDNS Service Policy: madhu-mDNS-Policy
  Overriding mDNS Service Policy: NO
  UDN-ID: 7777
  UDN-Status: Enabled
  Rdata: _airplay._tcp.local
.
.
.
```

To view the service instances that are learnt from a shared UDN ID, use this command:

```
Device# show mdns-sd cache udn shared
```

```
------------------------------------------------------------- PTR Records
-----------------------------------------------------------------
RECORD-NAME                                    TTL      TYPE      ID
CLIENT-MAC       RR-RECORD-DATA
=================================================================================================
9.1.1.7.5.D.E.F.F.F.6.C.7.E.2.1.0.0.0.0.0.0.0.0  4500     WLAN      2
10e7.c6d5.7119   HP10E7C6D57119-2860.local
_services._dns-sd._udp.local                   4500     WLAN      2
10e7.c6d5.7119   _ipps._tcp.local
_universal._sub._ipps._tcp.local               4500     WLAN      2
10e7.c6d5.7119   HP DeskJet 5000 series [D57119] (3127)._ipps._tcp.
_print._sub._ipps._tcp.local                   4500     WLAN      2
10e7.c6d5.7119   HP DeskJet 5000 series [D57119] (3127)._ipps._tcp.
_ePCL._sub._ipps._tcp.local                    4500     WLAN      2
10e7.c6d5.7119   HP DeskJet 5000 series [D57119] (3127)._ipps._tcp.
_ipps._tcp.local                               4500     WLAN      2
10e7.c6d5.7119   HP DeskJet 5000 series [D57119] (3127)._ipps._tcp.
_services._dns-sd._udp.local                   4500     WLAN      2
10e7.c6d5.7119   _ipp._tcp.local
_universal._sub._ipp._tcp.local                4500     WLAN      2
10e7.c6d5.7119   HP DeskJet 5000 series [D57119] (3127)._ipp._tcp.l
_print._sub._ipp._tcp.local                    4500     WLAN      2
10e7.c6d5.7119   HP DeskJet 5000 series [D57119] (3127)._ipp._tcp.l
_ePCL._sub._ipp._tcp.local                     4500     WLAN      2
10e7.c6d5.7119   HP DeskJet 5000 series [D57119] (3127)._ipp._tcp.l
_ipp._tcp.local                                4500     WLAN      2
10e7.c6d5.7119   HP DeskJet 5000 series [D57119] (3127)._ipp._tcp.l
.
.
.

------------------------------------------------------------- SRV Records
-----------------------------------------------------------------
RECORD-NAME                                    TTL      TYPE      ID
CLIENT-MAC       RR-RECORD-DATA
=================================================================================================
HP DeskJet 5000 series [D57119] (3127)._ipp._  4500     WLAN      2
10e7.c6d5.7119   0 0 631 HP10E7C6D57119-2860.local
HP DeskJet 5000 series [D57119] (3127)._http.  4500     WLAN      2
10e7.c6d5.7119   0 0 80 HP10E7C6D57119-2860.local
HP DeskJet 5000 series [D57119] (3127)._ipps.  4500     WLAN      2
10e7.c6d5.7119   0 0 631 HP10E7C6D57119-2860.local
HP DeskJet 5000 series [D57119] (3127)._uscan  4500     WLAN      2
10e7.c6d5.7119   0 0 8080 HP10E7C6D57119-2860.local
.
.
.
------------------------------------------------------------- A/AAAA Records
-----------------------------------------------------------------
RECORD-NAME                                    TTL      TYPE      ID
CLIENT-MAC       RR-RECORD-DATA
=================================================================================================
HP10E7C6D57119-2860.local                      4500     WLAN      2
10e7.c6d5.7119   8.16.16.99

------------------------------------------------------------- TXT Records
-----------------------------------------------------------------
RECORD-NAME                                    TTL      TYPE      ID
CLIENT-MAC       RR-RECORD-DATA
=================================================================================================
HP DeskJet 5000 series [D57119] (3127)._ipp._  4500     WLAN      2
10e7.c6d5.7119   [502]'txtvers=1''adminurl=http://HP10E7C6D57119-28
```

```
HPDeskJet 5000 series [D57119] (3127)._http.  4500     WLAN      2
10e7.c6d5.7119   [1]''
HP DeskJet 5000 series [D57119] (3127)._ipps.  4500     WLAN      2
10e7.c6d5.7119   [502]'txtvers=1''adminurl=http://HP10E7C6D57119-28
.
.
.
```

To view the multicast DNS (mDNS) Service Discovery cache detail, use this command:

```
Device# show mdns-sd cache detail

Name: _printer._tcp.local
  Type: PTR
  TTL: 4500
  VLAN: 21
  Client MAC: ace2.d3bc.047e
  Remaining-Time: 4383
  mDNS Service Policy: default-mdns-service-policy
  Rdata: HP OfficeJet Pro 8720 [BC047E] (2)._printer._tcp.local
```

222 Hotspot 2.0

Topics:

- Hotspot 2.0

Hotspot 2.0

Explains the Hotspot 2.0 feature that enables IEEE 802.11 devices to interwork with external networks.

Hotspot 2.0 is a network interworking feature that

- enables IEEE 802.11 devices to interwork with external networks
- provides network discovery and selection services, and
- allows mobile devices to join Wi-Fi networks automatically, including during roaming.

Hotspot 2.0 components

The Hotspot 2.0 feature has four distinct parts:

- Hotspot 2.0 Beacon Advertisement: Allows a mobile device to discover Hotspot 2.0-compatible and 802.11u-compatible WLANs.
- Access Network Query Protocol (ANQP) Queries: Sends queries about the networks from IEEE 802.11 devices, such as network type (private or public); connectivity type (local network, internet connection, and so on), or the network providers supported by a given network.
- Online Sign-up: Allows a mobile device to obtain credentials to authenticate itself with the Hotspot 2.0 or WLAN.
- Authentication and Session Management: Provides authentication (802.1x) and management of the STA session (session expiration, extension, and so on).

Hotspot 2.0, also known as HS2 and Wi-Fi Certified Passpoint, is based on the IEEE 802.11u and Wi-Fi Alliance Hotspot 2.0 standards. It seeks to provide better bandwidth and services-on-demand to end users.

The interworking service aids network discovery and selection, enabling information transfer from external networks. It provides information to the stations about the networks before association.

Interworking not only helps users within the home, enterprise, and public access domains, but also assists manufacturers and operators to provide common components and services for IEEE 802.11 customers. These services are configured on a per-WLAN basis on the Cisco Wireless Controller (controller).

In order to mark a WLAN as Hotspot 2.0-compatible, the 802.11u-mandated information element and the Hotspot 2.0 information element is added to the basic service set (BSS) beacon advertised by the corresponding AP, and in WLAN probe responses.

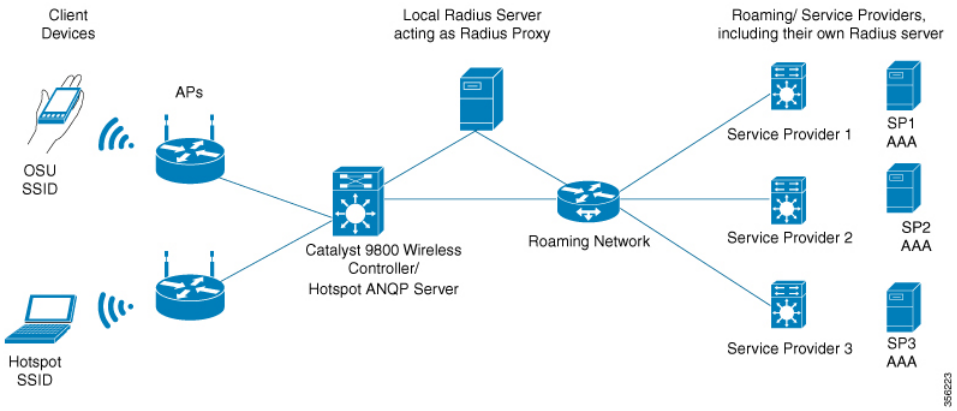
Note

The Hotspot 2.0 feature supports only local mode or FlexConnect mode (central switching and central authentication).

FlexConnect local switching is only supported when the Open Roaming configuration template is set up using the **wireless hotspot ANQP-server** *server-name* **type open-roaming** command. If the configuration diverges from this template, FlexConnect local switching will not be supported.

This figure shows a standard deployment of the Hotspot 2.0 network architecture:

Figure 40: Hotspot 2.0 Deployment Topology



Hotspot 2.0 enhancements

From Cisco IOS XE Amsterdam 17.3.1, the Hotspot 2.0 feature has been enhanced with these options:

- New ANQP elements:
  - Advice of charge: Provides information on the financial charges for using the SSID of the NAI realm
  - Operator icon metadata
  - Venue URL: Defines an optional URL for each of the configured venue names
- Introduction of Terms and Conditions: This requires a user to accept certain Terms and Conditions before being allowed internet access, after connecting to a Hotspot SSID.
- Integration of OSEN security and WPA2 security on the same SSID.

From Cisco IOS XE Amsterdam 17.3.1 onwards, two encryption methods are supported on a single SSID, namely WPA2 802.1x for Hotspot 2.0 and OSEN for online sign-up. Based on the type of encryption selected during client association, the client will be put on Hotspot 2.0 VLAN or online sign-up VLAN.

In WPA2 802.1x authentication, a client should match the credentials provisioned on a device. In online sign-up, a service provider WLAN is used by a client to perform online sign-up. For Hotspot 2.0 SSIDs, the RADIUS server enforces the terms and conditions before allowing internet connectivity to clients.

This release also supports OSEN-specific VLAN in a policy profile. If an OSEN VLAN is defined in a policy profile, OSEN clients are added to the VLAN. Otherwise, clients are added to the regular policy profile VLAN or to the default VLAN. If OSEN is enabled with WPA2 on an SSID, it is mandatory to define an OSEN VLAN in the policy profile. Otherwise, clients cannot join the VLAN.

In FlexConnect mode, if an OSEN VLAN is defined in a policy profile, the same VLAN needs to be added to the flex profile. Failing to do so excludes the clients from the VLAN.

Note

When Hotspot 2.0 is enabled in a WLAN, the Wi-Fi direct clients that support cross-connect feature should not be allowed to associate to the Hotspot 2.0 WLAN. To make sure this policy is enforced, ensure that the following configuration is in place:

```
wlan <wlan-name> <wlan-name> <ssid>
wifi-direct policy xconnect-not-allow
```

Restrictions

- Clients are excluded if an OSEN VLAN is not added to a flex profile.
- In FlexConnect mode, clients are excluded if an OSEN VLAN is not added in a flex profile.
- In FlexConnect deployments, the URL filter should reference an existing URL filter (configured using the **urlfilter list** *urlfilter-name* command). Otherwise, a client is added to the excluded list, after authentication.
- Only central authentication is supported.
- Fragmented ANQP replies are not synchronized to the standby controller in high-availability mode. Therefore, clients have to re-issue a query if there is a switchover.

Open roaming

Describes a configuration feature that enables mobile users to automatically and seamlessly roam across Wi-Fi and cellular networks.

Open roaming is a configuration feature that

- enables mobile users to automatically and seamlessly roam across Wi-Fi and cellular networks
- simplifies the task of setting up a Hotspot 2.0 ANQP server with a new configuration template, and
- is supported from Cisco IOS XE Amsterdam Release 17.2.1 on the controller.

Open roaming configuration options

When you configure open roaming, fixed ANQP parameters are automatically populated.

You can configure different identity types by defining roaming organizational identifiers. The organizational unique identifier (OUI) is a three-octet number that identifies the type of organizations available in a given roaming consortium. The OUI list determines the type of identities allowed to roam into the network. The default configuration allows all the identities on the access network. However, access networks can customize the Roaming Consortium Organization Identifier (RCOI) they advertise.

You can configure three types of policies for access networks:

- **Allow all**: Accepts users from any identity provider (IDP), with any privacy policy.
- **Real ID**: Accepts users from any IDP, but only with a privacy policy that shares real identity (anonymous not accepted).
- **Custom**: Accepts users of select identity types and privacy policies associated with the identity types; basically all the other RCOIs.

Users can select the following privacy modes:

- Anonymous
- Share real identity

The list of currently defined organizational identifiers and their aliases are given in the following table.

Table 194: Roaming organizational identifiers and aliases

Description	Roaming Organizational Identifier	WBA Value	Display Name
All	004096	5A03BA0000	All
All with real ID	00500b	5A03BA1000	All with real-ID only
All paid members	00500f	BAA2D00000	All paid
Device manufacturer all ID	00502a	5A03BA0A00	Device Manufacturer
Device manufacturer real ID only	0050a7	5A03BA1A00	Device Manufacturer real-ID
Cloud or Social ID	005014	5A03BA0200	Cloud ID
Cloud or Social real ID	0050bd	5A03BA1200	Cloud ID real-ID
Enterprise Employee ID	00503e	5A03BA0300	Enterprise ID
Enterprise Employee real ID	0050d1	5A03BA1300	Enterprise ID real ID
Enterprise Customer ID	005050	-	Enterprise Customer program ID
Enterprise Customer real ID	0050e2	-	Enterprise Customer program real ID
Loyalty Retail ID	005053	5A03BA0B00	Loyalty Retail
Loyalty Retail real ID	0050f0	5A03BA1B00	Loyalty Retail real ID
Loyalty Hospitality ID	005054	5A03BA0600	Loyalty Hospitality
Loyalty Hospitality real ID	00562b	5A03BA1600	Loyalty Hospitality real ID
SP free Bronze Qos	005073	5A03BA0100	SP free Bronze Qos
SP free Bronze Qos Real ID	0057D2	5A03BA1100	SP free Bronze Qos Real ID
SP paid Bronze QoS	-	BAA2D00100	SP paid Bronze QoS
SP paid Bronze QoS real ID	-	BAA2D01100	SP paid Bronze QoS real ID
SP paid Silver QoS	-	BAA2D02100	SP paid Silver QoS
SP paid Silver QoS real ID	-	BAA2D03100	SP paid Silver QoS real ID
SP paid Gold QoS	-	BAA2D04100	SP paid Gold QoS
SP paid Gold QoS real ID	-	BAA2D05100	SP paid Gold QoS real ID
Government ID free	-	5A03BA0400	Government ID free

Description	Roaming Organizational Identifier	WBA Value	Display Name
Automotive ID free	-	5A03BA0500	Automotive ID free
Automotive Paid	-	BAA2D00500	Automotive Paid
Education or Research ID free	-	5A03BA0800	Education or Research ID free
Cable ID free	-	5A03BA0900	Cable ID free

Configure Hotspot 2.0

Configure an access network query protocol server (CLI)

Configure an access network query protocol server.

Set up an Access Network Query Protocol (ANQP) server to define and advertise services offered by an AP at Wi-Fi Hotspot 2.0 locations.

The Access Network Query Protocol Server (ANQP) is a query and response protocol that defines the services offered by an AP, usually at a Wi-Fi Hotspot 2.0.

> **Note**
>
> When configuring roaming-OI in the ANQP server, ensure that you set the **beacon** keyword for at least one roaming-OI, as mandated by the 802.11u standard.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a Hotspot 2.0 ANQP server.

   Example

   ```
   Device(config)# wireless hotspot anqp-server server-name
   ```

   Example

   ```
   Device(config)# wireless hotspot anqp-server my_server
   ```

3. Add a description for the ANQP server.

Example

```
Device(config-wireless-anqp-server)# description description
```

Example

```
Device(config-wireless-anqp-server)# description "My Hotspot 2.0"
```

4. Configure a 802.11u Third Generation Partnership Project (3GPP) cellular network.

   Example

   ```
   Device(config-wireless-anqp-server)# 3gpp-info mobile-country-code
   mobile-network-code
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# 3gpp-info us mcc
   ```

   The *mobile-country-code* should be a 3-digit decimal number. The *mobile-network-code* should be a 2-digit or 3-digit decimal number.

5. Configure the ANQP reply fragmentation threshold, in bytes.

   Example

   ```
   Device(config-wireless-anqp-server)# anqp fragmentation-threshold
   threshold-value
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# anqp fragmentation-threshold 100
   ```

   The ANQP protocol can be customized by setting the fragmentation threshold, after which the ANQP reply is split into multiple messages.

   > **Note**
   >
   > We recommend that you use the default values for the deployment.

6. Configure the Hotspot 2.0 ANQP domain identifier.

   Example

   ```
   Device(config-wireless-anqp-server)# anqp-domain-id domain-id
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# anqp-domain-id 100
   ```

7. Configure the 802.11u network authentication type.

   Example

   ```
   Device(config-wireless-anqp-server)# authentication-type {dns-redirect |
   http-https-redirect | online-enrollment | terms-and-conditions}
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# authentication-type online-enrollment
   ```

   Depending on the authentication type, a URL is needed for HTTP and HTTPS.

8. Configure the Hotspot 2.0 protocol and port capabilities.

   Example

   ```
   Device(config-wireless-anqp-server)# connection-capability ip-protocol
   port-number {closed|open|unknown}
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# connection-capability 12 40 open
   ```

   Note

   Hotspot 2.0 specifications require that you predefine some open ports and protocols. Ensure that you meet these requirements in order to comply with the Hotspot 2.0 specifications. See the **connection-capability** command in the *Cisco Catalyst 9800 Series Wireless Controller Command Reference* document for a list of open ports and protocols.

9. Configure an 802.11u domain name.

   Example

   ```
   Device(config-wireless-anqp-server)# domain domain-name
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# domain my-domain
   ```

   You can configure up to 32 domain names. The *domain-name* should not exceed 220 characters.

10. Configure an 802.11u IPv4 address type in the Hotspot 2.0 network.

    Example

    ```
    Device(config-wireless-anqp-server)# ipv4-address-type ipv4-address-type
    ```

Example

```
Device(config-wireless-anqp-server)# ipv4-address-type public
```

11. Configure an 802.11u IPv6 address type in the Hotspot 2.0 network.

Example

```
Device(config-wireless-anqp-server)# ipv6-address-type ipv6-address-type
```

Example

```
Device(config-wireless-anqp-server)# ipv6-address-type available
```

12. Configure an 802.11u NAI realm profile that identifies the realm that is accessible using the AP.

Example

```
Device(config-wireless-anqp-server)# nai-realm realm-name
```

Example

```
Device(config-wireless-anqp-server)# nai cisco.com
```

13. Configure a Hotspot 2.0-operating class identifier.

Example

```
Device(config-wireless-anqp-server)# operating-class class-id
```

Example

```
Device(config-wireless-anqp-server)# operating-class 25
```

14. Configure a Hotspot 2.0 operator-friendly name in a given language.

Example

```
Device(config-wireless-anqp-server)# operator operator-name language-code
```

Example

```
Device(config-wireless-anqp-server)# operator XYZ-operator eng
```

Use only the first three letters of the language, in lower case, for the language code. For example, use *eng* for English.

To see the full list of language codes, go to: http://www.loc.gov/standards/iso639-2/php/code_list.php.

> **Note**
>
> You can configure only one operator per language.

15. Configure the SSID that wireless clients will use for OSU.

Example

```
Device(config-wireless-anqp-server)# osu-ssid SSID
```

Example

```
Device(config-wireless-anqp-server)# osu-ssid test
```

The SSID length can be up to 32 characters.

16. Configure the 802.11u roaming organization identifier.

Example

```
Device(config-wireless-anqp-server)# roaming-oi OI-value [beacon]
```

Example

```
Device(config-wireless-anqp-server)# roaming-oi 24 beacon
```

If the **beacon** keyword is specified, the roaming OUI is advertised in the AP WLAN beacon or probe response. Otherwise, it will only be returned while performing the roaming OUI ANQP query.

> **Note**
>
> The hex string of a roaming OUI should contain only lowercase letters.

17. Configure the 802.11u venue information.

Example

```
Device(config-wireless-anqp-server)# venue venue-name language-code
```

Example

```
Device(config-wireless-anqp-server)# venue bank eng
```

The *venue-name* should not exceed 220 characters and the *language-code* should only be 2 or 3 lowercase letters (a-z) in length.

The ANQP server is configured and will advertise the specified network services and capabilities to wireless clients performing 802.11u queries at the Hotspot 2.0 location.

Configure ANQP global server settings (GUI)

Configure ANQP global server settings using the GUI interface.

Configure ANQP (Access Network Query Protocol) global server settings to define network parameters and connection types for wireless hotspot services.

Use this procedure to configure global server settings for ANQP, which enables wireless clients to discover network services and connection information before associating with the access point.

Procedure

1. Choose **Configuration > Wireless > Hotspot/OpenRoaming**.
2. Select an existing server from the list of servers.
3. Click the **Server Settings** tab.
4. Go to the **Global Server Settings** section.
5. From the **IPv4 Type** drop-down list, choose an IPv4 type.
6. From the **IPv6 Type** drop-down list, choose an IPv6 type.
7. In the **OSU SSID** field, enter the SSID that wireless clients will use for Online Sign-Up (OSU).
8. Click the **Show Advanced Configuration** link to view the advanced options.
   - In the **Fragmentation Threshold (bytes)** field, enter the fragmentation threshold.

     Note

     Packets that are larger than the size you specify here will be fragmented.

   - In the **GAS Request Timeout (ms)** field, enter the number of Generic Advertisement Services (GAS) request action frames sent that can be sent to the controller by an AP in a given interval.
9. Click **Apply to Device**.

The ANQP global server settings are configured and applied to the device. Wireless clients can now query the access point for network information using the configured parameters.

Configure open roaming (CLI)

Configure open roaming using commands.

Set up a Hotspot 2.0 ANQP server with open roaming configuration to enable seamless wireless connectivity across participating networks.

The new configuration template of the open roaming ANQP server simplifies the task of setting up a Hotspot 2.0 ANQP server. When you configure open roaming using this template, default ANQP parameters are automatically populated. The default values defined in the template always override any user-defined configuration values.

For example, these are the default values enforced with the type open-roaming template:

- nai-realm open.openroaming.org

- eap-method eap-tls
- eap-method eap-ttls
- inner-auth-non-eap mschap-v2
- inner-auth-non-eap pap
- eap-method eap-aka

You can add more fields to the existing template, but ensure that they do not overlap with the existing default values. Also, if you change any of these default values, you will need to re-configure every time you enter in ANQP type open-roaming config.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a Hotspot 2.0 ANQP server with open roaming.

   Example

   ```
   Device(config)# wireless hotspot anqp-server server-name type open-roaming
   ```

   Example

   ```
   Device(config)# wireless hotspot anqp-server my-server type open-roaming
   ```

3. Set the open roaming element alias.

   Example

   ```
   Device(config-wireless-anqp-server)# open-roaming-oi alias
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# open-roaming-oi allow-all
   ```

4. Configure a preferred domain name to ensure that clients roam into a preferred network.

   Example

   ```
   Device(config)# domain domain-name
   ```

   Example

   ```
   Device(config)# domain my-domain
   ```

You can configure up to 32 domain names. The domain-name should not exceed 220 characters.

The open roaming ANQP server is configured with the specified settings, enabling clients to seamlessly roam across participating open roaming networks.

Configure open roaming (GUI)

Configure open roaming using GUI.

Configure an ANQP server as an open roaming server to enable seamless connectivity for users across different networks without requiring separate authentication credentials.

Use this procedure when you need to set up open roaming functionality through the GUI interface to allow automatic network access for compatible devices.

Procedure

---

1. Choose **Configuration > Wireless > Hotspot/OpenRoaming**.
2. Click **Add**.

   The **Add New ANQP Server** window is displayed.
3. In the **Name** field, enter a name for the server.
4. In the **Description** field, enter a description for the server.
5. Check the **OpenRoaming Server** check box to use the server as an open roaming server.

   > **Note**
   >
   > You can set the server as an open roaming server only at the time of server creation.

6. Check the **Internet Access** check box to enable internet access for the server.
7. From the **Network Type** drop-down list, choose the network type.
8. Click **Apply to Device**.

The ANQP server is configured as an open roaming server and applied to the device, enabling seamless network access for compatible client devices.

Configure NAI realms (GUI)

Configure NAI realms using the GUI to define authentication parameters for OpenRoaming networks.

Configure NAI realms to define Network Access Identifier realms for OpenRoaming authentication, specifying supported EAP methods and credential types for wireless clients.

NAI realms are used in 802.11u networks to specify authentication parameters and supported EAP methods for network access. This configuration is essential for OpenRoaming implementations where clients need to understand available authentication options.

Procedure

---

1. Choose **Configuration > Wireless > Hotspot/OpenRoaming**.

2. Select an existing server from the list of servers.
3. Go to the **NAI Realms** section.
4. Click **Add**.

   The **Add NAI Realm** window is displayed.

5. In the **NAI Realm Name** field, enter an 802.11u NAI realm of the OSU operator.
6. In the **EAP Methods** section, use the toggle button to enable the required EAP methods.

   After an EAP method is enabled, a pane is displayed to configure the details. Users are shown a configuration section where they can enable *credential*, *inner-auth-EAP*, *inner-auth-non-EAP*, *tunneled-EAP-credential*. The user can select multiple options for each of the configuration.

   - The **Credential** window has options such as certificate, hw-token, nfc, none, sim, softoken, username-password, and usim. Check the corresponding check box.
   - The **inner-auth-EAP** window has options such as EAP-aka, EAP-fast, EAP-sim, EAP-tls, EAP-ttls, EAP-leap, and EAP-peap. Check the corresponding check box.
   - The **inner-auth-EAP** window has options such as EAP-aka, EAP-fast, EAP-sim, EAP-tls, EAP-ttls, EAP-leap, and EAP-peap. Check the corresponding check box.
   - The **tunneled-EAP-credential** window has options such as anonymous, certificate, hw-token, nfc, sim, softoken, username-password, and usim. Check the corresponding check box.
   - Click **Save**.

7. Click **Apply to Device**.

The NAI realm is configured with the specified authentication methods and applied to the device, enabling OpenRoaming clients to identify supported authentication options for network access.

Configure organizational identifier alias (GUI)

Configure organizational identifier alias using the GUI to enable roaming organization identifiers and domain names for hotspot and OpenRoaming functionality.

Configure organizational identifier alias to enable proper roaming functionality and domain management for wireless hotspot and OpenRoaming services.

Use this procedure when you need to set up roaming organization identifiers and domain names for wireless hotspot or OpenRoaming configurations through the graphical user interface.

Procedure

---

1. Choose **Configuration > Wireless > Hotspot/OpenRoaming**.
2. Select an existing server from the list of servers.
3. In the **Roaming OIs** area, enter an 802.11u roaming organization identifier in the **Roaming OI** field.
4. Check the **Beacon State** check box to enable the beacon.

   If the beacon is specified, the roaming OUI is advertised in the AP WLAN beacon or probe response. Otherwise, it will only be returned while performing the roaming OUI ANQP query.

> **Note**
>
> Only three OUIs can be enabled in the beacon state.

5. Click **Add** to add a roaming OI.
6. In the **Available OpenRoaming OI** window, select an organizational identifier and click the right arrow to add an **OpenRoaming OI**.
7. In the **Domains** area, enter an 802.11u domain name in the **Domain Name** field.
8. Click **Add** to use the domain name that you have entered as the preferred domain.
9. Click **Apply to Device**.

The organizational identifier alias is configured with the specified roaming OIs and domain names. The configuration is applied to the device and roaming functionality is enabled for the wireless hotspot or OpenRoaming service.

Configure WAN metrics (GUI)

Configure WAN metrics using the GUI to set downlink and uplink parameters for hotspot servers.

Configure WAN metrics to define network performance parameters including downlink load, downlink speed, upload load, upload speed, and link status for hotspot servers.

WAN metrics configuration allows you to specify network performance characteristics that affect how the hotspot server manages wireless connections and traffic flow.

Procedure

---

1. Choose **Configuration > Wireless > Hotspot/OpenRoaming**.
2. Select an existing server from the list of servers.
3. Click the **Server Settings** tab.
4. Go to the **WAN Metrics** area.
5. In the **Downlink Load** field, enter the WAN downlink load.
6. In the **Downlink Speed (kbps)** field, enter the WAN downlink speed, in kbps.
7. In the **Load Duration (100ms)** field, enter the load duration.
8. In the **Upload Load** field, enter the WAN upload load.
9. In the **Upload Speed (kbps)** field, enter the WAN upload speed, in kbps.
10. From the **Link Status** drop-down list, choose the link status.
11. Use the **Full Capacity Link** toggle button to enable the WAN link to operate at its maximum capacity.
12. Click **Apply to Device**.

The WAN metrics configuration is applied to the selected hotspot server, and the network performance parameters are updated according to your specified settings.

Configure WAN metrics

Configure WAN metrics using commands.

Set up Wide Area Network (WAN) parameters such as uplink and downlink speed, link status, and load for Hotspot 2.0 ANQP server configuration.

This procedure shows you how to configure the Wide Area Network (WAN) parameters such as uplink and downlink speed, link status, load, and so on.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a Hotspot 2.0 ANQP server.

   Example

   ```
   Device(config)# wireless hotspot anqp-server server-name
   ```

   Example

   ```
   Device(config)# wireless hotspot anqp-server my_server
   ```

3. Configure the WAN downlink load.

   Example

   ```
   Device(config-wireless-anqp-server)# wan-metrics downlink-load load-value
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# wan-metrics downlink-load 100
   ```

4. Configure the WAN downlink speed, in kbps.

   Example

   ```
   Device(config-wireless-anqp-server)# wan-metrics downlink-speed speed
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# wan-metrics downlink-speed 1000
   ```

5. Configure the WAN link to operate at its maximum capacity.

   Example

   ```
   Device(config-wireless-anqp-server)# wan-metrics full-capacity-link
   ```

6. Set the WAN link status.

Example

```
Device(config-wireless-anqp-server)# wan-metrics link-status
{down|not-configured|test-state|up}
```

Example

```
Device(config-wireless-anqp-server)# wan-metrics link-status down
```

7. Configure the uplink or downlink load measurement duration.

Example

```
Device(config-wireless-anqp-server)# wan-metrics load-measurement-duration
duration
```

Example

```
Device(config-wireless-anqp-server)# wan-metrics
load-measurement-duration 100
```

8. Configure the WAN uplink load.

Example

```
Device(config-wireless-anqp-server)# wan-metrics uplink-load load-value
```

Example

```
Device(config-wireless-anqp-server)# wan-metrics uplink-load 100
```

9. Configure the WAN uplink speed, in kbps.

Example

```
Device(config-wireless-anqp-server)# wan-metrics uplink-speed speed
```

Example

```
Device(config-wireless-anqp-server)# wan-metrics uplink-speed 1000
```

The WAN metrics are now configured for the Hotspot 2.0 ANQP server with the specified parameters for network performance monitoring and reporting.

Configure beacon parameters (GUI)

Configure beacon parameters using the GUI interface to set up hotspot and OpenRoaming settings for wireless networks.

This task allows you to configure beacon parameters for hotspot and OpenRoaming functionality on wireless networks, including setting the homogenous extended service set identifier, domain identifier, and venue information.

Use this procedure when you need to establish beacon parameters for wireless hotspot services or OpenRoaming configurations through the web-based GUI interface.

Procedure

1. Choose **Configuration > Wireless > Hotspot/OpenRoaming**.
2. Select an existing server from the list of servers.
3. Click **Server Settings** tab.
4. Go to the **Beacon Parameters** section.
5. In the **Hess ID** field, enter the homogenous extended service set identifier. The Hess ID can be either in *xx:xx:xx:xx:xx:xx*, *xx-xx-xx-xx-xx-xx*, or *xxxx.xxxx.xxxx* format.
6. In the **Domain ID** field, enter the domain's identifier.
7. From the **Venue Type** drop-down list, select the venue.

   Choosing a venue activates the subvenue type.
8. From the **subvenue-type** drop-down list, select the sub-venue.
9. Click **Apply to Device**.

The beacon parameters are configured and applied to the device. The wireless network will now broadcast the specified hotspot and OpenRoaming configuration settings.

Configure authentication and venue (GUI)

Configure authentication and venue using the GUI.

Configure authentication types, venue details, and connection capabilities for Hotspot/OpenRoaming services through the wireless controller GUI.

Use this procedure to set up authentication methods, venue information, and connection capabilities for wireless hotspot services. This configuration ensures compliance with Hotspot 2.0 specifications and provides proper authentication and venue identification for wireless clients.

Procedure

1. Choose **Configuration > Wireless > Hotspot/OpenRoaming**.
2. Select an existing server from the list of servers.
3. Click the **Authentication/Venue** tab.
4. Under the **Network Auth Types** section, check the **DNS Redirect**, **Online Enrolment**, **HTTP/HTTPS Redirect**, **Terms and Conditions** check boxes.

   For **HTTP/HTTPS Redirect** and **Terms and Conditions**, the URL field is enabled after selecting them.
5. Add the URL for the corresponding authentication type.
6. Click **Apply**.
7. Go to the **Venues** section and click **Add**.

   The **Venue Details** pane is displayed.
8. In the **Language Code** field, enter the language code.

   Use the first two or three letters of the language, in lower case, for the language code. For example, use *eng* for English. To see the full list of language codes, go to: http://www.loc.gov/standards/iso639-2/php/code_list.php.
9. In the **Venue URL** field, enter the URL of the venue.

10. In the **Venue Name** field, enter the name of the venue.
11. Click check mark icon to add the venue details.
12. Go to the **Connection Capability** section and click **Add**.

    The **Connection Capabilities** pane is displayed. See the **connection-capability** command in the *Cisco Catalyst 9800 Series Wireless Controller Command Reference* document for a list of open ports and protocols.

13. In the **Port Number** field, enter the port number.
14. From the **Connection Status** drop-down list, choose a connection status.
15. In the **IP Protocol** field, enter the IP protocol number.

    Hotspot 2.0 specifications require that you predefine some open ports and protocols. Ensure that you meet these requirements in order to comply with the Hotspot 2.0 specifications. See the **connection-capability** command in the *Cisco Catalyst 9800 Series Wireless Controller Command Reference* document for a list of open ports and protocols.

16. Click the check mark icon to add the connection details.
17. Click **Apply to Device**.

The authentication types, venue details, and connection capabilities are configured and applied to the wireless controller. The hotspot service is now configured with the specified authentication methods and venue information.

Configure 3GPP/operator (GUI)

Configure 3GPP/operator using the GUI interface.

This task allows you to configure 3GPP cellular networks and Hotspot 2.0 operators for wireless hotspot and OpenRoaming functionality.

Use this configuration when you need to set up 3GPP cellular network information and operator details for wireless hotspot services. This configuration includes operating class indicators, mobile country and network codes, and operator language settings.

Procedure

---

1. Choose **Configuration > Wireless > Hotspot/OpenRoaming**.
2. Select an existing server from the list of servers.
3. Go to the **3GPP/Operator** tab.
4. In the **Operating Class Indicator** field, enter the operating class identifier and click the + icon.

   The operating class identifier is added and displayed in the pane below. Use the delete icon to delete them, if required.

   > **Note**
   >
   > Class IDs should be in the following ranges: 81-87, 94-96, 101-130, 180, and 192-254.

5. Go to the **3GPP Cellular Networks** section and click **Add**.

   The **3GPP Network Details** pane is displayed.

6. In the **Mobile Country Code (MCC)** field, enter the mobile country code, which should be a 3-digit decimal number.

7. In the **Mobile Network Code (MNC)** field, enter the mobile network code, which should be a 2 or 3-digit decimal number.

   For the list of Mobile Country Codes (MCC) and Mobile Network Codes (MNC), see the following links: https://www.itu.int/pub/T-SP-E.212B-2018 or https://www.MCC-MNC.com.

8. Click check mark icon to add the network details.
9. Go to the **Hotspot 2.0 Operators** section and click **Add**.

   The **Operator Details** pane is displayed.

10. In the **Language Code** field, enter the language code.

    Use only the first three letters of the language, in lower case, for the language code. For example, use *eng* for English. To see the full list of language codes, go to: http://www.loc.gov/standards/iso639-2/php/code_list.php.

11. In the **Name** field, enter the name of the OSU operator.
12. Click check mark icon to add the operator details.
13. Click **Apply to Device**.

The 3GPP/operator configuration is applied to the device. The configured operating class indicators, cellular network details, and operator information are now active for wireless hotspot and OpenRoaming services.

Configure OSU provider (GUI)

Configure OSU provider using GUI.

Configure an OSU (Online Sign-Up) provider to enable wireless clients to connect and provision network access through the Hotspot/OpenRoaming service.

Use this procedure to set up an OSU provider with the necessary configuration parameters including provider name, NAI realm, authentication methods, server URI, icon configuration, and friendly names for different languages.

Before you begin

Follow these steps to configure OSU provider using GUI:

Procedure

---

1. Choose **Configuration > Wireless > Hotspot/OpenRoaming**.
2. Select an existing server from the list of servers.
3. Go to the **OSU Provider** tab.
4. Click **Add**.

   The **General Config** pane is displayed.

5. In the **Provider Name** field, enter the OSU provider name.
6. In the **NAI Realm** field, enter the Network Access Identifier (NAI) realm of the OSU operator.
7. From the **Primary Method** drop-down list, choose the primary supported OSU method of the OSU operator.

   This activates the **Secondary Method** drop-down list. If you choose *None* as the primary supported OSU method, you will not get the secondary method.

8. (Optional) From the **Secondary Method** drop-down list, choose the secondary supported OSU method of the OSU operator.
9. In the **Server URI** field, enter the server Uniform Resource Identifier (URI) of the OSU operator.
10. Click **Icon Config** tab.

11. Click **Add**.
12. From the **Icon Name** drop-down list, choose the icon name.
13. Click **Save**.
14. Click **Friendly Names** tab.
15. Click **Add**.
16. In the **Language** field, enter the language code.
17. In the **Name** field, enter the name of the OSU operator.
18. In the **Description** field, enter the description for the OSU operator.
19. Click **Save**.
20. Click the check mark icon to save.
21. Click **Apply to Device**.

The OSU provider is configured and applied to the device, enabling wireless clients to use the Online Sign-Up service for network provisioning.

Configure OSU provider (CLI)

Configure OSU provider using CLI.

Configure an OSU (Online Sign-Up) provider to enable wireless clients to connect and provision network access through the Hotspot/OpenRoaming service.

Use this procedure to set up an OSU provider with the necessary configuration parameters including provider name, NAI realm, authentication methods, server URI, icon configuration, and friendly names for different languages.

Before you begin

Follow these steps to configure OSU provider using CLI:

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure an icon for Hotspot 2.0 and its parameters, such as media type, language code, icon width, and icon height.

   Example

   ```
   Device(config)# wireless hotspot icon bootflash:system-file-name media-type
   language-code icon-width icon-height
   ```

   ```
   Device(config)# wireless hotspot bootflash:logo1 image eng 100 200
   ```

3. Configure a Hotspot 2.0 ANQP server.

Example

```
Device(config)# wireless hotspot anqp-server server-name
```

```
Device(config)# wireless hotspot anqp-server my_server
```

4. Configure a Hotspot 2.0 OSU provider name.

   Example

   ```
   Device(config-wireless-anqp-server)# osu-provider osu-provider-name
   ```

   ```
   Device(config-wireless-anqp-server)# osu-provider my-osu
   ```

5. Configure the name of the OSU operator in a given language.

   Example

   ```
   Device(config-anqp-osu-provider)# name osu-operator-name lang-code description
   ```

   ```
   Device(config-anqp-osu-provider)# name xyz-oper eng xyz-operator
   ```

   The OSU operator name and description should not exceed 220 characters. The language code should be 2 or 3 lower-case letters (a-z).

6. Configure the server Uniform Resource Identifier (URI) of the OSU operator.

   Example

   ```
   Device(config-anqp-osu-provider)# server-uri server-uri
   ```

   ```
   Device(config-anqp-osu-provider)# server-uri cisco.com
   ```

7. Configure the primary supported OSU method of the OSU operator.

   Example

   ```
   Device(config-anqp-osu-provider)# method {oma-dm | soap-xml-spp}
   ```

   ```
   Device(config-anqp-osu-provider)# method oma-dm
   ```

8. Configures the Network Access Identifier (NAI) realm of the OSU operator.

   Example

   ```
   Device(config-anqp-osu-provider)# nai-realm nai-realm
   ```

   ```
   Device(config-anqp-osu-provider)# nai-realm cisco.com
   ```

   The *nai-realm* should not exceed 220 characters.

9. Configure the icon for the OSU provider.

Example

```
Device(config-anqp-osu-provider)# icon file-name
```

```
Device(config-anqp-osu-provider)# icon xyz.jpeg
```

The *file-name* should not exceed 100 characters.

The OSU provider is configured and applied to the device, enabling wireless clients to use the Online Sign-Up service for network provisioning.

Configure Hotspot 2.0 WLAN

Configure Hotspot 2.0 WLAN.

Set up a Hotspot 2.0 enabled WLAN to provide standardized Wi-Fi access for mobile devices.

Hotspot 2.0 WLANs enable automatic network discovery and connection for compatible devices, providing a seamless wireless experience in public and enterprise environments.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a WLAN and enter WLAN configuration mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id ssid
   ```

   Example

   ```
   Device(config)# wlan hs2 1 hs2
   ```

3. Configure random GTK for hole 196 mitigation.

   Example

   ```
   Device(config-wlan)# security wpa
   wpa2 gtk-randomize
   ```

   Hole 196 is the name of WPA2 vulnerability.

4. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

The Hotspot 2.0 WLAN is configured and enabled, allowing compatible devices to automatically discover and connect to the network.

Configure an online subscription with encryption WLAN (CLI)

Configure an online subscription with encryption WLAN.

Enable secure onboarding for Hotspot 2.0 networks by configuring an OSEN WLAN to obtain necessary credentials.

Online subscription with Encryption (OSEN) WLAN is used to onboard a Hotspot 2.0 network (to get the necessary credentials) in a secure manner.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a WLAN and enter WLAN configuration mode.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id ssid
   ```

   Example

   ```
   Device(config)# wlan hs2 1 hs2
   ```

3. Enable WPA OSEN security support.

   Example

   ```
   Device(config-wlan)# security wpa osen
   ```

   Note

   OSEN and robust security network (RSN) are mutually exclusive. If RSN is enabled on a WLAN, OSEN cannot be enabled on the same WLAN.

4. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

The OSEN WLAN is configured and enabled, allowing secure onboarding for Hotspot 2.0 network credentials. Note that you cannot apply a policy profile to the OSEN WLAN if a Hotspot 2.0 server is enabled on the WLAN.

Attach an ANQP server to a policy profile (CLI)

Configure an ANQP server attachment to a policy profile for Hotspot 2.0 functionality.

Enable Hotspot 2.0 functionality by attaching an ANQP server to a wireless policy profile.

ANQP (Access Network Query Protocol) servers provide network information to Hotspot 2.0 clients during the discovery and authentication process. This configuration is required for implementing Hotspot 2.0 services.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a policy profile.

   Example

   ```
   Device(config)# wireless profile policy policy-profile-name
   ```

   Example

   ```
   Device(config)# wireless profile policy policy-hotspot
   ```

3. Disable the policy profile.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

4. Attach the Hotspot 2.0 ANQP server to the policy profile.

   Example

   ```
   Device(config-wireless-policy)# hotspot anqp-server server-name
   ```

   Example

   ```
   Device(config-wireless-policy)# hotspot anqp-server my-server
   ```

5. Enable the policy profile.

Example

```
Device(config-wireless-policy)# no shutdown
```

The ANQP server is successfully attached to the policy profile. You must then attach the policy profile to the WLAN to make the WLAN Hotspot 2.0 enabled.

What to do next
Attach the policy profile to the WLAN to make the WLAN Hotspot 2.0 enabled.

Configure interworking for Hotspot 2.0

Configure interworking for Hotspot 2.0.

Enable Hotspot 2.0 interworking capabilities to provide seamless wireless network access for mobile devices.

Hotspot 2.0 interworking allows mobile devices to automatically discover and connect to Wi-Fi networks that provide internet access. This configuration sets up the ANQP (Access Network Query Protocol) server and defines network characteristics for Hotspot 2.0 compliance.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a Hotspot 2.0 ANQP server.

   Example

   ```
   Device(config)# wireless hotspot anqp-server server-name
   ```

   Example

   ```
   Device(config)# wireless hotspot anqp-server my_server
   ```

3. Configure a 802.11u network type.

   Example

   ```
   Device(config-wireless-anqp-server)# network-type network-type internet-access
    access-level
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# network-type guest-private
   internet-access allowed
   ```

   The access-level can be allowed or forbidden.

4. Optional: Configure a homogenous extended service set.

Example

```
Device(config-wireless-anqp-server)# hessid HESSID-value
```

Example

```
Device(config-wireless-anqp-server)# hessid 12.13.14
```

5. Select a group type and venue type from the list of available options.

Example

```
Device(config-wireless-anqp-server)# group venue-group venue-type
```

Example

```
Device(config-wireless-anqp-server)# group business bank
```

Hotspot 2.0 interworking is now configured with the specified ANQP server, network type, and venue information, enabling automatic network discovery and connection for compatible mobile devices.

Configure the Generic Advertisement Service rate limit (CLI)

Configure the generic advertisement service rate limit.

Control the rate of Generic Advertisement Services (GAS) request action frames to prevent network congestion and optimize performance.

GAS rate limiting helps manage the volume of hotspot service requests processed by the controller and APs, ensuring stable network performance in high-density environments.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure an AP profile and enter AP profile configuration mode.

Example

```
Device(config)# ap profile profile-name
```

Example

```
Device(config)# ap profile hs2-profile
```

3. Configure the number of Generic Advertisement Services (GAS) request action frames sent to the controller by an AP in a given interval.

Example

```
Device(config-ap-profile)# gas-ap-rate-limit request-number interval
```

Example

```
Device(config-ap-profile)# gas-ap-rate-limit 20 120
```

4. Return to global configuration mode.

Example

```
Device(config-ap-profile)# exit
```

5. Configure the number of GAS request action frames to be processed by the controller.

Example

```
Device(config)# wireless hotspot gas-rate-limit gas-requests-to-process
```

Example

```
Device(config)# wireless hotspot gas-rate-limit 100
```

The GAS rate limiting is configured to control the processing of hotspot service requests at both the AP and controller levels.

Configure global settings (GUI)

Configure global settings for wireless hotspot and OpenRoaming functionality.

Configure global settings to manage GAS rate limits and icon configurations for wireless hotspot and OpenRoaming services.

Use this task to set up global parameters that control the behavior of wireless hotspot and OpenRoaming features across your network.

Procedure

---

1. Choose **Configuration > Wireless > Hotspot/OpenRoaming > Global Settings**.
2. In the **GAS Rate Limit (Requests per sec)** field, enter the number of GAS request action frames to be processed by the controller.
3. Go to the **Icons Configuration** area.
4. Click **Add**.

   The **Add Global Icon** window is displayed.

5. From the **System Path** drop-down list, choose the path.
6. In the **Icon Name** field, enter the icon name.
7. In the **Icon Type** field, enter the icon type.
8. In the **Language Code** field, enter the language code.
9. In the **Icon Height** field, enter the icon height.

10. In the **Icon Width** field, enter the icon width.
11. Click **Apply to Device**.

The global settings for wireless hotspot and OpenRoaming are configured with the specified GAS rate limit and icon parameters applied to the device.

Configure advice of charge

Configure advice of charge information for using the SSID of the Network Access Identifier (NAI) realm.

Configure advice of charge to provide information on financial charges for using the SSID of the NAI realm in Hotspot 2.0 networks.

Advice of charge provides clients with information about the financial costs associated with using a particular SSID or NAI realm. This configuration is used in Hotspot 2.0 deployments where operators need to inform users about data usage charges.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a Hotspot 2.0 ANQP server.

   Example

   ```
   Device(config)# wireless hotspot anqp-server server-name
   ```

   Example

   ```
   Device(config)# wireless hotspot anqp-server my_server
   ```

3. Configure advice of charge for data usage.

   Example

   ```
   Device(config-wireless-anqp-server)# advice-charge type
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# advice-charge data
   ```

   Advice of charge provides information on the financial charges for using the SSID of the NAI realm.

4. Configure advice of charge information, which includes language, currency, and plan information.

   Example

   ```
   Device(config-anqp-advice-charge)# plan language currency info plan-info-file
   ```

Example

```
Device(config-anqp-advice-charge)# plan eng eur info bootflash:plan_eng.xml
```

Note

You can configure up to 32 plans.

5. Configure NAI realm for this advice of charge.

Example

```
Device(config-anqp-advice-charge)# nai-realm nai-realm
```

Example

```
Device(config-anqp-advice-charge)# nai-realm cisco
```

Note

You can configure up to 32 realms.

The advice of charge is now configured with the specified language, currency, plan information, and NAI realm, providing clients with financial charge information for network usage.

Configure terms and conditions (CLI)

Configure terms and conditions for Hotspot 2.0 ANQP server.

Set up terms and conditions configuration for wireless hotspot clients to ensure compliance and proper authentication.

Terms and conditions configuration is required for Hotspot 2.0 deployments where clients must acknowledge specific terms before accessing network resources. This configuration includes specifying the terms file, timestamp, and associated URL filtering policies.

Before you begin

Define a URL filter list, as shown in this example:

```
urlfilter list url-filter-name
    action permit
    filter-type post-authentication
    url allow-url
```

For information on configuring an URL list, see the *Defining URL Filter List* section.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a Hotspot 2.0 ANQP server.

   Example

   ```
   Device(config)# wireless hotspot anqp-server server-name
   ```

   Example

   ```
   Device(config)# wireless hotspot anqp-server my_server
   ```

3. Configure the terms and conditions filename for the clients.

   Example

   ```
   Device(config-wireless-anqp-server)# terms-conditions filename file-name
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# terms-conditions filename xyz-file
   ```

4. Configure the terms and conditions timestamp.

   Example

   ```
   Device(config-wireless-anqp-server)# terms-conditions timestamp date time
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# terms-conditions timestamp 2020-02-20
   20:20:20
   ```

5. Configure the terms and conditions URL filter list name.

   Example

   ```
   Device(config-wireless-anqp-server)# terms-conditions urlfilter list
   url-filter-list
   ```

   Example

   ```
   Device(config-wireless-anqp-server)# terms-conditions urlfilter list filter-yy
   ```

The terms and conditions are now configured for the Hotspot 2.0 ANQP server, enabling proper client authentication and compliance verification.

Define ACL and URL filter in AP for FlexConnect

Configure ACL and URL filter policies in AP for FlexConnect to control network access and web filtering for wireless clients.

Configure ACL and URL filter policies for FlexConnect access points to provide network access control and web content filtering for wireless clients.

FlexConnect allows access points to locally switch wireless traffic and apply policies when the connection to the wireless LAN controller is limited or unavailable. This configuration defines access control lists and URL filters that will be applied at the access point level.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Define an extended UDP access list and set access conditions for BOOTP client to server traffic.

   Example

   ```
   Device(config-ext-nacl)# sequence-number permit udp any eq bootpc any eq
   bootps
   ```

   Example

   ```
   Device(config-ext-nacl)# 10 permit udp any eq bootpc any eq bootps
   ```

   Sets access conditions to match only the packets on a given port number of bootstrap protocol (BOOTP) clients from any source host to match only the packets on a given port number of the bootstrap protocol (BOOTP) server of a destination host.

3. Define an extended UDP access list for BOOTP server to client traffic.

   Example

   ```
   Device(config-ext-nacl)# sequence-number permit udp any eq bootps any eq
   bootpc
   ```

   Example

   ```
   Device(config-ext-nacl)# 20 permit udp any eq bootps any eq bootpc
   ```

   Sets access conditions to match only the packets on a given port number of bootstrap protocol (BOOTP) server from any source host to match only the packets of a given port number of the bootstrap protocol (BOOTP) clients of a destination host.

4. Define an extended UDP access list for DNS traffic.

Example

```
Device(config-ext-nacl)# sequence-number permit udp any eq domain any eq
domain
```

Example

```
Device(config-ext-nacl)# 30 permit udp any eq domain any eq domain
```

Sets access conditions to match a destination host Domain Name Service (DNS) with only the packets from a given port number of the source DNS.

5. Define an extended IP access list to permit traffic from any source to a specific destination host.

Example

```
Device(config-ext-nacl)# sequence-number permit ip any host dest-address
```

Example

```
Device(config-ext-nacl)# 40 permit ip any host 10.10.10.8
```

Forwards packets from a source host to a single destination host.

6. Define an extended IP access list to permit traffic from a specific source host to any destination.

Example

```
Device(config-ext-nacl)# sequence-number permit ip host dest-address any
```

Example

```
Device(config-ext-nacl)# 50 permit ip host 10.10.10.8 any
```

Forwards packets from a single source host to a destination host.

7. Exit access list configuration mode and return to global configuration mode.

Example

```
Device(config-ext-nacl)# exit
```

8. Configure a new FlexConnect policy profile.

Example

```
Device(config)# wireless profile flex flex-profile-name
```

Example

```
Device(config)# wireless profile flex test-flex-profile
```

Enters wireless flex profile configuration mode.

9. Configure an ACL policy for the FlexConnect profile.

   Example

   ```
   Device(config-wireless-flex-profile)# acl-policy acl-policy-name
   ```

   Example

   ```
   Device(config-wireless-flex-profile)# acl-policy acl_name
   ```

10. Apply a URL filter list to the FlexConnect profile.

    Example

    ```
    Device(config-wireless-flex-profile)# urlfilter list url-filter-name
    ```

    Example

    ```
    Device(config-wireless-flex-profile)# urlfilter list urllist_flex
    ```

11. Configure a production VLAN name.

    Example

    ```
    Device(config-wireless-flex-profile)# vlan-name prod-vlan-name
    ```

    Example

    ```
    Device(config-wireless-flex-profile)# vlan-name test-vlan
    ```

    Ensure that filter-type post-authentication configuration is in place for the URL filter to work. For information on configuring URL filter list, see the Defining URL Filter List section of the chapter DNS-Based Access Control Lists.

12. Create a production VLAN ID.

    Example

    ```
    Device(config-wireless-flex-profile-vlan)# vlan-id prod-vlan-id
    ```

    Example

    ```
    Device(config-wireless-flex-profile-vlan)# vlan-id 10
    ```

13. Configure an OSU VLAN name.

    Example

    ```
    Device(config-wireless-flex-profile)# vlan-name osu-vlan-name
    ```

Example

```
Device(config-wireless-flex-profile)# vlan-name test-vlan
```

14. Create an OSU VLAN ID.

Example

```
Device(config-wireless-flex-profile-vlan)# vlan-id osu-vlan-id
```

Example

```
Device(config-wireless-flex-profile-vlan)# vlan-id 20
```

The FlexConnect profile is configured with ACL and URL filter policies, enabling local switching and policy enforcement at the access point level for wireless clients.

Configure an OSEN WLAN - single SSID (CLI)

Configure an OSEN WLAN (single SSID).

Configure an OSEN WLAN with single SSID to enable secure wireless access using OSEN encryption.

OSEN (OSU Server-only Authenticated L2 Encryption Network) provides secure connectivity for Hotspot 2.0 deployments. This configuration enables a single SSID WLAN with OSEN security capabilities.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a WLAN and enter WLAN configuration mode.

Example

```
Device(config)# wlan wlan-name wlan-id ssid
```

Example

```
Device(config)# wlan hs2 1 hs2
```

3. Disable fast transition over the data source on the WLAN.

Example

```
Device(config-wlan)# no security ft over-the-ds
```

4. Disable adaptive 11r.

Example

```
Device(config-wlan)# no security ft adaptive
```

5. Enable WPA2 security.

   Example

   ```
   Device(config-wlan)# security wpa wpa2
   ```

6. Enable WPA2 ciphers for AES.

   Example

   ```
   Device(config-wlan)# security wpa wpa2 ciphers aes
   ```

7. Enable WPA OSEN security support.

   Example

   ```
   Device(config-wlan)# security wpa osen
   ```

8. Enable the WLAN.

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

9. Return to global configuration mode.

   Example

   ```
   Device(config-wlan)# exit
   ```

10. Configure a policy profile.

    Example

    ```
    Device(config)# wireless profile policy policy-profile-name
    ```

    Example

    ```
    Device(config)# wireless profile policy policy-hotspot
    ```

11. Attach the Hotspot 2.0 ANQP server to the policy profile.

    Example

    ```
    Device(config-wireless-policy)# hotspot anqp-server server-name
    ```

Example

```
Device(config-wireless-policy)# hotspot anqp-server my-server
```

12. Configure the VLAN ID with OSEN encryption for single SSID.

Example

```
Device(config-wireless-policy)# vlan vlan-id encryption osen
```

Example

```
Device(config-wireless-policy)# vlan 10 encryption osen
```

The OSEN WLAN with single SSID is now configured and enabled, providing secure wireless access using OSEN encryption for Hotspot 2.0 deployments.

Verify Hotspot 2.0 configuration

Lists commands to verify Hotspot 2.0 configuration.

Use these **show** commands to verify the quality of service (QoS) and AP GAS rate limit.

To view whether a QoS map ID is user configured or the default one, use this command:

```
Device# show ap profile <profile name> detailed

QoS Map                         : user-configured
```

To view the QoS map values used and their source, use this command:

```
Device# show ap profile <profile name> qos-map

QoS Map                         : default
DSCP ranges to User Priorities
 User Priority   DSCP low   DSCP high   Upstream UP to DSCP
-------------------------------------------------------------
             0          0           7                     0
             2         16          23                    10
             3         24          31                    18
             4         32          39                    26
             5         40          47                    34
             6         48          55                    46
             7         56          63                    48

DSCP to UP mapping exceptions
 DSCP   User Priority
---------------------
    0               0
    2               1
    4               1
    6               1
   10               2
   12               2
   14               2
   18               3
   20               3
```

```
   22               3
```

To view the AP rate limiter configuration, use this command:

```
Device# show ap name AP0462.73e8.f2c0 config general | i GAS

GAS rate limit Admin status                     : Enabled
Number of GAS request per interval              : 30
GAS rate limit interval (msec)                  : 100
```

Verify client details

Provides command to verify wireless-specific configuration of active clients.

To verify the wireless-specific configuration of active clients based on their MAC address, use this command:

```
Device# show wireless client mac 001e.f64c.1eff detail
.
.
.
Hotspot version : Hotspot 2.0 Release 2
Hotspot PPS MO ID :
Hotspot Terms and Conditions URL :
http://host1.ciscohotspot.com/terms.php?addr=b8:27:eb:5a:dc:39&ap=123
.
.
.
Policy Type : OSEN (within RSN)
Resultant Policies:
    VLAN Name        : VLAN0010
    VLAN             : 10
```

223 Client Roaming Across Policy Profile

Topics:

- Client roaming policy profile

Introduces client roaming across policy profiles, outlining concepts, configuration steps, and verification procedures essential for ensuring seamless client mobility in network environments.

Client roaming policy profile

The client roaming policy profile enables seamless client transition between different profiles.

A client roaming policy profile is a wireless controller configuration element.

- Defines the rules for client transitions between different wireless policy profiles
- determines whether seamless roaming is permitted between WLANs with differing policy profiles, and
- evaluates and implements policy settings during client movement across access points.
- Policy profile: a set of network policies configured by an administrator and associated with a WLAN, governing client access and behavior.
- Policy tag: an identifier that links a WLAN to a specific policy profile in the Cisco Catalyst 9800.

Policy profile roaming behavior and exceptions

In Cisco Catalyst 9800 Series Wireless Controllers, each WLAN associates with a policy profile through a policy tag. By default, the controller does not allow seamless client roaming between the same WLANs when those WLANs are associated with different policy profiles. When this occurs, clients disconnect and must reauthenticate.

When you enable roaming across policy profiles, seamless roaming is allowed only if the profiles differ in a few settings, such as VLAN, ACLs, and session parameters. The controller retains the client's original IP address and applies all new attributes except the VLAN.

Seamless roaming is allowed if the two policy profiles differ only in the following settings:

- Accounting list
- Cisco TrustSec (CTS)
- DHCP TLV caching
- Dot11 5 GHz airtime fairness
- Dot11 2.4 GHz airtime fairness
- ET analytics enable
- HTTP TLV caching
- Idle threshold
- Idle timeout
- mDNS SD service policy
- IPv4 ACL
- IPv6 ACL
- QBSS load
- RADIUS profiling
- Session timeout
- SIP CAC disassociation client
- SIP CAC send-486busy
- VLAN

When a client attempts to roam across policy profiles, the system increments the roam across policy profiles counter for each attempt. If the roam is denied, the system increments the roam across policy profile deny delete reason counter.

> **Note**
>
> This feature is not supported on fabric or on Cisco Catalyst 9800 FlexConnect.

This example shows a client attempting to roam across policy profiles PP1 and PP2 when the roam is denied.

```
wireless profile policy PP1
vlan 42
no shutdown
wireless profile policy PP2
aaa-override
vlan 43
no shutdown
```

Configure client roaming across policy profile

Configure client roaming across policy profiles.

Allow wireless clients to maintain connectivity when moving between different policy profiles by enabling VLAN persistence.

Before you begin

Use these steps to configure client roaming across policy profiles.

Procedure

1. Enable configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable client roaming across different policy profiles.

   Example

   ```
   Device(config) # wireless client vlan-persistent
   ```

3. End the session.

   Example

   ```
   Device(config) # end
   ```

Wireless clients can retain their VLAN assignment as they roam between policy profiles, improving session continuity across the network.

Verify client roaming across policy profiles

Verify client roaming across policy profiles.

These show the client roaming from policy profile PP1 configured with VLAN 42 to policy profile PP2 configured with VLAN 43.

This is the sample output of the **show wireless client mac-address xxxx.xxxx.xxxx detail** command that shows the client is connected to policy profile PP1.

```
Device#show wireless client mac-address xxxx.xxxx.xxxx detail

Client MAC Address : xxxx.xxxx.xxxx
Client MAC Type : Universally Administered Address
Client IPv4 Address : 198.51.100.236
Client Username : cisco
AP MAC Address : xxxx.xxxx.xxxx
AP Name: vinks_ios
AP slot : 1
Client State : Associated
Policy Profile : PP1
Flex Profile : N/A
Wireless LAN Id: 3
WLAN Profile Name: Guest_WLAN_Profile
Wireless LAN Network Name (SSID): Guest_WLAN_Profile
BSSID : 0081.c4f6.6bfb
Connected For : 688 seconds
Protocol : 802.11ac
Channel : 161
Client IIF-ID : 0xa0000001
Association Id : 1
Authentication Algorithm : Open System
Idle state timeout : N/A
Re-Authentication Timeout : 1800 sec (Remaining time: 1112 sec)
Session Warning Time : Timer not running
Input Policy Name  : client-default
Input Policy State : Installed
Input Policy Source : QOS Internal Policy
Output Policy Name  : client-default
Output Policy State : Installed
Output Policy Source : QOS Internal Policy
WMM Support : Enabled
U-APSD Support : Enabled
  U-APSD value : 0
  APSD ACs    : BK, BE, VI, VO
Fastlane Support : Disabled
Client Active State : Active
Power Save : OFF
Current Rate : m8 ss1
Supported Rates : 9.0,18.0,36.0,48.0,54.0
Mobility:
  Move Count                  : 0
  Mobility Role               : Local
  Mobility Roam Type          : None
  Mobility Complete Timestamp : 07/13/2020 02:00:22 UTC
Client Join Time:
  Join Time Of Client : 07/13/2020 02:00:22 UTC
Client State Servers : None
Client ACLs : None
Policy Manager State: Run
Last Policy Manager State : IP Learn Complete
```

```
Client Entry Create Time : 688 seconds
Policy Type : WPA2
Encryption Cipher : CCMP (AES)
Authentication Key Management : 802.1x
User Defined (Private) Network : Disabled
User Defined (Private) Network Drop Unicast : Disabled
Encrypted Traffic Analytics : No
Protected Management Frame - 802.11w : No
EAP Type : EAP-FAST
VLAN Override after Webauth : No
VLAN : 42
Multicast VLAN : 0
WiFi Direct Capabilities:
  WiFi Direct Capable           : No
Central NAT : DISABLED
Session Manager:
  Point of Attachment : capwap_90400006
  IIF ID              : 0x90400006
  Authorized          : TRUE
  Session timeout     : 1800
  Common Session ID: 3C2A090900000000E45E6D59E
  Acct Session ID  : 0x00000000
  Last Tried Aaa Server Details:
   Server IP : 198.51.100.247
  Auth Method Status List
   Method : Dot1x
    SM State          : AUTHENTICATED
    SM Bend State     : IDLE
  Local Policies:
   Service Template : wlan_svc_PP1_local (priority 254)
    VLAN              : 42
    Absolute-Timer    : 1800
  Server Policies:
  Resultant Policies:
    VLAN Name         : VLAN0042
    VLAN              : 42
    Absolute-Timer    : 1800
DNS Snooped IPv4 Addresses : None
DNS Snooped IPv6 Addresses : None
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 0
Fast BSS Transition Details :
  Reassociation Timeout : 0
11v BSS Transition : Not implemented
11v DMS Capable : No
QoS Map Capable : No
FlexConnect Data Switching : N/A
FlexConnect Dhcp Status : N/A
FlexConnect Authentication : N/A
FlexConnect Central Association : N/A
Client Statistics:
  Number of Bytes Received from Client : 19442
  Number of Bytes Sent to Client : 3863
  Number of Packets Received from Client : 197
  Number of Packets Sent to Client : 36
  Number of Policy Errors : 0
  Radio Signal Strength Indicator : -39 dBm
```

```
  Signal to Noise Ratio : 55 dB
Fabric status : Disabled
Radio Measurement Enabled Capabilities
  Capabilities: None
Client Scan Report Time : Timer not running
Client Scan Reports
Assisted Roaming Neighbor List
Nearby AP Statistics:
EoGRE : Pending Classification
Device Type      : Apple-Device
Device Name      : APPLE, INC.
Protocol Map     : 0x000001  (OUI)
Max Client Protocol Capability: 802.11ac Wave 2
Cellular Capability : N/A
Apple Specific Requests(ASR) Capabilities/Statistics Summary
  Regular ASR support: : DISABLED
```

This is the sample output of the **show wireless client mac-address xxxx.xxxx.xxxx detail** command after client has roamed to a policy profile PP2.

```
Client MAC Address : xxxx.xxxx.xxxx
Client MAC Type : Universally Administered Address
Client IPv4 Address : 198.51.100.236
Client Username : cisco
AP MAC Address : xxxx.xxxx.xxxx
AP Name: prateekk_cos_1
AP slot : 1
Client State : Associated
Policy Profile : PP2
Flex Profile : N/A
Wireless LAN Id: 3
WLAN Profile Name: Guest_WLAN_Profile
Wireless LAN Network Name (SSID): Guest_WLAN_Profile
BSSID : a0f8.4985.0029
Connected For : 11 seconds
Protocol : 802.11ac
Channel : 36
Client IIF-ID : 0xa0000001
Association Id : 1
Authentication Algorithm : Open System
Idle state timeout : N/A
Re-Authentication Timeout : 1800 sec (Remaining time: 1789 sec)
Session Warning Time : Timer not running
Input Policy Name  : client-default
Input Policy State : Installed
Input Policy Source : QOS Internal Policy
Output Policy Name  : client-default
Output Policy State : Installed
Output Policy Source : QOS Internal Policy
WMM Support : Enabled
U-APSD Support : Enabled
  U-APSD value : 0
  APSD ACs    : BK, BE, VI, VO
Fastlane Support : Disabled
Client Active State : Active
Power Save : OFF
Current Rate : m9 ss3
Supported Rates : 9.0,18.0,36.0,48.0,54.0
Mobility:
  Move Count                  : 0
  Mobility Role               : Local
  Mobility Roam Type          : L2
```

```
  Mobility Complete Timestamp : 07/13/2020 02:12:19 UTC
Client Join Time:
  Join Time Of Client : 07/13/2020 02:12:19 UTC
Client State Servers : None
Client ACLs : None
Policy Manager State: Run
Last Policy Manager State : IP Learn Complete
Client Entry Create Time : 728 seconds
Policy Type : WPA2
Encryption Cipher : CCMP (AES)
Authentication Key Management : 802.1x
User Defined (Private) Network : Disabled
User Defined (Private) Network Drop Unicast : Disabled
Encrypted Traffic Analytics : No
Protected Management Frame - 802.11w : No
EAP Type : EAP-FAST
VLAN Override after Webauth : No
VLAN : 43
Multicast VLAN : 0
WiFi Direct Capabilities:
  WiFi Direct Capable           : No
Central NAT : DISABLED
Session Manager:
  Point of Attachment : capwap_90000005
  IIF ID              : 0x90000005
  Authorized          : TRUE
  Session timeout     : 1800
  Common Session ID: 3C2A090900000000E45E6D59E
  Acct Session ID  : 0x00000000
  Last Tried Aaa Server Details:
   Server IP : 198.51.100.247
  Auth Method Status List
   Method : Dot1x
    SM State          : AUTHENTICATED
    SM Bend State     : IDLE
  Local Policies:
   Service Template : vlan-42-template (priority 200)
    VLAN              : 42
   Service Template : wlan_svc_PP2_local (priority 254)
    Absolute-Timer    : 1800
  Server Policies:
  Resultant Policies:
    VLAN Name         : VLAN0042
    VLAN              : 42
    Absolute-Timer    : 1800
DNS Snooped IPv4 Addresses : None
DNS Snooped IPv6 Addresses : None
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 0
Fast BSS Transition Details :
  Reassociation Timeout : 0
11v BSS Transition : Not implemented
11v DMS Capable : No
QoS Map Capable : No
FlexConnect Data Switching : N/A
FlexConnect Dhcp Status : N/A
FlexConnect Authentication : N/A
```

```
FlexConnect Central Association : N/A
Client Statistics:
  Number of Bytes Received from Client : 23551
  Number of Bytes Sent to Client : 12588
  Number of Packets Received from Client : 239
  Number of Packets Sent to Client : 71
  Number of Policy Errors : 0
  Radio Signal Strength Indicator : -28 dBm
  Signal to Noise Ratio : 60 dB
Fabric status : Disabled
Radio Measurement Enabled Capabilities
  Capabilities: None
Client Scan Report Time : Timer not running
Client Scan Reports
Assisted Roaming Neighbor List
Nearby AP Statistics:
  prateekk_cos_1 (slot 1)
   antenna 0: 13 s ago ........ -25  dBm
   antenna 1: 13 s ago ........ -25  dBm
EoGRE : No/Simple client
Device Type      : Apple-Device
Device Name      : APPLE, INC.
Protocol Map     : 0x000001  (OUI)
Protocol         : DHCP
Type             : 0    0
Data             : 00

Max Client Protocol Capability: 802.11ac Wave 2
Cellular Capability : N/A
Apple Specific Requests(ASR) Capabilities/Statistics Summary
  Regular ASR support: : DISABLED
```

This is the sample output of the **show wireless stats client detail** command that displays that client roam across policy profile is attempted and roam across policy is not denied.

```
Device #show wireless stats client detail | in Roam
Total Roam Across Policy Profiles : 1
Roam across policy profile deny : 0
```

224 Assisted Roaming

Topics:

- 802.11k neighbor list and assisted roaming

Explains how assisted roaming technology improves wireless client connectivity by enabling access points to guide devices toward optimal connection points within the network.

802.11k neighbor list and assisted roaming

Describes the 802.11k standard that allows an AP to inform 802.11k-capable clients of neighboring BSSIDs and help them optimize scanning and roaming behavior.

The 802.11k neighbor list and assisted roaming is a wireless roaming optimization feature that

- allows an AP to inform 802.11k-capable clients of neighboring BSSIDs (APs in the same SSID)
- helps clients optimize their scanning and roaming behavior, and
- includes Assisted Roaming Prediction Optimization for non-802.11k clients to discourage roaming to suboptimal APs.

Prediction based roaming for non-802.11k clients

You can optimize roaming for non-802.11k clients by generating a prediction neighbor list for each client without sending an 802.11k neighbor list request. When prediction based roaming enables a WLAN, after each successful client association/re-association, the same neighbor list optimization applies on the non-802.11k client to generate and store the neighbor list in the mobile station software data structure. Clients at different locations have different lists because the client probes are seen with different RSSI values by the different neighbors as the clients usually probe before any association or re-association. This list is created with the most updated probe data and predicts the next AP that the client is likely to roam to.

The wireless infrastructure discourages clients from roaming to those less desirable neighbors by denying association if the association request to an AP does not match the entries on the stored prediction neighbor list.

Key configuration parameters for prediction based roaming:

- Denial count: Maximum number of times a client is refused association.
- Prediction threshold: Minimum number of entries required in the prediction list for the assisted roaming feature to activate.

> **Note**
>
> We recommend not configuring two SSIDs with the same name in the controller, which may cause roaming issues.

For more information, see https://www.cisco.com/c/en/us/td/docs/wireless/controller/8-5/Enterprise-Mobility-8-5-Design-Guide/Enterprise_Mobility_8-5_Deployment_Guide/Chapter-11.html#pgfId-1140097.

Restrictions for assisted roaming

Outlines the restrictions and limitations for implementing assisted roaming functionality.

- This feature is supported only on 802.11n capable indoor access points. For a single band configuration, a maximum of 6 neighbors are visible in a neighbor list. For dual band configuration, a maximum of 12 neighbors are visible.
- You can configure assisted roaming only using the CLI.

Configure assisted roaming (GUI)

Configure assisted roaming using GUI to optimize client roaming for both 802.11k and non-802.11k clients.

Configure assisted roaming to enable clients to request neighbor reports containing information about known neighbor access points that are candidates for a service set transition.

Assisted roaming allows clients to request neighbor reports containing information about known neighbor access points that are candidates for a service set transition.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLAN** and click **Add** to add a WLAN or select an existing WLAN.
2. On the **Advanced** tab, go to the **Assisted Roaming (11K)** and select the **Prediction Optimization** checkbox to optimize roaming for non 802.11k clients by generating a prediction neighbor list for each client without sending an 802.11k neighbor list request.
3. Select the **Neighbor List** checkbox to optimize roaming for 802.11K clients by generating a neighbor list for each client without sending an 802.11k neighbor list request. By default, the neighbor list contains only neighbors in the same band with which the client is associated. However, if you select the Dual Band Neighbor List checkbox, it allows 802.11k to return neighbors in both bands.
4. Click **Apply to Device**.

Assisted roaming is configured for the WLAN with the selected optimization options for both 802.11k and non-802.11k clients.

Configure assisted roaming (CLI)

Configure assisted roaming using commands.

Enable assisted roaming to improve client roaming performance by providing neighbor AP information and prediction capabilities.

Assisted roaming helps wireless clients make better roaming decisions by providing neighbor lists, dual-band lists, and prediction features. This configuration improves overall wireless network performance and client experience.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure neighbor floor label bias.

   Example

   ```
   Device(config)# wireless assisted-roaming floor-bias dBm-value
   ```

   Example

   ```
   Device(config)# wireless assisted-roaming floor-bias 20
   ```

   The valid range is from 5 to 25 dBm, and the default value is 15 dBm.

3. Enter WLAN configuration submode.

Example

```
Device(config)# wlan wlan-id
```

Example

```
Device(config)# wlan wlan1
```

The *wlan-id* is the profile name of the configured WLAN.

4. Configure an 802.11k neighbor list for a WLAN.

Example

```
Device(wlan)# assisted-roaming neighbor-list
```

By default, assisted roaming is enabled on the neighbor list when you create a WLAN. The **no** form of the command disables assisted roaming neighbor list.

5. Configure a dual-band 802.11k dual list for a WLAN.

Example

```
Device(wlan)# assisted-roaming dual-list
```

By default, assisted roaming is enabled on the dual list when you create a WLAN. The **no** form of the command disables assisted roaming dual list.

6. Configure assisted roaming prediction list feature for a WLAN.

Example

```
Device(wlan)# assisted-roaming prediction
```

By default, the assisted roaming prediction list is disabled.

> **Note**
>
> A warning message is displayed and load balancing is disabled for the WLAN if load balancing is already enabled for the WLAN.

7. Configure the minimum number of predicted APs required for the prediction list feature to be activated.

Example

```
Device# wireless assisted-roaming prediction-minimum count
```

The default value is 3.

> **Note**
>
> If the number of the AP in the prediction assigned to the client is less than the number that you specify, the assisted roaming feature will not apply on this roam.

8. Configure the maximum number of times a client can be denied association if the association request is sent to an AP that does not match any AP on the prediction.

   Example

   ```
   Device# wireless assisted-roaming denial-maximum count
   ```

   Example

   ```
   Device# wireless assisted-roaming denial-maximum 8
   ```

   The valid range is from 1 to 10, and the default value is 5.

9. Exit configuration mode and return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

Assisted roaming is now configured with the specified parameters, enabling improved client roaming performance through neighbor lists, dual-band support, and prediction capabilities.

Verify assisted roaming

Lists the command to verify assisted roaming configuration on a WLAN.

This command can be used to verify assisted roaming configured on a WLAN:

Command	Description
show WLAN id *WLAN-id*	Displays the WLAN parameters on the WLAN.

Configuration examples for assisted roaming

Provides configuration examples for assisted roaming.

Configuration examples for assisted roaming demonstrate the necessary commands and syntax to configure neighbor floor label bias, disable neighbor list on specific WLANs, configure prediction lists, and set prediction thresholds and maximum denial counts.

This example shows how to configure the neighbor floor label bias:

```
Device# configure terminal
Device(config)# wireless assisted-roaming floor-bias 10
```

```
Device(config)# end
Device# show wlan id 23
```

This example shows how to disable neighbor list on a specific WLAN:

```
Device# configure terminal
Device(config)# wlan test1
Device(config (wlan)# no assisted-roaming neighbor-list
Device(config)(wlan)# end
Device# show wlan id 23
```

This example shows how to configure the prediction list on a specific WLAN:

```
Device# configure terminal
Device(config)# wlan test1
Device(config)(wlan)# assisted-roaming prediction
Device(config)(wlan)# end
Device# show wlan id 23
```

This example shows how to configure the prediction list based on assisted roaming prediction threshold and maximum denial count on a specific WLAN:

```
Device# configure terminal
Device(config)# wireless assisted-roaming prediction-minimum 4
Device(config)# wireless assisted-roaming denial-maximum 4
Device(config)# wlan wlan-name
Device(config)(wlan)# end
Device# show wlan id 23
```

225 802.11r BSS Fast Transition

Topics:

- 802.11r Fast Transition
- 802.11r Fast Transition for SAE authenticated clients

Explains the 802.11r Fast Transition feature for seamless wireless roaming and provides configuration procedures for various security methods including Dot1x, PSK, and SAE authentication.

802.11r Fast Transition

Describes a roaming mechanism that enables initial handshake with a new AP before the client roams to the target access point.

802.11r Fast Transition is a roaming mechanism that

- enables the initial handshake with a new AP even before the corresponding client roams to the target access point
- allows a client and the access points to do the Pairwise Transient Key (PTK) calculation in advance, and
- applies PTK keys to the client and the access points after the client responds to the reassociation request or responds to the exchange with new target AP.

802.11r Fast Transition features

The FT key hierarchy is designed to allow clients to make fast BSS transitions between APs without requiring reauthentication at every AP. WLAN configuration contains a new Authenticated Key Management (AKM) type called Fast Transition (FT).

Client roaming

For a client to move from its current AP to a target AP using the FT protocols, message exchanges are performed using one of these methods:

- Over-the-Air: The client communicates directly with the target AP using IEEE 802.11 authentication with the FT authentication algorithm.
- Over-the-Distribution System (DS): The client communicates with the target AP through the current AP. The communication between the client and the target AP is carried in FT action frames between the client and the current AP and is then sent through the device.

Figure 41: Message exchanges when over–the–air client roaming is configured

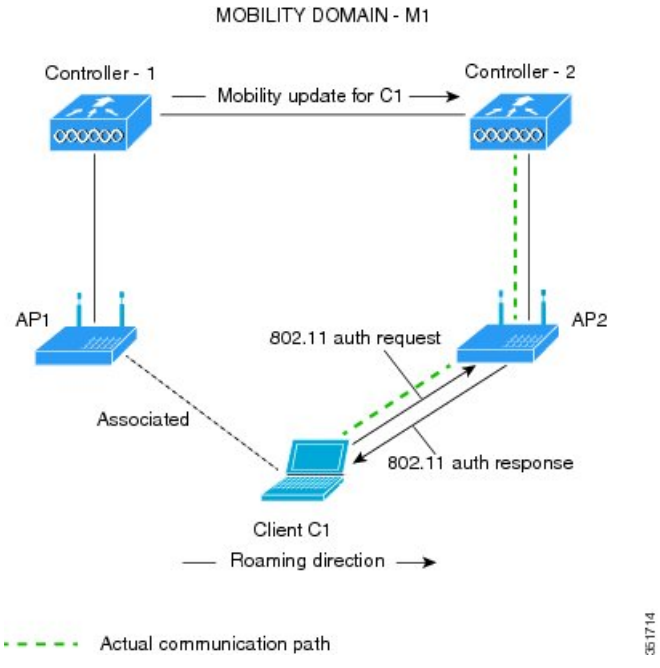
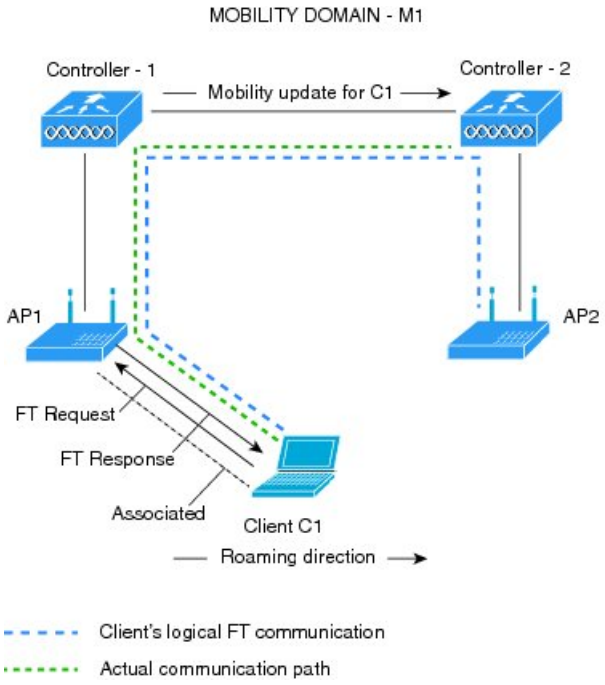


Figure 42: Message exchanges when over–the–DS client roaming is configured



Note

The 802.11r Fast Transition for SAE (FT-SAE) is not restricted to inter controller roaming.

Restrictions for 802.11r Fast Transition

Consider these restrictions when configuring 802.11r Fast Transition.

Consider these restrictions when implementing 802.11r Fast Transition:

- EAP LEAP method is not supported.
- Traffic Specification (TSPEC) is not supported for 802.11r fast roaming. Therefore, RIC IE handling is not supported.
- If WAN link latency exists, fast roaming is also delayed. Voice or data maximum latency should be verified. The Cisco WLC handles 802.11r Fast Transition authentication requests during roaming for both Over-the-Air and Over-the-DS methods.
- Legacy clients cannot associate with a WLAN that has 802.11r enabled if the driver of the supplicant that is responsible for parsing the Robust Security Network Information Exchange (RSN IE) is old and not aware of the additional AKM suites in the IE. Due to this limitation, clients cannot send association requests to WLANs. These clients, however, can still associate with non-802.11r WLANs. Clients that are 802.11r-capable can associate as 802.11i clients on WLANs that have both 802.11i and 802.11r Authentication Key Management Suites enabled.

  The workaround is to enable or upgrade the driver of the legacy clients to work with the new 802.11r AKMs, after which the legacy clients can successfully associate with 802.11r-enabled WLANs.

  Another workaround is to have two SSIDs with the same name, but with different security settings (FT and non-FT).
- Fast Transition resource-request protocol is not supported because clients do not support this protocol. Also, the resource-request protocol is an optional protocol.
- To avoid any Denial of Service (DoS) attack, each Cisco WLC allows a maximum of three Fast Transition handshakes with different APs.

- Non-802.11r-capable devices will not be able to associate with FT-enabled WLAN.
- We recommend 802.11r FT Over-the-Air roaming for FlexConnect deployments.
- FT-SAE Over-the-DS roam is not supported in FlexConnect local authentication mode.
- 802.11r FT-over-DS is enabled by default, when a WLAN is created in the controller. In Cisco Wave 2 APs, local switching local authentication with 802.11r is not supported. To make the local switching local authentication work with Cisco Wave 2 APs, explicitly disable 802.11r in WLAN. A sample configuration is given below:

```
wlan local-dot1x 24 local-dot1x
no security ft over-the-ds
no security ft adaptive
security dot1x authentication-list spwifi_dot1x
no shutdown
```

Monitor 802.11r fast transition

Lists the CLI commands to monitor 802.11r fast transition.

These commands can be used to monitor 802.11r Fast Transition:

Command	Description
show WLAN name *WLAN-name*	Displays a summary of the configured parameters on the WLAN.

Command	Description
show wireless client mac-address *mac-address*	Displays the summary of the 802.11r authentication key management configuration on a client.

```
. . .
. . .
Client Capabilities
  CF Pollable : Not implemented
  CF Poll Request : Not implemented
  Short Preamble : Not implemented
  PBCC : Not implemented
  Channel Agility : Not implemented
  Listen Interval : 15
  Fast BSS Transition : Implemented
Fast BSS Transition Details :
Client Statistics:
  Number of Bytes Received : 9019
  Number of Bytes Sent : 3765
  Number of Packets Received : 130
  Number of Packets Sent : 36
  Number of EAP Id Request Msg
Timeouts : 0
  Number of EAP Request Msg Timeouts
 : 0
  Number of EAP Key Msg Timeouts : 0
  Number of Data Retries : 1
  Number of RTS Retries : 0
  Number of Duplicate Received Packets
 : 1
  Number of Decrypt Failed Packets :
 0
  Number of Mic Failured Packets : 0
  Number of Mic Missing Packets : 0
  Number of Policy Errors : 0
  Radio Signal Strength Indicator :
-48 dBm
  Signal to Noise Ratio : 40 dB
. . .
. . .
```

Configure 802.11r Fast Transition in an open WLAN (CLI)

Configure 802.11r Fast Transition on an open WLAN using commands.

Enable 802.11r Fast Transition capabilities on an open WLAN to improve roaming performance for wireless clients.

802.11r Fast Transition allows wireless clients to roam between access points more quickly by pre-authenticating with target access points. This configuration is performed on an open WLAN without WPA security.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enter WLAN configuration submode.

Example

```
Device# wlan profile-name
```

Example

```
Device# wlan test4
```

The *profile-name* is the profile name of the configured WLAN.

3. Associate the client VLAN to the WLAN.

Example

```
Device(config-wlan)# client vlan vlan-id
```

Example

```
Device(config-wlan)# client vlan 0120
```

4. Disable WPA security.

Example

```
Device(config-wlan)# no security wpa
```

5. Disable security AKM for dot1x.

Example

```
Device(config-wlan)# no security wpa akm dot1x
```

6. Disable WPA2 security.

Example

```
Device(config-wlan)# no security wpa wpa2
```

7. Disable WPA2 ciphers for AES.

Example

```
Device(config-wlan)# no security wpa wpa2 ciphers aes
```

8. Enable 802.11r Fast Transition parameters.

Example

```
Device(config-wlan)# security ft
```

9. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

10. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

Alternatively, you can also press Ctrl-z to exit global configuration mode

---

The WLAN is now configured with 802.11r Fast Transition enabled, allowing wireless clients to perform faster roaming between access points.

Configure 802.11r BSS Fast Transition on a Dot1x security enabled WLAN (CLI)

Configure 802.11r Fast Transition in an open WLAN using commands.

Enable 802.11r BSS Fast Transition on a WLAN with 802.1x authentication to provide seamless roaming for wireless clients.

802.11r Fast Transition reduces the time required for wireless clients to roam between access points by enabling pre-authentication and key caching. This configuration is particularly useful in environments where clients need to maintain connectivity while moving between coverage areas.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enter WLAN configuration submode.

Example

```
Device# wlan profile-name
```

```
Device# wlan wlan-bss-ft
```

The *profile-name* is the profile name of the configured WLAN.

3. Associate the client VLAN to this WLAN.

Example

```
Device(config-wlan)# client vlan vlan-name
```

Example

```
Device(config-wlan)# client vlan 0120
```

4. Enable the local auth EAP profile.

   Example

   ```
   Device(config-wlan)# local-auth local-auth-profile-eap
   ```

5. Enable security authentication list for dot1x security.

   Example

   ```
   Device(config-wlan)# security dot1x authentication-list default
   ```

   The configuration is similar for all dot1x security WLANs.

6. Enable 802.11r Fast Transition on the WLAN.

   Example

   ```
   Device(config-wlan)# security ft
   ```

7. Enable 802.1x security on the WLAN.

   Example

   ```
   Device(config-wlan)# security wpa akm ft dot1x
   ```

8. Enable the WLAN.

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

9. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wlan)# end
   ```

   Alternatively, you can also press Ctrl-z to exit global configuration mode

---

Configure 802.11r Fast Transition on a PSK security–enabled WLAN (CLI)

Configure 802.11r Fast Transition on a PSK security-enabled WLAN using commands.

Enable 802.11r Fast Transition capability on a WLAN that uses PSK security to improve roaming performance for wireless clients.

802.11r Fast Transition reduces the time required for client devices to roam between access points by pre-authenticating with neighboring APs. This configuration is performed on WLANs that use pre-shared key (PSK) authentication.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter WLAN configuration mode.

   Example

   ```
   Device(config)# wlan wlan-profile-name
   ```

3. Associate the client VLAN to this WLAN.

   Example

   ```
   Device(config-wlan)# client vlan vlan-name
   ```

   Example

   ```
   Device(config-wlan)# client vlan 0120
   ```

4. Disable security AKM for dot1x.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

5. Configure Fast Transition PSK support.

   Example

   ```
   Device(config-wlan)# security wpa akm ft psk
   ```

6. Configure PSK AKM shared key.

   Example

   ```
   Device(config-wlan)# security wpa akm psk set-key {ascii {0 | 8} | hex {0 |
   8}} key-value
   ```

   Example

   ```
   Device(config-wlan)# security wpa akm psk set-key ascii 0 test
   ```

7. Configure 802.11r Fast Transition.

   Example

   ```
   Device(config-wlan)# security ft
   ```

8. Enable the WLAN.

   Example

   ```
   Device(config-wlan)# no shutdown
   ```

9. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wlan)# end
   ```

   Alternatively, you can also press Ctrl-Z to exit global configuration mode.

The WLAN is now configured with 802.11r Fast Transition support using PSK authentication, enabling faster client roaming between access points.

Disable 802.11r fast transition (GUI)

Configure the system to disable 802.11r Fast Transition using the GUI interface.

Disable 802.11r Fast Transition functionality on a WLAN configuration to prevent automatic roaming assistance between access points.

Use this procedure when you need to turn off Fast Transition capabilities for a specific WLAN. Note that fast transition cannot be enabled or disabled if you have configured an SSID with Open Authentication.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. On the **WLANs** page, click the WLAN name.
3. In the **Edit WLAN** window, click the **Security** > **Layer2** tab.
4. From the **Fast Transition** drop-down list, choose **Disabled**.

   > **Note**
   >
   > You cannot enable or disable Fast Transition, if you have configured an SSID with Open Authentication.

5. Click **Update & Apply to Device**.

The 802.11r Fast Transition feature is disabled for the selected WLAN, and the configuration changes are applied to the device.

Disable 802.11r Fast Transition (CLI)

Configure 802.11r Fast Transition disablement using commands.

Disable 802.11r Fast Transition on WLANs to prevent Fast Transitions in 802.11r-enabled networks.

Use this procedure when you need to prevent Fast Transitions in 802.11r-enabled WLANs using the command line interface.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the WLAN configuration submode.

   Example

   ```
   Device(config)# wlan profile-name
   ```

   ```
   Device(config)# wlan wlan-test4
   ```

3. Disable the FT AKMs that are enabled before disabling FT.

   Example

   ```
   Device(config-wlan)# no security wpa akm ft dot1x
   ```

   The available FT AKMs are: SAE-FT, SAE-EXT-FT and 802.1x-FT.

4. Disable 802.11r Fast Transition on the WLAN.

   Example

   ```
   Device(config-wlan)# no security ft {adaptive | over-the-ds |
   reassociation-timeout timeout-in-seconds}
   ```

   ```
   Device(config-wlan)# no security ft over-the-ds
   ```

5. Return to privileged EXEC mode.

   Example

   ```
   Device(config)# end
   ```

802.11r Fast Transition is disabled on the WLAN and Fast Transitions will no longer occur for clients connecting to this network.

802.11r Fast Transition for SAE authenticated clients

Explains fast transition feature support for Simultaneous Authentication of Equals-based (SAE-based) fast roaming along with Pairwise Master Key (PMK) caching.

802.11r Fast Transition for SAE authenticated clients is a wireless security feature that

- supports Simultaneous Authentication of Equals-based (SAE-based) fast roaming support along with Pairwise Master Key (PMK) caching
- is an addition to the existing PMK caching-based fast roaming support, and
- is available from Cisco-IOS XE 17.9.1 release onwards.

Fast Transition protocol

During a Base Station Subsystem (BSS) transition, the Fast BSS transition feature reduces the connectivity time loss between an Station (STA) and Direct Switching. The Fast Transition protocols are part of the reassociation service, and apply to the STA transitions between the APs in the same mobility domain and Exteneded Service Set (ESS). The Fast Transition protocols need information to be exchanged during the initial association (or a later reassociation) between an STA and an AP. The initial exchange is referred to as the *FT initial mobility domain association*. Similarly, subsequent reassociations to the APs in the same mobility domain use the Fast Transition protocols.

Note

STA is known as Fast Transition Originator.

These are the FT protocols:

- **Fast Transition Protocol**: This protocol is executed when a Fast Transition Originator makes a transition to a target AP and does not require a resource request before its transition.
- **Fast Transition Resource Request Protocol**: This protocol is executed when a Fast Transition Originator requires a resource request prior to its transition.
- **Over-the-Air**: The Fast Transition Originator communicates with the target AP using IEEE 802.11 authentication with Fast Transition authentication algorithm.
- **Over-the-DS**: The Fast Transition Originator communicates with the target AP using the current AP. The communication between the Fast Transition Originator and target AP is carried in Fast Transition action frames between the Fast Transition Originator and the current AP.

The Fast Transition feature supports a new AKM for FT-SAE, specifically the **00-0F-AC:9**.

Fast Transition initial mobility domain association

An STA includes Mobility Domain Element (MDE) and Robust Security Network Element (RSNE) in the (re)association request frame. The AP responds by including FTE, MDE, and RSNE in the (re)association response frame.

That is, an STA initiates the Fast Transition initial mobility domain association procedures by performing an IEEE 802.11 authentication using the SAE algorithm.

After successful SAE authentication, the STA and AP perform a Fast Transition four-way handshake.

> **Note**
>
> - If the MDE that is received by an AP or a controller does not match the contents advertised in the beacon and probe response frames, the AP or controller rejects the (re)association request frame with the STATUS_INVALID_MDE code.
> - If an MDE is available in the (re)association request frame and the contents of RSNE do not indicate a negotiated SAE AKM of Fast BSS Transition (00-0F-AC:9 suite type), the AP rejects with STATUS_INVALID_AKMP code.
>
> After an SAE authentication, the controller receives the PMK, resulting in the successful completion of SAE.

Feature history for 802.11r Fast Transition

Lists the feature history for 802.11r fast transition.

This table provides release and related information about the feature explained in this section.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 195: Feature History for 802.11r Fast Transition

Feature	Release	Feature Information
802.11r Fast Transition for SAE (FT-SAE) Authenticated Clients	Cisco IOS XE Cupertino 17.9.1	From Cisco-IOS XE 17.9.1 release onwards, the Fast Transition supports SAE-based Fast Roaming support along with PMK caching. This feature is an addition to the existing PMK caching-based fast roam support.

Configure 802.11r Fast Transition on a SAE security-enabled WLAN (GUI)

Configure 802.11r fast transition on a SAE security-enabled WLAN using GUI.

This task configures 802.11r Fast Transition on a SAE (Simultaneous Authentication of Equals) security-enabled WLAN to enable seamless roaming for wireless clients.

802.11r Fast Transition reduces the time required for client authentication when roaming between access points in the same mobility domain. When combined with SAE security (WPA3), it provides both security and seamless connectivity for wireless clients.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add**.
3. In the **General** tab, enter the **Profile Name**, the **SSID**, and the **WLAN ID**.
4. Choose **Security** > **Layer2** tab.
5. Click the **WPA3** radio button as security mode.
6. Check the required **WPA Parameters** check boxes and the **AES(CCMP128)** check box.

7. From the **Status** drop-down list, choose **Enabled**.
8. Check the **FT+SAE** check box.
9. Enter the **Pre-Shared Key**.
10. From the **PSK Format** drop-down list, choose PSK Format and from the **PSK Type** drop-down list, choose PSK Type.
11. Click **Apply to Device**.

The WLAN is configured with 802.11r fast transition and SAE security enabled. Wireless clients can now roam seamlessly between access points in the mobility domain while maintaining WPA3 security.

Configure 802.11r Fast Transition on an SAE security-enabled WLAN (CLI)

Configure 802.11r Fast Transition on an SAE security-enabled WLAN using commands.

Enable 802.11r Fast Transition capabilities on a WLAN that uses SAE (WPA3) security to provide seamless roaming for wireless clients.

802.11r Fast Transition reduces the time required for clients to roam between access points by pre-authenticating with neighboring access points. This configuration is particularly useful in environments where clients need to maintain connectivity while moving between coverage areas.

Procedure

---

1. Enable privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

   Enter your password, if prompted.

2. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Configure the WLAN and SSID.

   Example

   ```
   Device(config)# wlan wlan-name wlan-id ssid
   ```

   Example

   ```
   Device(config)# wlan wlan-ft-sae 10 wlan-ft-sae
   ```

4. Enable 802.11r Fast Transition on the WLAN.

   Example

   ```
   Device(config-wlan)# security ft
   ```

5. Disable WPA2 security.

   Example

   ```
   Device(config-wlan)# no security wpa wpa2
   ```

6. Configure the preshared key on a WLAN.

   Example

   ```
   Device(config-wlan)# security wpa psk set-key ascii encryption-type key-string
   ```

   Example

   ```
   Device(config-wlan)# security wpa psk set-key ascii 0 123456789
   ```

   Note

   WPA preshared keys must contain 8 to 63 ASCII text characters or 64 hexadecimal characters.

7. Disable security AKM for dot1x.

   Example

   ```
   Device(config-wlan)# no security wpa akm dot1x
   ```

8. Configure 802.11r Fast Transition on an SAE security-enabled WLAN.

   Example

   ```
   Device(config-wlan)# security wpa akm ft sae
   ```

9. Enable WPA3 support.

   Example

   ```
   Device(config-wlan)# security wpa wpa3
   ```

10. Require clients to negotiate 802.11w PMF protection on a WLAN.

    Example

    ```
    Device(config-wlan)# security pmf mandatory
    ```

11. Enable the WLAN.

Example

```
Device(config-wlan)# no shutdown
```

The WLAN is now configured with 802.11r Fast Transition and SAE security, enabling secure and seamless client roaming between APs.

Verify 802.11r Fast Transition SAE

Lists commands to verify 802.11r fast transition SAE configuration and status.

Use these verification commands to confirm 802.11r Fast Transition SAE is working correctly and to view related client details, authentication information, and statistics.

Client summary verification

To view the Fast Transition SAE details, use this command:

```
Device# show wireless client summary
Number of Clients: 1

MAC Address AP Name Type ID State Protocol Method Role
---------------------------------------------------------------------------------
2c33.7a5b.8fc5 APF4BD.9EBD.A66C WLAN 10 Run 11n(2.4) FT-SAE Local

Number of Excluded Clients: 0
```

To view the client summary details from an AP, use this command:

```
AP# show client summary

Radio Driver client Summary:
============================
apr0v1
-------
apr0v4
-------
ADDR AID CHAN TXRATE RXRATE RSSI MINRSSI MAXRSSI IDLE TXSEQ RXSEQ CAPS XCAPS
ACAPS ERP STATE MAXRATE(DOT11) HTCAPS VHTCAPS ASSOCTIME IEs MODE RXNSS TXNSS
PSMODE
a0:fb:c5:ab:c3:41 1 11 114M 97M -47 -60 -40 0 0 65535 EPSs BORI NULL 0 f 286800
 AP 1g 00:19:53 RSN WME IEEE80211_MODE_11AXG_HE20 2 2 1
LM BRP BRA
RSSI is combined over chains in dBm
Minimum Tx Power : 0
Maximum Tx Power : 0
HT Capability : Yes
VHT Capability : No
MU capable : No
SNR : 48
Operating band : 2.4GHz
Current Operating class : 0
Supported Rates : 2 4 11 22 12 18 24 36 48 72 96 108
Channels supported : 2412 2417 2422 2427 2432 2437 2442 2447 2452 2457 2462
2467 2472
Max STA phymode : IEEE80211_MODE_11AXG_HE20
apr1v1
-------
apr1v4
```

```
-------

WCP client Summary:
====================
mac radio vap aid state encr Maxrate Assoc Cap is_wgb_wired wgb_mac_addr
A0:FB:C5:AB:C3:41 0 4 1 FWD AES_CCM128 MCS92SS HE HE false 00:00:00:00:00:00



Assoc time:
============
mac assoc_time
A0:FB:C5:AB:C3:41 00d:00h:19m:55s


Datapath IPv4 client Summary:
==============================
id vap port node tunnel mac seen_ip hashed_ip sniff_ago confirm_ago
A0:FB:C5:AB:C3:41 4 apr0v4 6.4.26.28 - A0:FB:C5:AB:C3:41 192.100.2.153
10.0.21.68 0.110000 0.100000

Datapath IPv6 client Summary:
==============================
client mac seen_ip6 age scope port
1 A0:FB:C5:AB:C3:41 fe80::c2f:f0c4:9fa5:2608 1 link-local apr0v4
```

FlexConnect verification

To view FlexConnect-related details from an AP, use this command:

```
AP# show flexconnect dot11R

Total number of DOT11R cache entries: 1

HW Address Life Time(s) BSSID R0KhId R1KhId vlanOverride aclOverride
ipv6AclOverride qosOverride iPSK
A0:FB:C5:AB:C3:41 558 2C:57:41:59:F5:C4 239.13.224.36 45:49:7B:38:11:6A N/A 0
 <>
```

Authentication verification

To view the authentication key management details, use this command:

```
Device# show wireless client mac-address 28c2.1f54.e6d6 detail
Authentication Algorithm : Open System
Authentication Key Management : FT-SAE
FlexConnect Authentication : Central
```

To verify whether AKM Fast Transition-SAE is enabled or not, use this command:

```
Device# show wlan name [wlan-profile-name]

Auth Key Management
FT SAE : [Enabled | Disabled]
```

PMK cache and statistics verification

To verify the PMK cache details, use this command:

```
Device# show wireless pmk-cache
…...
Type Dot11R
…..
```

To view the WPA3 SAE details, use this command:

```
Device# show wireless stats client detail

Total FT/LocalAuth requests                        : 20
Total 11r ft authentication requests received    : 9
Total 11r ft authentication response success     : 9
Total 11r ft authentication response failure     : 0
Total 11r ft action requests received            : 17
Total 11r ft action response success             : 8
Total 11r ft action response failure             : 9
Total 11r PMKR0-Name mismatch                    : 0
Total 11r PMKR1-Name mismatch                    : 5
Total 11r MDID mismatch                          : 9
Total roam attempts                              : 15
  Total 11r roam attempts                        : 15
……
……
Total WPA3 SAE attempts                          : 0
Total WPA3 SAE successful authentications        : 0
Total WPA3 SAE authentication failures           : 0
  Total incomplete protocol failures             : 0
Total WPA3 SAE commit messages received          : 0
Total WPA3 SAE commit messages rejected                                  : 0
  Total unsupported group rejections                                     : 0
  Total PWE method mismatch for SAE Hash to Element commit received      : 0
Total PWE method mismatch for SAE Hunting And Pecking commit received : 0
Total WPA3 SAE commit messages sent              : 0
Total WPA3 SAE confirm messages received         : 0
Total WPA3 SAE confirm messages rejected         : 0
  Total WPA3 SAE message confirm field mismatch  : 0
  Total WPA3 SAE confirm message invalid length  : 0
Total WPA3 SAE confirm messages sent             : 0
Total WPA3 SAE Open Sessions                     : 0
Total SAE Message drops due to throttling        : 0
Total WPA3 SAE Hash to Element commit received   : 0
Total WPA3 SAE Hunting and Pecking commit received : 0
……
……
Total Flexconnect local-auth roam attempts       : 8
  Total 11r flex roam attempts                   : 0…..
….
Total client delete reasons
  SAE authentication failure                                        : 0
  DOT11 SAE invalid message                                         : 0
```

226 802.11v

Topics:

- 802.11v amendment

Outlines 802.11v capabilities, focusing on prerequisites, restrictions, BSS Transition Management features, and step-by-step procedures for configuring and enabling 802.11v BSS Transition Management on wireless networks using both GUI and CLI methods.

802.11v amendment

Introduces the 802.11v amendment, highlighting its enhancements for wireless network management, including network assisted power savings, network assisted roaming, Directed Multicast Service, and BSS max idle period to optimize client battery life and improve connectivity.

The 802.11v amendment is a wireless network standard that

- describes numerous enhancements to wireless network management
- enables Network assisted Power Savings which helps clients to improve the battery life by enabling them to sleep longer, and
- enables Network assisted Roaming which enables the WLAN to send requests to associated clients, advising the clients as to better APs to associate to, for both load balancing and in directing poorly connected clients.

802.11v network assisted power savings

Wireless devices consume battery to maintain their connection to the clients, in several ways:

- By waking up at regular intervals to listen to the access point beacons containing a DTIM, which indicates buffered broadcast or multicast traffic that the access point delivers to the clients.
- By sending null frames to the access points, in the form of keepalive messages– to maintain connection with access points.
- Devices also periodically listen to beacons (even in the absence of DTIM fields) to synchronize their clock to that of the corresponding access point.

All these processes consume battery and this consumption particularly impacts devices (such as Apple), because these devices use a conservative session timeout estimation, and therefore, wake up often to send keepalive messages. The 802.11 standard, without 802.11v, does not include any mechanism for the controller or the access points to communicate to wireless clients about the session timeout for the local client.

To save the power of clients due to the mentioned tasks in wireless network, these features in the 802.11v standard are used:

- Directed multicast service
- Base station subsystem (BSS) max idle period

Directed multicast service:

Using Directed Multicast Service (DMS), the client requests the access point to transmit the required multicast packet as unicast frames. This allows the client to receive the multicast packets it has ignored while in sleep mode and also ensures Layer 2 reliability. Furthermore, the unicast frame is transmitted to the client at a potentially higher wireless link rate which enables the client to receive the packet quickly by enabling the radio for a shorter duration, thus also saving battery power. Since the wireless client also does not have to wake up at each DTIM interval in order to receive multicast traffic, longer sleeping intervals are allowed.

BSS max idle period:

The BSS max idle period is the timeframe during which an access point (AP) does not disassociate a client due to nonreceipt of frames from the connected client. This helps ensure that the client device does not send keepalive messages frequently. The idle period timer value is transmitted using the association and reassociation response frame from the access point to the client. The idle time value indicates the maximum time that a client can remain idle without transmitting any frame to an access point. As a result, the clients remain in sleep mode for a longer duration without transmitting the keepalive messages often. This in turn contributes to saving battery power.

Prerequisites for configuring 802.11v

Outlines prerequisites for configuring 802.11v.

Requirements for configuring 802.11v support in your wireless network environment.

- Applies for Apple clients like Apple iPad, iPhone, and so on, that run on Apple iOS version 7 or later.
- Supports local mode; also supports FlexConnect access points in central authentication modes only.

Restrictions for 802.11v

Describes the limitations and requirements for implementing 802.11v, specifically that client devices must support BSS Transition capabilities for proper functionality.

Client needs to support 802.11v BSS Transition.

802.11v BSS transition management

Describes 802.11v BSS transition management scenarios and functionality.

802.11v BSS transition management

- enables clients to request better AP options before roaming
- allows APs to suggest client transitions for load balancing, and
- supports optimized roaming based on RSSI and rate requirements.

802.11v BSS transition scenarios

802.11v BSS Transition is applied in these three scenarios:

- Solicited request: Client can send an 802.11v Basic Service Set (BSS) Transition Management Query before roaming for a better option of AP to reassociate with.
- Unsolicited Load Balancing request: If an AP is heavily loaded, it sends out an 802.11v BSS Transition Management Request to an associated client.
- Unsolicited Optimized Roaming request: If a client's RSSI and rate do not meet the requirements, the corresponding AP sends out an 802.11v BSS Transition Management Request to this client.

> **Note**
>
> 802.11v BSS Transition Management Request is a suggestion (or advice) given to a client, which the client can choose to follow or ignore. To force the task of disassociating a client, turn on the disassociation-imminent function. This disassociates the client after a period if the client is not reassociated to another AP.

Configure 802.11v BSS transition management (GUI)

Configure 802.11v BSS transition management using GUI.

Enable and customize 802.11v BSS transition management features for a particular WLAN to improve client roaming, battery life, and network efficiency.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click **Add** to create WLANs.

   The **Add WLAN** page is displayed.
3. In the **Advanced** tab and **11v BSS Transition Support** section, select the **BSS Transition** check box to enable BSS transition per WLAN.
4. Enable the **Dual Neighbor List** check box to include the neighbors of other radio slots of the same AP in the BSS transition response.

   Note

   This is applicable only for 2.4 GHz and 5 GHz radio slots.
5. Enable the **BSS Max Idle Service** check box to help clients and APs efficiently decide how long to remain associated when no traffic is being transmitted. The device uses this information to preserve device battery life.
6. Enable the **BSS Max Idle Protected** check box to enable the AP to accept only authenticated frames (encrypted with Robust Security Network (RSN) information) from the client to reset the BSS Max Idle period counter. Without protected mode, any data or management frame (encrypted or unencrypted) sent by the client will reset the idle timer for the client.
7. Enable the **Directed Multicast Service** check box to request the AP to send a multicast stream as unicast, to any DMS capable client on this WLAN.
8. Click **Save & Apply to Device**.

The selected WLAN is now configured with 802.11v BSS transition management features. Clients benefit from improved roaming and battery conservation, and multicast streams can be delivered more efficiently.

Configure 802.11v BSS transition management (CLI)

Configure 802.11v BSS transition management using commands.

Enable 802.11v BSS transition management on a WLAN to improve client roaming and load balancing.

802.11v BSS transition management allows access points to provide roaming assistance to clients by suggesting better access points and triggering client roaming decisions. This feature is applied in scenarios where network optimization and seamless client mobility are required.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Configure WLAN profile and enter the WLAN profile configuration mode.

Example

```
Device(config)# wlan profile-name
```

Example

```
Device(config)# wlan test-wlan
```

3. Shutdown the WLAN profile.

Example

```
Device(config-wlan)# shutdown
```

4. Configure BSS transition per WLAN.

Example

```
Device(config-wlan)# bss-transition
```

5. Configure BSS transition disassociation imminent per WLAN.

Example

```
Device(config-wlan)# bss-transition disassociation-imminent
```

6. Enable the WLAN profile.

Example

```
Device(config-wlan)# no shutdown
```

7. Return to privileged EXEC mode.

Example

```
Device(config-wlan)# end
```

Alternatively, you can press CTRL + Z to exit global configuration mode.

802.11v BSS transition management is now configured on the WLAN, enabling improved client roaming assistance and network optimization.

227 Virtual Routing and Forwarding

Topics:

- VRF support

Outlines Virtual Routing and Forwarding implementation, including creation and configuration of VRF definitions, mapping to switched virtual interfaces, and DHCP relay integration.

VRF support

Explains Virtual Routing and Forwarding (VRF) support functionality for wireless controllers.

Virtual Routing and Forwarding (VRF) support is a logical representation feature that

- groups Layer 3 entities such as IP addresses and routes
- provides the controller with the capability to split the control plane and data plane into multiple segregated logical instances within the same controller platform
- makes these planes VRF aware.

VRF use cases

VRF plays a crucial role in these use cases:

- Enabling flexible routing in infrastructure services such as AAA, DHCP, DNS, and more.
- Facilitating support for overlapping IP addresses.
- Permitting traffic from VRF-A to VRF-B using GRT. Route leak between two VRFs (VRF-A and VRF-B) is possible using a Global Routing Table (GRT). The direct route leak between VRFs are not supported.

> **Note**
>
> Direct route leakage between VRFs is not permitted. It should proceed from VRF A to GRT, then to the intended destination, VRF B.

For a multitenant network such as an airport, this allows you to provide wireless services to different tenants (including airlines and shops) at the airport by supporting two clients with different MAC addresses using the same IP address. With VRF support, AP in local mode or AP in FlexConnect mode with central switching policy can have two clients with the same IP even if they belong to different VRFs.

> **Note**
>
> - From Cisco IOS XE Dublin 17.12.1, overlapping IP address can be supported without disabling device tracking, by using VRF.
> - The configuration of VRF is not exclusive to this release, but its effectiveness begins from this release.

VRFs supported per platform:

- Cisco Catalyst 9800-80 Wireless Controller: 8181
- Cisco Catalyst 9800-40 Wireless Controller: 8181
- Cisco Catalyst 9800-L Wireless Controller: 8181

Guidelines and restrictions for VRF support

Outlines guidelines and restrictions for VRF support.

Follow these guidelines and restrictions when implementing VRF support:

- Supports only Local mode and FlexConnect mode (central DHCP and central switching).
- Supports only one VRF per WLAN.

> **Note**
>
> The maximum number of VRFs supported on a platform depends on the number of WLANs supported on the hardware platform.

- Supports static VRF ID allocation. All the configured VRFs should be associated with an SVI.
- Supports switch virtual interfaces (SVI) other than Wireless Management Interface (WMI).
- Supports only external DHCP servers.
- mDNS gateway is not supported.
- We recommend using commands to configure the feature because all VRF configurations are currently not supported through GUI.

Create a VRF instance

Configure a VRF instance to create isolated routing tables for network segmentation.

Create a VRF instance to enable network virtualization and traffic isolation within a single physical device.

VRF (Virtual Routing and Forwarding) instances allow you to create multiple virtual routers within a single physical router, providing network segmentation and isolation. This is useful for service providers or enterprises that need to maintain separate routing domains.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a VRF instance and enter VRF configuration mode.

   Example

   ```
   Device(config)# vrf definition vrf-name
   ```

   Example

   ```
   Device(config)# vrf definition red-vrf
   ```

3. Set an IPv4 address family.

   Example

   ```
   Device(config-vrf)# address-family ipv4
   ```

4. Exit from VRF address-family configuration submode.

   Example

   ```
   Device((config-vrf-af)# exit-address-family
   ```

5. Set an IPv6 address family.

   Example

   ```
   Device(config-vrf)# address-family ipv6
   ```

6. Exit from VRF address-family configuration submode.

   Example

   ```
   Device((config-vrf-af)# exit-address-family
   ```

7. Return to privileged EXEC mode.

   Example

   ```
   Device(config-vrf)# end
   ```

The VRF instance is created with both IPv4 and IPv6 address families configured, providing isolated routing tables for network segmentation.

Map VRF to SVI

Configure VRF mapping to an SVI interface.

Associate a VRF with a VLAN interface to enable Layer 3 separation and routing isolation.

Mapping a VRF to an SVI allows you to create isolated routing domains where traffic from different VRFs remains separated. This configuration is used when implementing network segmentation or multi-tenancy scenarios.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the VLAN interface to be associated with the VRF. Enter the interface configuration mode for the specified VLAN interface.

Example

```
Device(config)# interface interface-type-number
```

Example

```
Device(config)# interface vlan181
```

3. Associate the VRF with the Layer 3 interface.

Example

```
Device(config-if)# vrf forwarding vrf-name
```

Example

```
Device(config-if)# vrf forwarding red-vrf
```

This command activates multiprotocol VRF on the interface.

4. Disable proxy ARP.

Example

```
Device(config-if)# no ip proxy-arp
```

5. Enable the interface.

Example

```
Device(config-if)# no shutdown
```

6. Return to privileged EXEC mode.

Example

```
Device(config-if)# end
```

The VRF is now mapped to the SVI, and the interface is configured for VRF-aware routing with proxy ARP disabled.

Add VRF name through Option 82 for DHCP relay

Configure VRF name transmission through Option 82 for DHCP relay.

Enable the transmission of VRF name through Option 82 during DHCP relay operations.

VRF-based Sub Option 151 allows DHCP relay agents to include VRF information in Option 82, helping DHCP servers identify the originating VRF for client requests. This configuration is performed on wireless profile policies.

Procedure

---

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Enable configuration for the specified profile policy.

   Example

   ```
   Device(config)# wireless profile policy policy-name
   ```

   Example

   ```
   Device(config)# wireless profile policy red-vrf
   ```

3. Shut down the wireless profile policy.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

4. Enable VRF-based Sub Option 151.

   Example

   ```
   Device(config-wireless-policy)# ipv4 dhcp opt82 VRF
   ```

5. Enable the wireless profile policy.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

6. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

The wireless profile policy is now configured to transmit VRF name through Option 82 during DHCP relay operations.

Add VRF name to DHCP server for DHCP relay

Configure VRF name to DHCP server for DHCP relay to enable separate VRF configuration between the DHCP server and client.

Enable separate VRF configuration between the DHCP server and client in DHCP relay scenarios.

When implementing DHCP relay, this procedure allows you to configure the DHCP server's VRF separately from the VRF of the client.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enable configuration for the specified profile policy.

   Example

   ```
   Device(config)# wireless profile policy policy-name
   ```

   Example

   ```
   Device(config)# wireless profile policy red-vrf
   ```

3. Shut down the wireless profile policy.

   Example

   ```
   Device(config-wireless-policy)# shutdown
   ```

4. Configure the WLAN's IPv4 DHCP server IP address and VRF name.

   Example

   ```
   Device(config-wireless-policy)# ipv4 dhcp server ip-address vrf vrf-name
   ```

   Example

   ```
   Device(config-wireless-policy)# ipv4 dhcp server 1.2.3.4 vrf red-vrf
   ```

5. Enable the wireless profile policy.

   Example

   ```
   Device(config-wireless-policy)# no shutdown
   ```

6. Return to privileged EXEC mode.

   Example

   ```
   Device(config-wireless-policy)# end
   ```

The DHCP server VRF is now configured separately from the client VRF, enabling proper DHCP relay functionality between different VRF domains.

VRF support verification

Lists commands to verify VRF support and check VRF and client overlap IP addresses.

To verify the VRF support, use these commands:

```
Device# show wireless client mac-address aaaa.facc.cccc detail

Client MAC Address : MAC XX
Client MAC Type : Universally Administered Address
Client DUID: NA
Client IPv4 Address : IP XX
Client Username : vrf1
AP MAC Address : aaa.bbb.ccc
AP Name: VRF_TEST_AP
AP slot : 1
Client State : Associated
Policy Profile : VRF_TEST_1
Flex Profile : N/A
Wireless LAN Id: 111
WLAN Profile Name: VRF_TEST_1
Wireless LAN Network Name (SSID): VRF_TEST_1
BSSID : aaa.bbb.ccc
Connected For : 543 seconds
Protocol : 802.11ax - 5 GHz
Channel : 44
Client IIF-ID : xxx
Association Id : 2
Authentication Algorithm : Open System
Idle state timeout : N/A
Re-Authentication Timeout : 43200 sec (Remaining time: 42670 sec)
Session Warning Time : Timer not running
Input Policy Name  : None
Input Policy State : None
Input Policy Source : None
Output Policy Name  : None
Output Policy State : None
Output Policy Source : None
WMM Support : Enabled
U-APSD Support : Disabled
Fastlane Support : Disabled
Client Active State : Active
Power Save : OFF
Current Rate : m11 ss2
Supported Rates : 24.0,36.0,48.0,54.0
AAA QoS Rate Limit Parameters:
QoS Average Data Rate Upstream             : 0 (kbps)
QoS Realtime Average Data Rate Upstream    : 0 (kbps)
QoS Burst Data Rate Upstream               : 0 (kbps)
QoS Realtime Burst Data Rate Upstream      : 0 (kbps)
QoS Average Data Rate Downstream           : 0 (kbps)
QoS Realtime Average Data Rate Downstream  : 0 (kbps)
QoS Burst Data Rate Downstream             : 0 (kbps)
QoS Realtime Burst Data Rate Downstream    : 0 (kbps)
Mobility:
Move Count                  : 0
Mobility Role               : Local
Mobility Roam Type          : None
Mobility Complete Timestamp : 01/09/2025
Client Join Time:
Join Time Of Client : 01/09/2025
Client State Servers : None
Client ACLs : None
```

```
Policy Manager State: Run
Last Policy Manager State : IP Learn Complete
Client Entry Create Time :
Policy Type : WPA2
Encryption Cipher : CCMP (AES)
Authentication Key Management : 802.1x
Transition Disable Bitmap : None
User Defined (Private) Network : Disabled
User Defined (Private) Network Drop Unicast : Disabled
Encrypted Traffic Analytics : No
Protected Management Frame - 802.11w : No
EAP Type : PEAP
VLAN Override after Webauth : No
VLAN : VRF_TEST_1
Multicast VLAN : 0
VRF Name : VRF_TEST_1
```

```
Device# show wireless device-tracking database mac

  MAC             VLAN  IF-HDL      VRF-Name   IP

-----------------------------------------------------------------------------------------

  6c40.088c.a452  16    0x9040000e  red-vrf     9.10.16.64
```

```
Device# show wireless profile policy detailed test

Policy Profile Name                 : test_vrf
Description                         :
Status                              : ENABLED
VLAN                                : 1
Multicast VLAN                      : 0
OSEN client VLAN                    :
Wireless management interface VLAN  : 40
Multicast Filter                    : DISABLED
QBSS Load                           : ENABLED
Passive Client                      : DISABLED
ET-Analytics                        : DISABLED
StaticIP Mobility                   : DISABLED
WLAN Switching Policy
Flex Central Switching              : ENABLED
Flex Central Authentication         : ENABLED
Flex Central DHCP                   : ENABLED
Flex NAT PAT                        : DISABLED
WLAN Flex Policy
VLAN based Central Switching        : DISABLED
WLAN ACL
IPv4 ACL                            : Not Configured
IPv6 ACL                            : Not Configured
Layer2 ACL                          : Not Configured
Preauth urlfilter list              : Not Configured
Postauth urlfilter list             : Not Configured
WLAN Timeout
Session Timeout                     : 28800
Idle Timeout                        : 300
Idle Threshold                      : 0
Guest LAN Session Timeout           : DISABLED
WLAN Local Profiling
Subscriber Policy Name              : Not Configured
```

```
RADIUS Profiling                        : DISABLED
HTTP TLV caching                        : DISABLED
DHCP TLV caching                        : DISABLED
DOT11 TLV accounting                    : DISABLED
CTS Policy
Inline Tagging                          : DISABLED
SGACL Enforcement                       : DISABLED
Default SGT                             : 0
WLAN Mobility
Anchor                                  : DISABLED
AVC VISIBILITY                          : Disabled
IPv4 Flow Monitors
Ingress
Egress
IPv6 Flow Monitors
Ingress
Egress
NBAR Protocol Discovery                 : Disabled
Reanchoring                             : Disabled
Classmap name for Reanchoring
Reanchoring Classmap Name               : Not Configured
QOS per SSID
Ingress Service Name                    : Not Configured
Egress Service Name                     : Not Configured
QOS per Client
Ingress Service Name                    : Not Configured
Egress Service Name                     : Not Configured
Umbrella information
Cisco Umbrella Parameter Map            : Not Configured
DHCP DNS Option                         : ENABLED
Mode                                    : ignore
Autoqos Mode                            : None
Call Snooping                           : Disabled
Tunnel Profile
Profile Name                            : Not Configured
Calendar Profile
Fabric Profile
Profile Name                            : Not Configured
Accounting list
Accounting List                         : Not Configured
DHCP
required                                : DISABLED
server address                          : 1.2.3.4
Opt82
DhcpOpt82Enable                         : DISABLED
DhcpOpt82Ascii                          : DISABLED
DhcpOpt82Rid                            : DISABLED
APMAC                                   : DISABLED
SSID                                    : DISABLED
AP_ETHMAC                               : DISABLED
APNAME                                  : DISABLED
POLICY TAG                              : DISABLED
AP_LOCATION                             : DISABLED
VLAN_ID                                 : DISABLED
VRF_NAME                                : ENABLED
.
.
.
```

To check VRF and client overlap IP address, use these commands:

```
Device# show wireless device-tracking database mac

MAC VLAN IF-HDL IP ZONE-ID/VRF-NAME
-----------------------------------------------------------------------------------------------
6038.e0dc.317e 172 0x90400004 172.172.172.254 red-vrf

60f8.1dce.39b0 173 0x90000006 172.172.172.254 blue-vrf
```

```
Device# show wireless cli summary detail

Number of Clients: 2

MAC Address    SSIDAP Name State IP Address         Device-type VLAN VRF Name
  BSSID           Auth Method  Created
-------------------------------------------------------------------------------------------
6038.e0dc.317e UI_172 AP9120  Run  172.172.172.254 172                red-vrf
    7c21.0d31.dcef [PSK]        02:09:08
60f8.1dce.39b0 UI_173 AP2702I Run  172.172.172.254 173                red-vrf
    80e0.1d81.c64f [PSK]         07:41

Connected Protocol Channel Width SGI NSS  Rate  CAP Username Rx packets Tx
packets Rx bytes Tx bytes   6E capability
-------------------------------------------------------------------------------------------
02:09:11  11n(5)             36   40/40 Y/Y 2/2  m15   E     19214      12028
       2300155  1939782    N
07:44     11ac               36   20/80 Y/Y 3/3  m8ss3 E     29165      25429
       5110                N
```

228 Automated Frequency Coordination

Topics:

- Automated frequency coordination
- AP height configuration through priming profile

Introduces Automated Frequency Coordination technology and guides users through configuring access point height settings via priming profiles for optimal wireless performance.

Automated frequency coordination

Describes an advanced system for coordinating spectrum allocation and utilization for access points operating within the 6-GHz band.

Automated Frequency Coordination (AFC) is an advanced system that

- coordinates the allocation and utilization of spectrum for access points (APs) that operate within the 6-GHz band
- utilizes the Universal Licensing System (ULS) as a regulatory database (encompassing a wide range of frequency bands used by various radio frequency services within a specific region), which is updated daily by the FCC, the regultory body, and
- ensures seamless and efficient management of spectrum resources, enabling optimal utilization of spectrum and minimizing interference among different radio frequency services.

AFC system benefits and operation

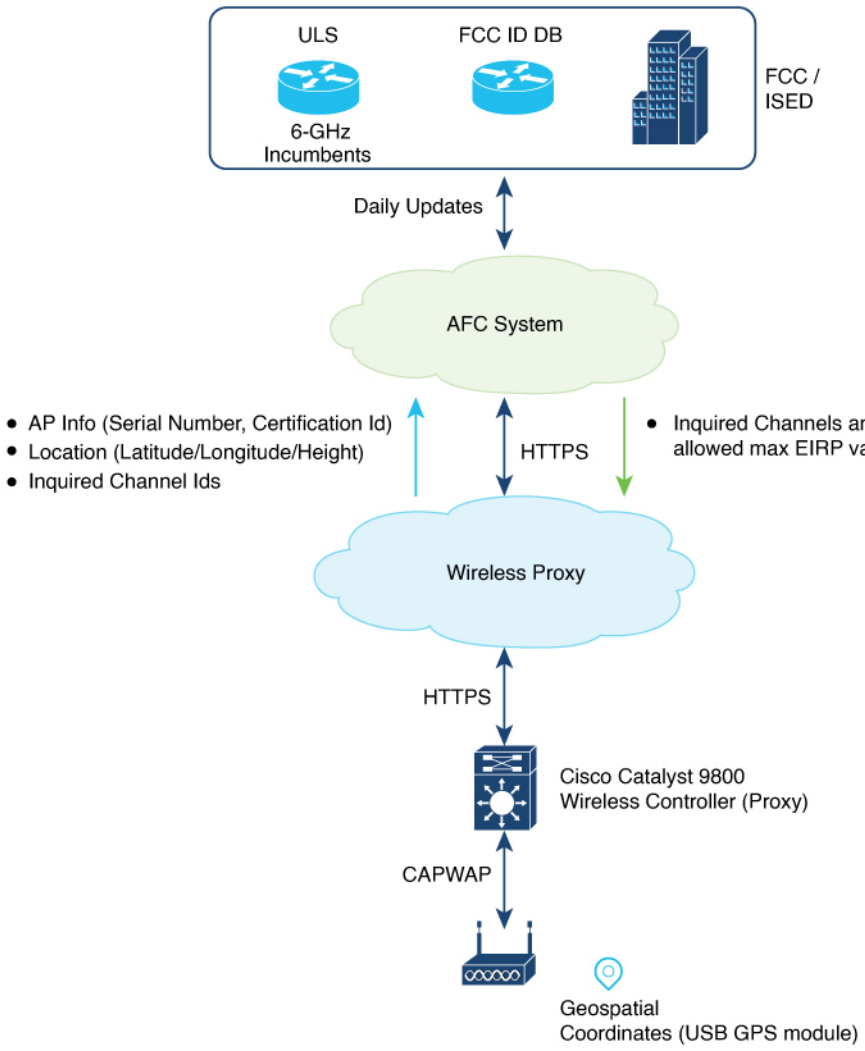
The FCC's decision to open the 6-GHz band for unlicensed Wi-Fi usage brings about significant advantages, including accelerated connectivity and expanded capacity. By unlocking the potential of the 6-GHz band, you can experience unprecedented performance and seamless connectivity, effortlessly engage in data-intensive activities and enjoy immersive online experiences.

Several other technologies, including fixed satellite services (FSS) used in the broadcast and cable industries, are already active in the 6-GHz band. To ensure that the new unlicensed Wi-Fi entrants do not interfere with current services, FCC has implemented the AFC system for Wi-Fi operation within this band. Indoor APs operate at reduced power levels and are less likely to interfere with current 6-GHz users. The outdoor or standard power APs have a higher probability of causing interference with the existing 6-GHz users. These APs are permitted to operate only within the frequency ranges allocated to each country. For example, in the US, APs are authorized to function in the 5.925-6.425 GHz and 6.525-7.125 GHz bands.

Under the AFC system, a new wireless device is required to verify its compatibility with the existing services. This is done by accessing a registered database of the local AFC system provided by an AFC service provider before starting data transmission. This compatibility check ensures that the device's operation does not cause any interference with the current services. To facilitate the compatibility check, the AFC provider maintains an extensive database containing information about the existing 6-GHz operators, including geolocation details, frequencies in use, power levels, antenna coverage, and other relevant data. For areas where an AFC service is available, outdoor or standard power 6-GHz Wi-Fi deployments are only authorized if they comply with the AFC guidelines.

Individual standard power APs are exempt from interfacing directly with the AFC system, if the necessary registration data is communicated by a proxy device such as the wireless controller. Still, APs are responsible for providing their precise location information. In return, they receive a designated set of available frequencies suitable for that specific location.

Figure 43: AFC architecture



AFC architecture includes these components:

- Cisco Catalyst APs supporting 6-GHz
- Cisco Catalyst 9800 Series Wireless Controller acting as AFC proxy
- A cloud-based wireless proxy tunneling the communication between the AFC proxy and the AFC system
- AFC System

The 6-GHz AFC workflow operates as follows:

1. A standard power AP joins the system. Before enabling standard power, the AP must get the available frequencies and the power in each frequency range from the AFC system.
2. The AFC proxy sends the AP information to the AFC system.
3. The AFC system computes the available frequencies and maximum allowable power based on the information provided by the regulatory body (FCC for United States).
4. The response is sent back to controller, which may assign a standard power channel to the AP based on the allowed channel list returned by the AFC system.

Supported APs

These Wi-Fi 7 APs support AFC:

Table 196: Supported APs

AP Model	9136I	9166I	9163E	9166D1	9164I	9162I	9178I	9176D1	9172I	9179F	9171I	9174I	9174E
First Supported Release	17.13.1, 17.12.3	17.13.1, 17.12.3	17.13.1, 17.12.3	17.14.1, 17.12.3	17.14.1, 17.12.3	17.14.1, 17.12.3	17.15.2	17.15.2	17.15.2, 17.17.1	17.18.1	17.18.2	17.18.2	17.18.2
6GHz/6E slots in AP	3	2	2	2	2	2	3	2	2	3	1	2	2
LPI and SP Support	LPI, SP	LPI, SP	SP only	LPI, SP	LPI, SP	LPI, SP	LPI, SP	LPI, SP	LPI, SP	LPI, SP	LPI, SP	LPI, SP	SP only
AP Type	Indoor	Indoor	Outdoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor	Indoor/Outdoor	Indoor	Indoor	Indoor
GNSS	External	External	External	External	External	External	Internal	Internal	External	Internal	External	External	External

Note

The Cisco Wireless 9178I, 9176I, 9176D1, and 9172I Wi-Fi 7 Access Points are not applicable for 17.16.1 release.

Standard power enablement using AFC for US and Canada

Table 197: AFC power enablement: US and Canada

Country	Regulatory Body	Supported Bands	Power Levels	Supported Release
United States (US)	FCC	UNII-5, UNII-7	PSD 23dBm/MHz EIRP 36dBm	17.12.3 onwards
Canada (CA)	ISED	UNII-5, UNII-6, and UNII-7	PSD 23dBm/MHz EIRP 36dBm	17.17.1 onwards

> **Note**
>
> AFC is applicable or active for FCC (US) and Canada. You can verify this using the **show wireless afc ap** and **show wireless afc statistics** commands. In countries where AFC is not required, these commands will show no queries or responses for any access points, indicating that AFC is inactive inthat region.

Feature history for automated frequency coordination

Describes the release history and supported access points for the Automated Frequency Coordination feature, which manages spectrum allocation for 6-GHz band operations.

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 198: Feature history for automated frequency coordination

Release	Feature	Feature Information
Cisco IOS XE 17.12.3	Automated Frequency Coordination	Automated Frequency Coordination (AFC) is an advanced system for coordinating the allocation and utilization of spectrum for access points (APs) that operate within the 6-GHz band. This feature is supported on the following APs: • Cisco Catalyst Wireless 9136I Access Point • Cisco Catalyst Wireless 9163E Access Point • Cisco Catalyst Wireless 9166D Access Point • Cisco Catalyst Wireless 9166I Access Point • Cisco Catalyst Wireless 9162I Access Point • Cisco Catalyst Wireless 9164I Access Point • Cisco Catalyst Wireless 9167IW IIoT Access Point

Prerequisites for Automated Frequency Coordination

These are the prerequisites for Automated Frequency Coordination:

- Ensure that there is cloud connectivity from the controller to the cloud, with a DNS entry in place. AFC operates through either the management port or data ports.

  The AFC request is sent only when the controller is onboarded with cloud. This is automatic for hardware platforms like 9800-80, 9800-40 and 9800-L. For cloud controller, you have to manually enter a one-time password (OTP). See Onboarding the Cloud Controller.

- Before sending an AFC request, check whether the AFC service can be requested by using the show wireless afc ap command. If command output shows yes or up status for all the parameters of an AP, then request is sent out.
- Standard APs must register with the AFC system by providing these parameters:
  - Geographic coordinates (latitude and longitude)
  - Antenna height above ground level and tolerance as uncertainty height
  - FCC identification number
  - Manufacturer's unique serial number
  - Firewall Requirements - Ensure your network firewall allows outbound traffic to:
    - dnaservices.cisco.com on port 443
    - commercial.ocsp.identrust.com on port 80

Restrictions for Automated Frequency Coordination

These are the restrictions for AFC:

- AFC is not supported on Embedded Wireless Controller (EWC).

Onboard the cloud controller (GUI)

Configure the cloud controller onboarding process by generating and importing a one-time password (OTP).

Onboarding the cloud controller requires the generation of a one-time password (OTP). Controller must register with cloud framework to access any cloud service. The cloud service provides identity management services to the controller and securely authenticates the device.

> **Note**
>
> These steps apply only to the Cisco Catalyst 9800-CL Wireless Controller for Cloud, and are not applicable to the hardware-based Catalyst 9800 Wireless Controllers.

Follow these steps to onboard the cloud controller:

Procedure

1. Go to **Cisco DNA portal**.
2. Click **Application/Products**.
3. Select **Region** using the drop-down and click Register.
4. Click **Register**.
5. Enter the product details such as **Host Name/IP**, **Product Name**, and **Description**.

6. Click **Register**.

   This generates an OTP.

   The OTP token is a BASE64 encoded JSON payload. A sample OTP token is given below:

   Example

   ```
   eyJiYXNlX3VybCI6ICJodHRwczovL21hZ2xldi5tYWdsZXZjbG91ZDMudGVzc2VyYWN0
   aW50ZXJuYWwuY29tIiwib3RwIjogImNjNDY3ZDA5NzE1ZjQ3ZWI4NzhhN2UyOTA2YjllNzE0MTYzY
   mQzZjA1MWExNDUxZjg2ODViMjVhZDhmNWNkMTQifQo=
   ```

7. Copy the OTP.
8. Go to the controller console.
9. Run **cloud otp token import** command.
10. Paste the OTP copied in **Step 7** to the console.

This completes the onboarding procedure. Wait for a few minutes to complete the process.

Configure DNA services (GUI)

Configure DNA Service using GUI.

Set up integration with the Cisco DNA Cloud by configuring service connection parameters and authentication details for secure operation.

Procedure

1. Choose **Configuration > Services > Cloud Services**.
2. Click **DNA Services Configuration** tab.

   The DNA Services Configuration section displays these details:

   > **Note**
   >
   > We recommend that you retain default values for URL, API endpoint path and certificate bundle path.

   - **URL**: The base web address of the Cisco DNA Cloud instance that issued the token. This will be used to address the enrollment request.
   - **API Endpoint Path**: An API path that allows client applications like AFC to register for notifications for any Cloud state change.
   - **Certificate Bundle Path**: The path for certificate-based authentication. In addition to OTP-based authentication, you can use certificate-based authentication.
   - **HTTP Proxy (Hostname/IP)**: The HTTP proxy that serves incoming HTTPs requests from clients and processes received HTTPs responses.

3. In the **DNA OTP Configuration** section and under **Generate OTP**, select the link provided to navigate to the Cisco DNA website to generate an OTP.
4. Enter the generated OTP in the **OTP Token** field.

5. Click **Submit OTP**.

DNA services are configured. The system establishes a secure connection with the Cisco DNA Cloud, enabling authorized access to cloud-managed services and device management features.

Configure power mode per RF profile (CLI)

Configure Power Mode per RF Profile using commands.

Allow or disallow standard power mode in configuration per RF profile for 6-GHz band APs.

You can allow or disallow standard power mode in configuration per RF profile. However, the operating mode for each RF profile is determined based on the configuration, the capabilities of the AP, and the response from the AFC system regarding channel and transmit power values. If there is no connection with the AFC system or if the received Tx power values are extremely low, the AP might automatically switch from standard-power mode to low-power mode as a fallback.

If the standard power granted by the AFC server is greater than the low power, the radio is switched to standard power mode in the case of mixed-mode APs. If the granted standard power is equal to or greater than the low power, the radio is switched to standard power mode.

Note

As outdoor APs operate only in standard power mode, the RF profile configuration for the standard power mode is not evaluated.

Note

We recommend using standard power indoors for unparalleled speed and coverage. While the client and vendor ecosystem is evolving, the current Cisco standard power technology continues to provide advantages for existing standard power and dual-mode clients. Low-power clients will default to 2.4/5GHz until they are fully upgraded to standard power by the client vendors.

Procedure

1. Enter global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the RF profile and enter RF profile configuration mode.

Example

```
Device(config)# ap dot11 6ghz rf-profile rf-profile-name
```

Example

```
Device(config)# ap dot11 6ghz rf-profile prof-afc
```

3. Configure standard-power mode for 6 GHz band for APs that are capable of low power (LP) and standard power (SP).

   Example

   ```
   Device(config-rf-profile)# tx-power standard
   ```

   APs that support only SP mode are not affected by this setting (SP mode is always enabled).

4. Return to privileged EXEC mode.

   Example

   ```
   Device(config-rf-profile)# end
   ```

The RF profile is configured with standard power mode for the 6 GHz band, allowing APs to operate in standard power mode when conditions permit.

Configure power mode per RF profile (GUI)

Configure power mode per RF profile using the GUI to allow dual-power mode APs to operate as standard power APs in the 6 GHz spectrum.

This task enables standard-power service for dual-power mode APs to operate as standard power APs in the 6 GHz spectrum, allowing access to external AFC service through cloud.

The AP can access the external AFC service through cloud. Based on geographical coordinates and spectrum-inquiry requests from APs, the AFC provides responses on the available frequencies and the maximum permissible power in each frequency range.

Procedure

---

1. Go to **Configuration > Tags & Profiles > RF/Radio**.
2. Click **RF** tab.

   To add a new RF profile, see *Configuring an RF Profile (GUI)*. To modify an existing RF profile, select the required RF profile.

3. Enable **Standard-Power Service** to allow dual-power (low power or standard power) mode APs to operate as standard power APs in the 6 GHz spectrum.

   The AP can access the external AFC service through cloud. Based on geographical coordinates and spectrum-inquiry requests from APs, the AFC provides responses on the available frequencies and the maximum permissible power in each frequency range.

> **Note**
>
> - This setting is available for the 6-GHz band radio and standard-power supported APs only.
> - DNA services need to be configured to connect to AFC. You can configure DNA services in **Configuration > Services > Cloud Services > DNA Services** page.

4. Click **Apply to Device**.

Dual-power mode APs are now configured to operate as standard power APs in the 6 GHz band. They can access external AFC service through the cloud, with available frequencies and maximum permissible power determined by AFC responses.

Configure AP parameters (GUI)

Configure AP parameters using GUI.

AFC requires the longitude, latitude, and height of the AP to determine the geolocation of the AP. Longitude and latitude of the AP will be automatically calculated based on one of the following methods:

- From an internal or external GPS attached to the AP.
- From an another AP with GPS (also known as anchor AP) and the distance between the two APs.

> **Note**
>
> Only a few APs on a floor require the GPS unit. The remaining APs can derive their location from the AP with the GPS unit.
>
> You can use bulk AP provisioning to configure multiple AP parameters for multiple APs simultaneously. GUI path is **Configure > Wireless > Bulk AP Provisioning**.

Procedure

---

1. Go to **Configuration > Wireless > Access Points**.
2. In the **All Access Points** section, select a supported AP.
3. In the **Geolocation** tab, enter these details:
   a) In the **Height Configuration** section, enter the above ground level height in meters. The range is from -100 to 1000. Enter the tolerance as uncertainty height in meters. The range is from 1 to 100.
   b) In the **GNSS External Antenna Configuration** section, enter the cable length in meters.The range is from 1 to 100, with a default of 10. We recommend that you keep the default value of 10 meters with the Cisco provided external antenna.

> **Note**
>
> This option is available on selected models that support an external antenna.

4. Click **Apply to Device**.

The AP geolocation and antenna parameters are updated. APs without direct GPS will derive their location from anchor APs as needed. Bulk provisioning allows you to configure multiple APs efficiently.

Configure AP parameters (CLI)

Configure the height and GNSS antenna details of the AP.

Set up AP geolocation parameters including height and GNSS antenna configuration for accurate positioning.

To configure the height and GNSS antenna details of the AP, use the following procedure:

Procedure

1. Enable privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

2. Configure the AP Above Ground Level (AGL) height in meters.

   Example

   ```
   Device# ap name ap-name geolocation height height uncertainty uncertainty
   ```

   Example

   ```
   Device# ap name cisco-ap1 geolocation height 100 uncertainty 10
   ```

   The value range for height is -100 to 1000 meters. The valid range for height uncertainty is 1 to 100 meters.

3. Optional: Configure AP GNSS antenna external cable length, in meters.

   Example

   ```
   Device# ap name ap-name geolocation gnss antenna external cable-length length
   ```

   Example

   ```
   Device# ap name cisco-ap1 geolocation gnss antenna external cable-length 10
   ```

   The value range for cable length is 1 to 100, with a default value of 10.

> **Note**
>
> This configuration step is required only if GNSS is using an external antenna.

The AP geolocation parameters are now configured with the specified height and GNSS antenna settings.

AFC details verification

Lists the commands to verify AFC details and configuration information.

Use the verification commands to check AFC statistics, channel information, request status, responses, AP geolocation data, and power mode settings for troubleshooting and monitoring AFC functionality on 6 GHz APs.

AFC statistics verification

To see AFC statistics information, use this command:

```
Device# show wireless afc statistics

Total number of 6GHz APs : 4
Number of APs requiring AFC service : 4
Messages sent to AFC : 229
Successful messages received from AFC : 229
Errored AFC messages : 0
AFC messages pending : 0
Last InquiredChannel message sent:
requestId : 12195125900336565222
AP MAC : 10f9.20fd.54e0
Sent timestamp : 08/09/2023 15:14:20
Last InquiredChannel message received:
requestId : 12195125900336565222
AP MAC : 10f9.20fd.54e0
Received timestamp : 08/09/2023 15:14:21
Minimum response time (ms) : 337
Maximum response time (ms) : 40842
Average response time (ms) : 1315
Health check query : Idle
Health check status : OK
Health check timestamp : 08/09/2023 14:58:50
Number of times health check went down : 0

Health check event history
Timestamp #Times Event State RC Context
---------------------------- -------- ----------------------
---------------------------- --- ---------------
08/09/2023 14:58:50.348063 53 Response received OK 0
08/09/2023 14:58:48.271529 58 Scheduled 0 Timer: 3600s
08/09/2023 14:58:48.271507 53 Sent 0
08/07/2023 09:33:55.990412 6 Not sent No token 0
```

AFC channel information

To see information of AFC channels, use this command:

```
Device# show wireless afc channels 20mhz

802.11 6ghz    :  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1  1
1  1  1  1  1
Channel IDs    :  1  1  2  2  2  3  3  4  4  4  5  5  6  6  6  7  7  8  8  8
9  9  0  0  0
(20MHz width) :  1  5  9  3  7  1  5  9  3  7  1  5  9  3  7  1  5  9  3  7
1  5  9  3  7
----------------:+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--+--
AP687D.B45C.08F0 : 36 33 36 36 35 36 36 36 26 27 36 36 36 36 36 36 33 33 36
36 36 36 36 36 36
AP687D.B45C.1824 : 36 33 36 36 35 36 36 36 24 26 27 36 36 36 36 36 33 33 36
36 36 36 36 36 36

! Due to space constraints, the channel IDs (for 2-digit and 3-digit channel
IDs)
! are given vertically in the output. The output is also truncated to fit the
 page width.
```

AFC request status

To see status of the request sent to AFC, use this command:

```
Device# show wireless afc request

--------------------------------------------------------------------------------------------------

                                  Last AFC Request Sent to AFC Service:

AP Name               Radio MAC      Request Id           AFC Request Status
  Status Timestamp
--------------------------------------------------------------------------------------------------
APCC9C.3EF1.1620      10f9.20fe.36a0 1219512590033656 5035 Response Received
  08/07/2023 13:20:39
AP687D.B45C.17AC      687d.b45f.1af0 12195125900336565037 Response Received
  08/07/2023 16:07:01
AP687D.B45C.321B      fc58.9a18.c840 12195125900336566836 Sent
  08/07/2023 17:36:11
AP687D.B45C.321F      fc58.9a18.c850 12195125900336566832 Timeout
  08/07/2023 17:36:08
.
.
.
```

AFC response information

To see AFC response to requests, use this command:

```
Device# show wireless afc response

------------------------------------------------------------------------------
AP Name               Radio MAC         Request Id  Expire Time            Last
Rcvd Time        Response Code
------------------------------------------------------------------------------
AP687D.B45C.16A4     687d.b45f.0e90    1            11/14/2021 21:56:29
```

```
11/14/2021 17:56:29  SUCCESS
AP687D.B45C.17AC    687d.b45f.1af0   2           11/14/2021 21:56:30
11/14/2021 17:56:29  SUCCESS
AP687D.B45C.2276    687d.b45f.3190   3           11/14/2021 21:56:30
11/14/2021 17:56:29  INVALID VALUE

! See the Response Code column.
```

AFC service request capability

To check whether AFC service request can be sent, use this command:

```
Device# show wireless afc ap

------------------------------------------------------------------------------------------------------------
AP Name             Radio MAC         AFC          Power Mode  Current
AP       6GHz Radio   RF-Profile   RF-Profile    AFC Country  Location  Height

                                      Status       Capability  Power Mode  Admin
 State  Admin State  Admin State  tx-power std  allowed       known      known

------------------------------------------------------------------------------------------------------------

APCC9C.3EF1.1620   10f9.20fe.36a0   Inactive    SP/LPi       LPi          Up
        Down          Yes           Yes           Yes            Yes       Yes

AP687D.B45C.17AC   687d.b45f.1af0   Active      SP/LPi       SP           Up
        Up            Yes           Yes           Yes            Yes       Yes

AP-ARCTIC          fc58.9a18.c890   Inactive    SP/LPi       LPi          Up
        Up            Yes           Yes           Yes            Yes        No
```

AFC geolocation information

To see AFC geolocation information to be used in an AFC Request, use this command:

```
Device# show wireless afc geolocation

------------------------------------------------------------------------------------------------------------
AP Name             Radio MAC      Location    Longitude   Latitude
Major-axis Minor-axis Orientation Area-of-uncert Height Height   Uncertainty
                                  Type        (degrees)   (degrees)    (meters)
   (meters)   (degrees)   (sq. meters)   Type   (meters) (meters)
------------------------------------------------------------------------------------------------------------
AP687D.B45C.16A4   687d.b45f.0e90 Lin Polygon -122.400140 37.794910
                            1122           AGL    150      6
                                                          -122.399340
37.795020
                                                          -122.399180
37.794390
                                                          -122.400040
37.794270
AP687D.B45C.17AC   687d.b45f.1af0 Ellipse     -73.977760  40.760168    9
     9          5.340000    254            AGL    129      3


! The afc geolocation information given in the output refers to coordinates
to be used for future AFC requests.
```

```
! They do not represent the coordinates used by past AFC requests; for such
info see AFC responses in 'show wireless afc response' command.
.
.
.
```

AFC geolocation information of a Cisco AP

To see geolocation information of a Cisco AP, use this command:

```
Device# show ap name AP687D.B45C.17AC afc geolocation

Location type                 : Ellipse
Center ellipse - longitude    : -73.977769
Center ellipse - latitude     : 40.760168
Ellipse major-axis (meters)   : 8
Ellipse minor-axis (meters)   : 8
Ellipse orientation           : 2.5
Height (meters)               : 129
Uncertainty (meters)          : 3
```

AFC response for an AP service request

To see the response from AFC for a service request from an AP, use this command:

```
Device# show ap name AP687D.B45C.17AC afc response

AP name                 : AP687D.B45C.17AC
AP MAC Address          : 687d.b45f.1af0
Response Code           : SUCCESS
Request ID              : 2
Expire Time             : 11/14/2021 21:56:30
Last Rcvd Time          : 11/14/2021 17:56:29
Global Operating Class : 131
    Channel Cfi / Max Eirp :
        1       / 36.000000
        5       / 33.668693
        9       / 36.000000
        13      / 36.000000
        17      / 35.223532
        21      / 36.000000
        25      / 36.000000
        29      / 36.000000
        33      / 26.664962
        37      / 27.799478
        41      / 36.000000
        45      / 36.000000
        49      / 36.000000
        53      / 36.000000

.
.
.
```

RF profile details

To see the details of an RF profile, use the following command:

```
Device# show ap rf-profile name default-rf-profile-6ghz detail

Description                          : default rfprofile for 6GHz radio
RF Profile Name                      : default-rf-profile-6ghz
Band                                 : 6 GHz
Transmit Power Threshold v1          : -70 dBm
Min Transmit Power                   : -10 dBm
Max Transmit Power                   : 30 dBm
Operational Rates
  802.11 6GHZ 6M Rate                   : Mandatory
  802.11 6GHZ 9M Rate                   : Supported
  802.11 6GHZ 12M Rate                  : Mandatory
  802.11 6GHZ 18M Rate                  : Supported
  802.11 6GHZ 24M Rate                  : Mandatory
  802.11 6GHZ 36M Rate                  : Supported
  802.11 6GHZ 48M Rate                  : Supported
  802.11 6GHZ 54M Rate                  : Supported
Max Clients                          : 200
Trap Threshold
  Clients                            : 12 clients
  Interference                       : 10%
  Noise                              : -70 dBm
  Utilization                        : 80%
Multicast Data Rate                  : auto
Rx SOP Threshold                     : auto
Load Balancing
  Window                             :  5 clients
  Denial                             :  3 count
Coverage Data
  Data                               : -80 dBm
  Voice                              : -80 dBm
  Minimum Client Level               : 3 clients
  Exception Level                    : 25%
RSSI Settings
  RSSI Low Check                     : Disabled
  RSSI Threshold                     : -127 dbm
DCA Channel List                     :
Unused Channel List                  :
PSC Channel List                     :
DCA Bandwidth                        : best
DBS Min Channel Width                : 20 MHz
DBS Max Channel Width                : MAX ALLOWED
DCA Foreign AP Contribution          : Enabled
State                                : Up
  Client utilization threshold  : 5%
  Client Reset count            : 1
Client Network Preference            : default
802.11ax
  OBSS PD                            : Disabled
  Non-SRG OBSS PD Maximum            : -62 dBm
  SRG OBSS PD                        : Disabled
  SRG OBSS PD Minimum                : -82 dBm
  SRG OBSS PD Maximum                : -62 dBm
  Broadcast Probe Response           : Disabled
  FILS Discovery                     : Disabled
  Multi-BSSID Profile Name           : default-multi-bssid-profile
NDP mode                             : Auto
Guard Interval                       : 800ns
PSC Enforcement                      : Disabled
```

```
Standard-Power mode               : Allowed
```

AFC power-mode of a Cisco AP

To see AFC power-mode of a Cisco AP, use the following command:

```
Device# show ap name AP687D.B45C.17AC dot11 6ghz power-mode

Standard-power mode               : Allowed
```

802.11 parameter configuration of a Cisco AP

To see the 802.11 parameter configuration of a Cisco AP, use the following command:

```
Device# show ap name AP687D.B45C.1908 config dot11 6ghz

AP 6GHZ Power Mode                              : Low Power Indoor
Low Power Indoor Capable                        : Yes
Standard Power Capable                          : Yes
```

AFC event details

To see detailed information about AFC events for a Cisco AP, use the following command:

```
Device# show ap name AP687D.B45C.1908 afc detail

AFC event history

Timestamp                    Event                       AFC Status     Context

---------------------------- --------------------------- --------------
----------------------------------------
08/07/2023 16:07:01.244077   AFC_EVT_REQ_RESP_RECEIVED   Active         ReqId
 12195125900336565037 Response code: 0; SUCCESS
08/07/2023 16:06:55.491957   AFC_EVT_REQ_SENT            Inactive       ReqId
 12195125900336565037 Sent to AFC System
08/07/2023 16:06:51.823938   AFC_EVT_REQ_QUEUED          Inactive       AFC
request queued: immediate scheduling
08/07/2023 15:28:54.608117   AFC_EVT_REQ_INACTIVE        Inactive       No
location information available; No height information available;
```

AP height configuration through priming profile

Describes how to configure AP height through priming profile for AFC 6-GHz channel availability requests.

AP height configuration through priming profile is a provisioning method that

- enables configuration of the Z coordinate (height) for multiple access points through a centralized profile approach
- provides pre-provisioning capabilities for seamless addition of new APs to the network, and
- ensures guaranteed configuration push when APs join the controller, overcoming the limitations of individual AP height configuration in Privileged EXEC mode.

Height configuration methods comparison

Until Cisco IOS XE 17.14.1, Automated Frequency Coordination (AFC) used geolocation attributes (X,Y, and Z) for an AP to be determined, to request for 6 GHz channel availability from the AFC server, for a given location. As per the regulatory requirement, the X and Y coordinates are obtained autonomously through GPS or GNSS directly, or through derivation. The Z coordinate (mandatory parameter), which refers to the height, is configured manually. In case of the controller, the height is programmed per AP, in the Privileged EXEC mode.

In this approach, the AP is joined, in order to configure height, and additional parameters that are configured in the Privileged EXEC mode.

However, following are the restrictions to the current implementation:

- There is no guarantee that all the APs will be configured with height. APs may flap during configuration, failing to set the height.
- Provisioning height individually for an AP in a large scale deployment could become cumbersome.
- The height once configured is persistent. Therefore, if you move an AP across floors, you must reprovision height as well.
- The AP height should be reprovisioned when a new AP is added to the network.

To overcome these constraints, from Cisco IOS XE 17.15.1 onwards, AP height is configured through priming profile.

These are the advantages when you configure the height of the AP through priming profile:

- Easy to update the height for a group of APs
- Pre-Provisioning feature
- Guaranteed configuration push when AP joins
- Moving around APs across different floors inherit the consistent configurations based on the filter
- Seamless addition of new APs

Guidelines for AP height configuration through priming profile

Outlines guidelines for AP height configuration through priming profile.

Follow these guidelines when configuring AP height through priming profile:

- If the height attribute is configured under the priming profile, the Priming Override attribute replaces the already configured height of the AP.
- The Priming Override attribute is unique for each priming profile, therefore, all the attributes within the priming profile are overwritten.
- Removal of height related configuration through priming profile has no effect on the configured height, but the configuration attributes are removed. The last configured height is retained.
- During an upgrade, if the height is not configured in the priming profile, it does not replace the already configured height.

Configure AP height through priming profile (GUI)

Configure the height and height uncertainty for access points through the AP priming profile to enable geolocation services.

Use this task to set the height and height uncertainty values for access points, which are required for geolocation services. The height is measured in relation to the main ground surface (Above Ground Level - AGL).

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **AP Priming**.
2. Click **Add**.

   The **Add New AP Priming Profile** window is displayed.
3. In the **Add New AP Priming Profile** window, complete the following steps:
   a) In the **Profile Name** field, enter the new AP priming profile name.
   b) Enter the **Primary Controller Name**, **Primary Controller IP**, **Secondary Controller Name**, **Secondary Controller IP**, **Tertiary Controller Name**, **Tertiary Controller IP** details.
   c) Check the **Geolocation** check box to configure the geolocation details.
   d) In the **Height** field, enter the height of the AP in meters. The default value is 0.

   > **Note**
   >
   > Configure **Height** together with **Height Uncertainty**.
   >
   > The height of the AP is measured in relation to the main ground surface, also known as Above Ground Level (AGL). Remove height and uncertainty values to clear the height configuration.

   e) In the **Height Uncertainty** field, enter the height uncertainty in meters. The default value is 1.

   > **Note**
   >
   > Configure **Height Uncertainty** together with **Height**.
4. Use the **Priming Override** slider to enable or disable priming override.

   > **Note**
   >
   > When the Priming Override features is enabled, all the attributes configured under priming profile is pushed to AP.
5. Click **Apply to Device**.

The AP priming profile with height configuration is created and applied to the device.

Configure AP height through priming profile (CLI)

Configure AP height parameters using a priming profile to set geolocation coordinates for access points.

Set AP height and uncertainty values through a priming profile to enable accurate geolocation tracking.

Use priming profiles to configure AP height parameters when you need to establish geolocation coordinates for access points in your wireless network deployment.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure a profile to prime APs and enter priming configuration mode.

   Example

   ```
   Device(config)# wireless profile ap priming priming-profile-name
   ```

3. Configure the AP geolocation height and uncertainty, in meters.

   Example

   ```
   Device(config-priming)# geolocation height ap-geolocation-height uncertainty
    height-uncertainty
   ```

   Example

   ```
   Device(config-priming)# geolocation height 372 uncertainty 64
   ```

4. Override the existing priming configuration.

   Example

   ```
   Device(config-priming)# priming-override
   ```

   Note

   The height and uncertainty for an AP is updated only if priming-override is enabled.

The AP priming profile is configured with height parameters and can be applied to access points to set their geolocation coordinates.

Apply priming profile using filters (GUI)

Configure priming profile application to access points using filter-based rules through the graphical user interface.

Configure and apply a priming profile to selected access points efficiently.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.

2. Click the **AP** tab and then click the **Filter** tab.
3. Click **Add**.

   The **Associate Tags to AP** window is displayed.
4. In the **Associate Tags to AP** window, complete the following steps:
   a) In the **Rule Name** field, enter the rule name.
   b) Use the **Active** slider to indicate if the rule is active or not.
   c) In the **Priority** field, set a priority for the active rule. The valid range is between 0 and 1023.
   d) From the **Tag** drop-down list, choose **Priming**.
   e) From the **Priming Profile** drop-down list, choose the priming profile, or, search by navigating to the **Priming Profile configuration** window.
5. Click **Apply to Device**.

The priming profile is applied to access points matching the filter criteria.

Apply priming profile using filter (CLI)

Configure a priming profile using filters to automatically provision access points with specific configurations.

Apply priming profiles to APs using filters to automate the provisioning process based on AP naming patterns.

Priming profiles allow you to pre-configure access points with specific settings before they join the network. Using filters based on AP names enables automatic application of these profiles to matching access points.

Procedure

---

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```
2. Configure a filter to prime APs and enter the AP priming filter configuration mode.

   Example

   ```
   Device(config)# ap filter name ap-filter-name type priming
   ```

   Priming filter is always persistent on the AP.

   > **Note**
   >
   > Assign filter priority after filter creation.
3. Configure a filter based on the AP name regular expression match.

   Example

   ```
   Device(config-ap-pr-filter)# ap name-regex ap-filter-string
   ```

4. Map the priming profile to the filter.

   Example

   ```
   Device(config-ap-pr-filter)# profile priming-profile-name
   ```

5. Return to global configuration.

   Example

   ```
   Device(config-ap-pr-filter)# exit
   ```

6. Configure priority for the named AP filter.

   Example

   ```
   Device(config)# ap filter priority priority-value filter-name filter-name
   ```

   Example

   ```
   Device(config)# ap filter priority 2 filter-name height_filter
   ```

The priming profile is now applied using the filter configuration, and matching access points will automatically receive the specified profile settings when they join the network.

Apply priming profile statically (GUI)

Configure static priming profile associations for access points using the GUI interface.

Establish static associations between access points and priming profiles using the graphical user interface.

Procedure

---

1. Choose **Configuration** > **Tags & Profiles** > **Tags**.
2. Click the **AP** tab and then click the **Static** tab.
3. Click **Add**.

   The **Associate Tags to AP** window is displayed.

4. In the **Associate Tags to AP** window, complete the following steps:
   a) In the **AP MAC Address** field, enter the AP MAC address.
   b) From the **Policy Tag Name** drop-down list, choose the required tag name or search by navigating to the **Policy Tag Configuration** window.
   c) From the **Site Tag Name** drop-down list, choose the required tag name or search by navigating to the **Site Tag Configuration** window.
   d) From the **RF Tag Name** drop-down list, choose the required tag name or search by navigating to the **RF Tag Configuration** window.
   e) From the **Priming Profile** drop-down list, choose the priming profile, or, search by navigating to the **Priming Profile Configuration** window.
5. Click **Apply to Device**.

The static priming profile association is now configured for the selected access point.

Apply priming profile statically (CLI)

Configure a priming profile to be statically applied to an access point using CLI commands.

Apply a priming profile to an AP to configure its initial settings and behavior.

Priming profiles allow you to statically configure APs with specific settings before deployment. This is useful for standardizing AP configurations across your network infrastructure.

Procedure

1. Enter global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the Ethernet MAC address of the AP.

   Example

   ```
   Device(config)# ap H.H.H
   ```

   Example

   ```
   Device(config)# ap CC-XX-3E-XX-FC-XX
   ```

3. Map a priming profile to the AP.

   Example

   ```
   Device(config-ap-tag)# priming priming-profile-name
   ```

   Example

   ```
   Device(config-ap-tag)# priming priming-profile-height
   ```

The priming profile is now statically applied to the specified AP, configuring it with the predefined settings from the profile.

AP priming profile verification

Lists commands to verify AP priming profile configurations and geolocation information.

Verify AP priming profile configurations, including all profiles overview, detailed profile information, AP configuration details, and geolocation data.

Verify all priming AP profiles

To verify all AP priming profiles, use this command:

```
Device# show wireless profile ap priming all
Profile Name                 Primary Controller Name       Primary Controller
 IP     Secondary Controller Name    Secondary Controller IP      Tertiary
Controller Name     Tertiary Controller IP          Height       Uncertainty
       Type        Override
--------------------------------------------------------------------------------
priming_profile_height        8GBXXX                          9.9.XX.XX
                                          0.0.0.0
                     0.0.0.0                          372           64
       AGL           Enabled
```

Verify AP priming profile details

To verify the details of an AP priming profile, use this command:

```
Device# show wireless profile ap priming detailed priming_profile_height
Profile Name                          : priming_profile_height
Primary Controller Name               : 8GBXXX
Primary Controller IP                 : 9.9.XX.XX
Secondary Controller Name             :
Secondary Controller IP               : 0.0.0.0
Tertiary Controller Name              :
Tertiary Controller IP                : 0.0.0.0
Height                                : 372
Uncertainty                           : 64
Type                                  : AGL
Override                              : Enabled
```

Verify AP information

To verify information of a Cisco AP, which includes the AP priming details, use this command:

```
Device# show ap name Cisco-AP conf general | sec Prim
Priming Profile                                 : priming_profile_height
Priming Override                                : Enabled
Priming Source                                  : Filter
Priming Filter name                             : height_filter
```

Verify AP geolocation information

To verify the detailed information of the AP geolocation, use this command:

```
Device# show ap name Cisco-AP geolocation detail
AP Name                          : Cisco-AP
GNSS present                     : No
Location                         : NA
Height type                      : AGL
 Source                          : Manual
 Height (meters)                 : 372
 Uncertainty (meters)            : 64
 Last update                     : 05/22/2024 08:54:34
```

Part XXII

Cisco DNA Service for Bonjour

Topics:

- Cisco DNA Service for Bonjour Solution Overview
- Configuring Local and Wide Area Bonjour Domains
- Configuring Local Area Bonjour for Wireless Local Mode
- Configuring Local Area Bonjour for Wireless FlexConnect Mode
- Configuration Example for Local Mode - Wireless and Wired
- Configuration Example for FlexConnect Mode - Wireless and Wired

229 Cisco DNA Service for Bonjour Solution Overview

Topics:

- Cisco DNA Service for Bonjour Solution

Cisco DNA Service for Bonjour Solution

Explains the Cisco Catalyst Center service for Bonjour solution, including its architecture, service routing, and interoperability across enterprise-grade wired and wireless networks.

A Cisco DNA Service for Bonjour solution is a network service architecture that

- eliminates the single Layer 2 domain constraint and expands service discovery to enterprise-grade wired and wireless networks
- enables mDNS service routing to advertise and discover services across distributed networks, and
- supports interoperability with RFC 6762-based multicast DNS (mDNS) compatible wired and wireless consumer products in enterprise environments.

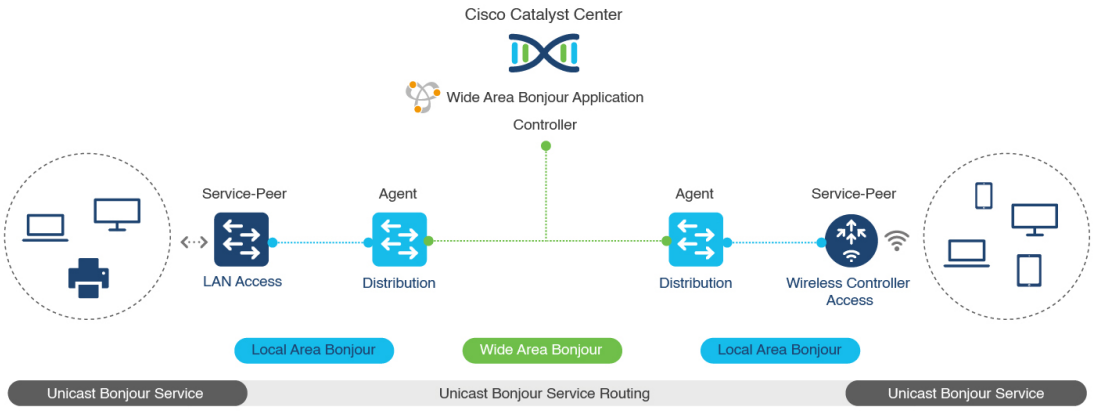
Service routing domains in the Cisco DNA Service for Bonjour solution

The Apple Bonjour protocol is a zero-configuration solution that simplifies rich services and enables intuitive experience between connected devices, services, and applications. Using Bonjour, you can discover and use IT-managed, peer-to-peer, audio and video, or Internet of Things (IoT) services with minimal intervention and technical knowledge. Bonjour is originally designed for single Layer 2 small to mid-size networks, such as home or branch networks. The Cisco DNA Service for Bonjour solution eliminates the single Layer 2 domain constraint and expands the matrix to enterprise-grade traditional wired and wireless networks, including overlay networks such as Cisco Software-Defined Access (SD-Access) and industry-standard BGP EVPN with VXLAN. The Cisco Catalyst 9000 Series LAN switches, Cisco Nexus 9300 Series Switches, and Cisco Catalyst 9800 Series Wireless Controller follow the industry standard, RFC 6762-based multicast DNS (mDNS) specification to support interoperability with various compatible wired and wireless consumer products in enterprise networks.

The Cisco Wide Area Bonjour application on Catalyst Center enables mDNS service routing to advertise and discover services across enterprise-grade wired and wireless networks. The distributed architecture eliminates mDNS flood boundaries and transitions to unicast-based service routing, providing policy enforcement points and enabling management of Bonjour services.

The following figure illustrates how the Cisco Wide Area Bonjour application operates across two integrated service-routing domains.

Figure 44: Cisco Wide Area Bonjour Solution architecture



- **Local Area Service Discovery Gateway Domain - Unicast Mode**: The enhanced Layer 2 unicast policy-based deployment model. mDNS service discovery and distribution using Layer 2 unicast address enables flood-free LAN and wireless networks. Cisco Catalyst 9000 Series Switches and Cisco Catalyst 9800 Series Wireless Controller in Layer 2 mode introduce a new service-peer role, replacing the classic flood-n-learn, for unicast-based service routing support. The service-peer switch and wireless controller replace mDNS flood-n-learn with unicast-based communication with any RFC 6762 mDNS-compatible wired and wireless endpoints.

- **Wide-Area Service Discovery Gateway Domain**: The Wide Area Bonjour domain is a controller-based solution. The Bonjour gateway role and responsibilities of Cisco Catalyst and Cisco Nexus 9300 Series Switches are extended from a single SDG switch to an SDG agent, enabling Wide Area Bonjour service routing beyond a single IP gateway. Distributed SDG agent devices establish a lightweight, stateful, and reliable communication channel with a centralized Catalyst Center controller running the Cisco Wide Area Bonjour application. SDG agents route locally discovered services based on the export policy.

> **Note**
>
> The classic Layer 2 multicast flood-n-learn continues to be supported on wired and wireless networks with certain restrictions to support enhanced security and location-based policy enforcement. The Cisco Catalyst and Cisco Nexus 9300 Series Switches at Layer 3 boundary function as an SDG to discover and distribute services between local wired or wireless VLANs based on applied policies.

Solution components

Explains key components and system roles that enable unicast-based service routing across the local area and Wide Area Bonjour domain.

A solution component is a system element that

- enables unicast-based service routing across the local area and Wide Area Bonjour domain
- includes Cisco Service Peer, Cisco SDG Agent, Catalyst Center controller, and endpoints, and
- supports integration with RFC 6762-compliant Bonjour services.

Service peer, SDG agent, controller, and endpoints

The Cisco DNA Service for Bonjour solution is an end-to-end solution that includes the following key components and system roles to enable unicast-based service routing across the local area and Wide Area Bonjour domain:

- **Cisco Service Peer**: Cisco Catalyst Switches and Cisco Wireless Controllers in Layer 2 access function in service peer mode to support unicast-based communication with local attached endpoints and export service information to the upstream Cisco Catalyst SDG agent in the distribution layer.

> **Note**
>
> Cisco Nexus 9300 Series Switches don't support unicast-based service routing with downstream Layer 2 access network devices.

- **Cisco SDG Agent**: Cisco Catalyst and Cisco Nexus 9300 Series Switches function as an SDG agent and communicate with the Bonjour service endpoints in Layer 3 access mode. At the distribution layer, the SDG agent aggregates information from the downstream Cisco service peer switch and wireless controller, or local Layer 2 networks, and exports information to the central Catalyst Center controller.

> **Note**
>
> Cisco Nexus 9300 Series Switches don't support multilayer LAN-unicast deployment mode.

- Catalyst Center controller: The Catalyst Center controller builds the Wide Area Bonjour domain with network-wide and distributed trusted SDG agents using a secure communication channel for centralized services management and controlled service routing.
- **Endpoints**: A Bonjour endpoint is any device that advertises or queries Bonjour services conforming to RFC 6762. The Bonjour endpoints can be in either LANs or WLANs. The Cisco Wide Area Bonjour application is designed to integrate with RFC 6762-compliant Bonjour services, including AirPlay, Google Chrome cast, AirPrint, and so on.

Supported platforms

Lists supported platforms for Cisco DNA Service for Bonjour solution.

The following tables list the supported controllers, hardware, software versions, and SDG agents for the Cisco DNA Service for Bonjour solution.

Supported controllers, hardware, and software versions

Table 199: Supported Controllers with Supported Hardware and Software Versions

Supported Controller	Hardware	Software Version
Catalyst Center appliance	DN2-HW-APL DN2-HW-APL-L DN2-HW-APL-XL	Catalyst Center, Release 2.3.7.6
Catalyst Center on ESXi	–	Catalyst Center, Release 2.3.7.6
Cisco Wide Area Bonjour application on Catalyst Center	–	2.4.718.75196
Cisco Wide Area Bonjour application on Catalyst Center on ESXi	–	2.718.77018

Supported SDG agents, licenses, and software requirements

Table 200: Supported SDG Agents with Supported License and Software Requirements

Supported Platform	Supported Role	Local Area SDG	Wide Area SDG	Minimum Software
Cisco Catalyst 9200 Series Switches	SDG agent Service peer	Cisco DNA Advantage	Cisco DNA Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9200L Series Switches	SDG agent Service peer	Cisco DNA Advantage	Cisco DNA Advantage	Cisco IOS XE Release 17.11.1

Supported Platform	Supported Role	Local Area SDG	Wide Area SDG	Minimum Software
Cisco Catalyst 9300 and 9300-X Series Switches	Service peer SDG agent	Cisco DNA Advantage	Cisco DNA Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9400 and 9400-X Series Switches	Service peer SDG agent	Cisco DNA Advantage	Cisco DNA Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9500 and 9500-X Series Switches	Service peer SDG agent	Cisco DNA Advantage	Cisco DNA Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9500 High Performance Series Switches	Service peer SDG agent	Cisco DNA Advantage	Cisco DNA Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9600 and 9600-X Series Switches	Service peer SDG agent	Cisco DNA Advantage	Cisco DNA Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9800 Wireless Controller	Service peer	Cisco DNA Advantage	Cisco DNA Advantage	Cisco IOS XE Release 17.11.1
Cisco Catalyst 9800-L Wireless Controller	Service peer	Cisco DNA Advantage	Cisco DNA Advantage	Cisco IOS XE Release 17.11.1
Cisco Nexus 9300 Series Switches	SDG agent	Cisco DNA Advantage	Cisco DNA Advantage	Cisco NX-OS Release 10.2(3)F

Supported network designs

Explains the types of enterprise-grade networks that support the Cisco Catalyst Center Service for Bonjour, including traditional, Cisco SD-Access, and BGP EVPN-enabled wired and wireless networks.

TheCisco DNA Service for Bonjouris supported on traditional, Cisco SD-Access, and BGP EVPN-enabled wired and wireless networks.

The end-to-end unicast-based Bonjour service routing is supported on traditional, Cisco SD-Access, and BGP EVPN-enabled wired and wireless networks.

Traditional wired and wireless networks

Explains traditional networks as classic Layer 2 or Layer 3 networks for wired and wireless modes deployed in enterprise environments.

A traditional wired and wireless network is a network architecture that

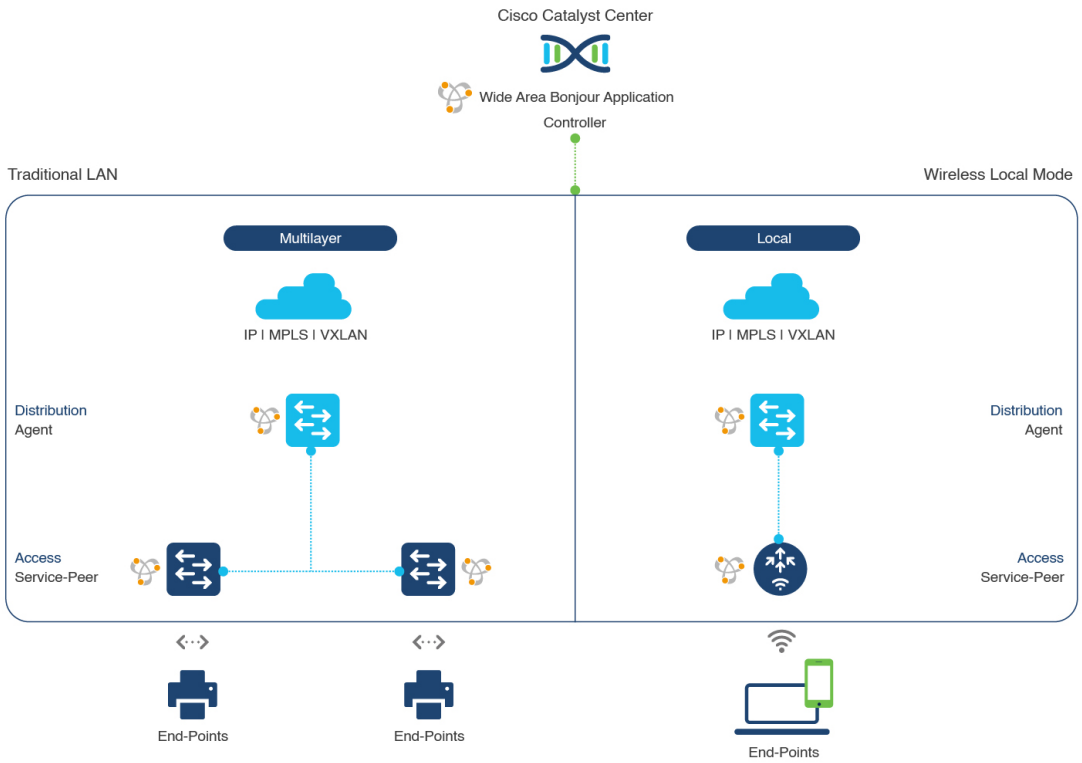
- uses classic Layer 2 or Layer 3 designs for wired and wireless modes, and
- is deployed in enterprise networks.

Network design features

Cisco DNA Service for Bonjour supports a broad range of network designs to enable end-to-end service routing and replace flood-n-learn-based deployment with a unicast mode-based solution.

Traditional LAN and central-switching wireless local mode network designs are commonly deployed in enterprise environments.

Figure 45: Enterprise traditional LAN and wireless local mode network design



Wired networks

Explains supported traditional LAN network designs that are commonly deployed in an enterprise.

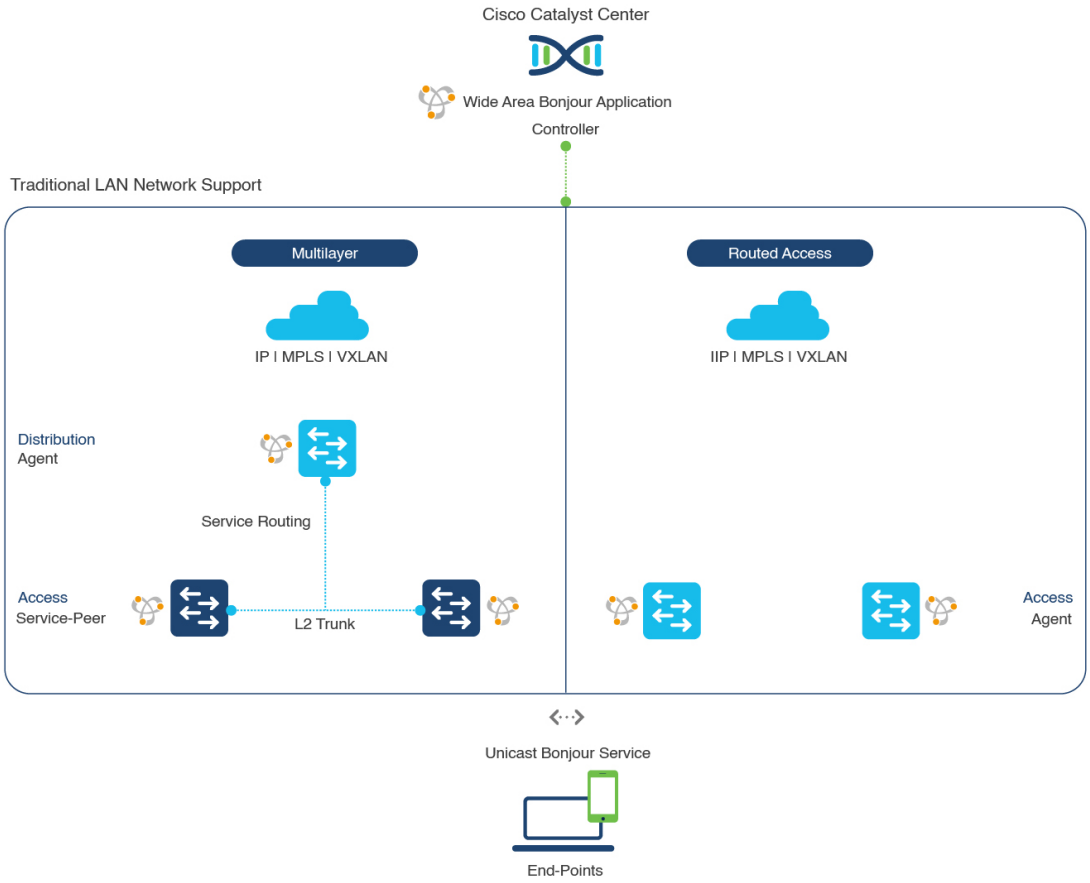
A wired network is a LAN network design that

- supports traditional enterprise deployments
- uses Cisco Catalyst or Cisco Nexus 9300 Series Switches in SDG agent role to provide Bonjour gateway functions, and
- enables IP gateway functionality for wired endpoints in multilayer or Layer 3 routed access network designs.

Enterprise wired multilayer and routed access network designs

The following figure shows the supported traditional LAN network designs that are commonly deployed in an enterprise.

Figure 46: Enterprise wired multilayer and routed access network design



The Cisco Catalyst or Cisco Nexus 9300 Series Switches in SDG agent role that provide Bonjour gateway functions are typically IP gateways for wired endpoints that could reside in the distribution layer in multilayer network designs, or in the access layer in Layer 3 routed access network designs:

- **Multilayer LAN–Unicast mode**: In this deployment mode, the Layer 2 access switch provides the first-hop mDNS gateway function to locally attached wired endpoints. In unicast mode, the mDNS services are routed to the distribution layer systems providing IP gateway and SDG agent mode. The policy-based service routing between the SDG agents is performed by the Catalyst Center controller.
- **Multilayer LAN–Flood-n-Learn mode**: In this deployment mode, the Layer 2 access switch or wireless controller are in mDNS passthrough modes with the Cisco Catalyst or Cisco Nexus 9300 Series Switches operating in the SDG agent mode. The mDNS gateway function at distribution layer in a network enables inter-VLAN mDNS local proxy. It also builds stateful Wide Area Bonjour unicast service routing with the Catalyst Center to discover or distribute mDNS services beyond a single IP gateway.
- **Routed access**: In this deployment mode, the first-hop Cisco Catalyst or Cisco Nexus 9300 Series Switch is an IP gateway boundary and, therefore, it must also perform the SDG agent role. The policy-based service routing between the SDG agents is performed by the Catalyst Center controller.

Wireless networks

Explains modes for wireless LAN networks supported by Cisco Catalyst Center Service for Bonjour.

A wireless network is a network architecture that

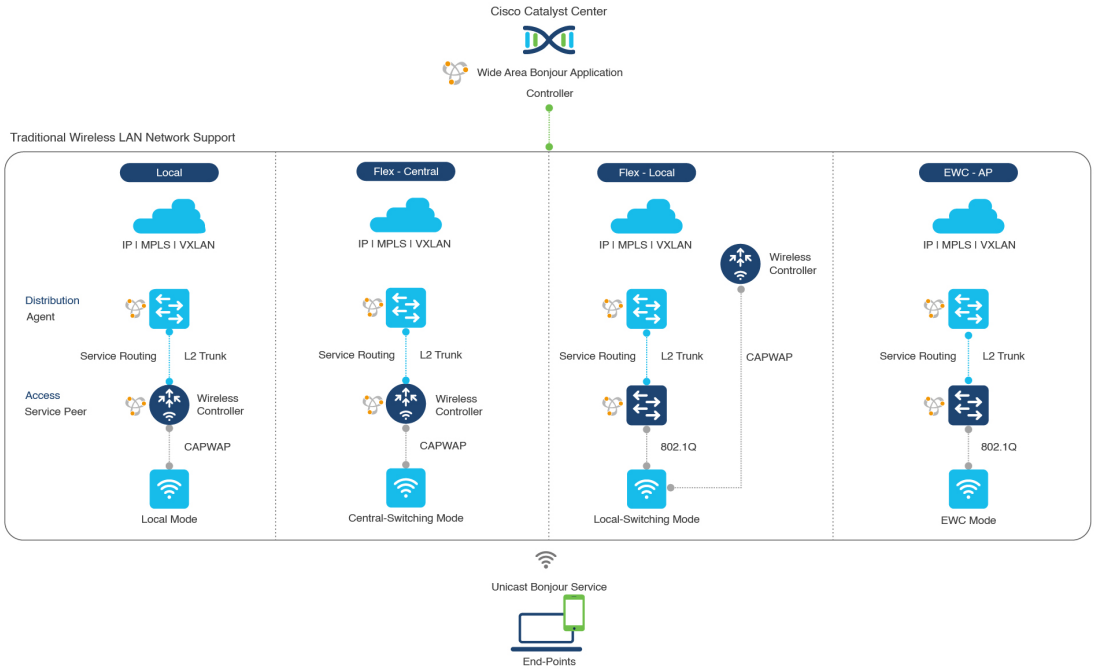
- enables wireless LAN connectivity for enterprise environments
- supports mDNS gateway functions for service discovery, and
- offers multiple deployment modes for flexible integration with Cisco Catalyst Center Service for Bonjour.

Supported wireless LAN network designs

The Cisco DNA Service for Bonjour extends the single wireless controller mDNS gateway function into the Wide Area Bonjour solution. The mDNS gateway on Cisco Catalyst 9800 Series Wireless Controller can be deployed in an enhanced mode as a service peer. In this mode, the wireless controller builds unicast service routing with an upstream Cisco Catalyst gateway switch for end-to-end mDNS service discovery. It replaces the classic flood-n-learn mDNS services from wired network using mDNS AP or other methods.

The following figure shows the supported traditional wireless LAN network designs that are commonly deployed in an enterprise. Based on the wireless network design, the mDNS gateway function may be on the wireless controller, or first-hop Layer 2 or Layer 3 Ethernet switch of an Access Point in local-switching mode.

Figure 47: Enterprise traditional wireless LAN network design



The Cisco DNA Service for Bonjour supports the following modes for wireless LAN networks:

- **Local mode**: In the central switching wireless deployment mode, the m-DNS traffic from local mode Cisco access points is terminated on the Cisco Catalyst 9800 Series Wireless Controller. The Cisco Catalyst 9800 Series Wireless Controller extends the mDNS gateway function to the new service peer mode. The wireless controller can discover and distribute services to local wireless users and perform unicast service routing over a wireless management interface to the upstream Cisco Catalyst Switch in the distribution layer, which acts as the IP gateway and the SDG agent.
- **FlexConnect–central**: The mDNS gateway function for Cisco access point in FlexConnect central switch SSID functions consistently as described in **Local mode**. The new extended mDNS gateway mode on the Cisco Wireless Controller and upstream service routing with SDG agent operate consistently to discover services across network based on policies and locations.
- **FlexConnect–local**: In FlexConnect local switching mode, the Layer 2 access switch in mDNS gateway service peer mode provides the policy-based mDNS gateway function to locally attached wired and wireless users. The Cisco Catalyst Switches in the distribution layer function as SDG agents and enable mDNS service-routing across all Layer 2 ethernet switches to support unicast-based service routing to LAN and wireless LAN user groups.
- **Embedded Wireless Controller–access point**: The Layer 2 access switch in service peer mode provides unified mDNS gateway function to wired and wireless endpoints associated with Cisco Embedded Wireless Controller on Cisco Catalyst 9100 Series Access Points. The SDG agent in the distribution layer provides unicast service routing across all Layer 2 service peer switches in the Layer 2 network block without any mDNS flooding.

Cisco SD-Access wired and wireless networks

Explains how Cisco SD-Access-enabled wired and wireless networks support Cisco Catalyst Center Service for Bonjour across fabric networks, providing secure and segmented mDNS service discovery and distribution management.

A Cisco SD-Access wired and wireless network is a network infrastructure solution that

- supports Cisco DNA Service for Bonjour across fabric networks
- provides VRF-aware Wide Area Bonjour service routing for secure and segmented mDNS service discovery and distribution management, and
- eliminates the need to extend Layer 2 flooding, improving the scale and performance of the fabric core network and endpoints.

Network design alternatives for Cisco SD-Access wired and wireless networks

Cisco SD-Access supports flexible wired and wireless network design alternatives to manage fully distributed, integrated, and backward-compatible traditional network infrastructure. Wide Area Bonjour service routing is supported in all network designs providing intuitive user experience.

Figure 48: Cisco SD-Access Wired and wireless network design

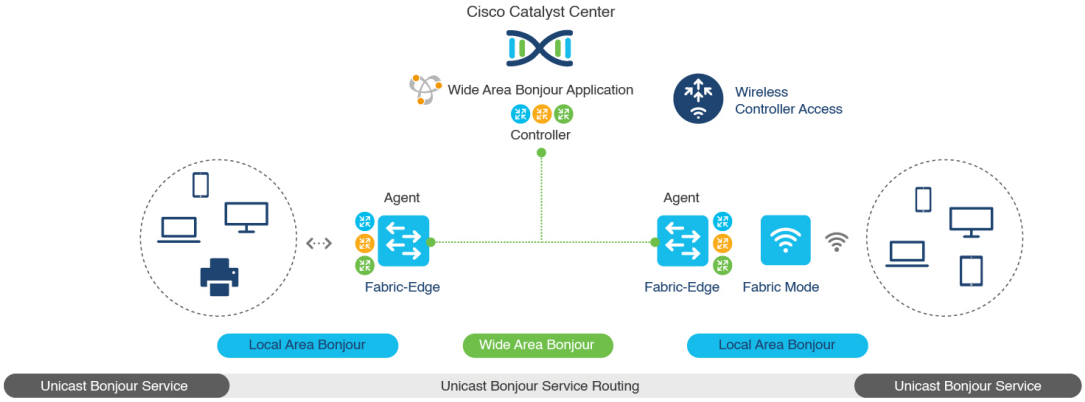
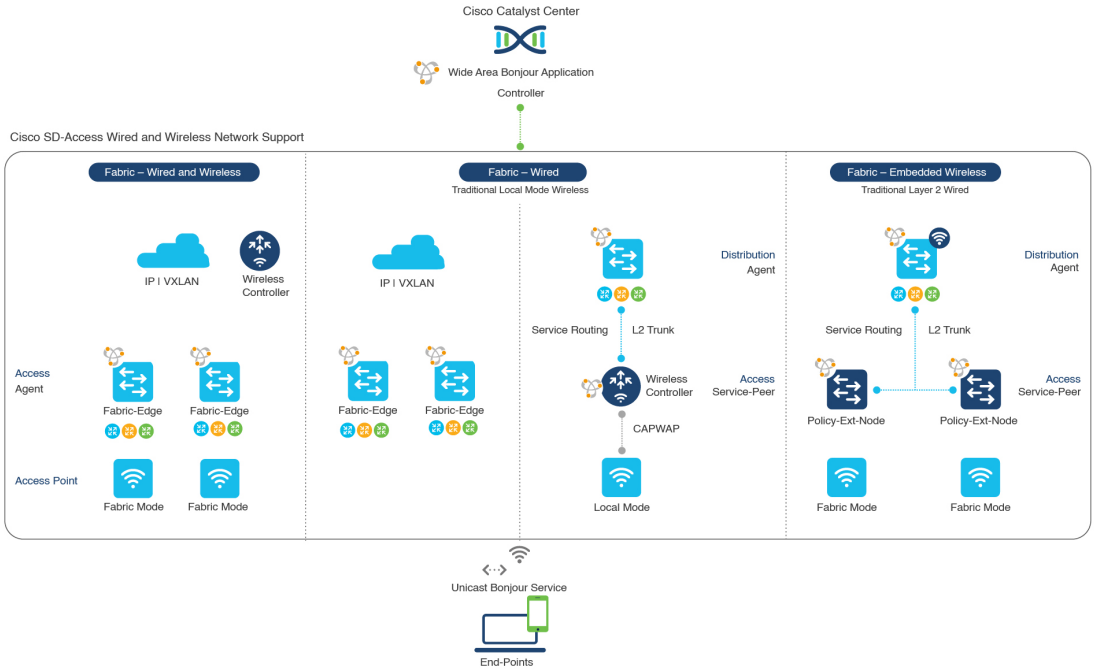


Figure 49: Cisco SD-Access Wired and wireless network design alternatives



The Cisco DNA Service for Bonjour for SD-Access enabled wired and fabric, or traditional mode-wireless networks use two-tier service routing providing end-to-end unicast-based mDNS solution. Based on the network design, each solution component is enabled in a unique role to support the Wide Area Bonjour domain:

- **Fabric Edge SDG Agent**: The Layer 3 Cisco Catalyst Fabric Edge switch in the access layer configured as SDG agent provides unicast-based mDNS gateway function to the locally attached wired and wireless endpoints. The VRF-aware mDNS service policy provides network service security and segmentation in a virtual network environment. The mDNS services can be locally distributed and routed through centralized Catalyst Center.
- **Policy Extended Node**: The Layer 2 Cisco Catalyst access layer switch enables first-hop mDNS gateway function without flooding across the Layer 2 broadcast domain. The unicast-based service routing with upstream Fabric Edge switch in the distribution layer enables mDNS service routing within the same Layer 2 network block. It can also perform remote service discovery and distribution from centralized Catalyst Center.
- Cisco Wireless Controller: Based on the following wireless deployment modes, Cisco Wireless Controller supports unique function to enable mDNS service routing in Cisco SD-Access enabled network:
  - **Fabric-Enabled Wireless**: Cisco Wireless Controller doesn't require any mDNS gateway capability to be enabled in distributed fabric-enabled wireless deployments.
  - **Local Mode Wireless**: As Cisco Wireless Controller provides central control and data plane termination, it provides mDNS gateway in service peer mode for wireless endpoints. The wireless controller provides mDNS gateway between locally associated wireless clients. The wireless controller builds service routing with upstream SDG agent Catalyst switch providing IP gateway and service routing function for wireless endpoints.
  - **Embedded Wireless Controller–Switch**: The Cisco Embedded Wireless Controller solution enables the lightweight integrated wireless controller function within the Cisco Catalyst 9300 Series Switch. The Cisco Catalyst switches in the distribution layer function as SDG agents to the wired and wireless endpoints. The SDG agent in the distribution layer provides unicast service routing across all wireless access points and Layer 2 service peer switches without mDNS flooding.
- Catalyst Center controller: The Cisco Wide Area Bonjour application on Catalyst Center supports policy and location-based service discovery, and distribution between network-wide distributed Fabric Edge switches in SDG agent mode.

The Wide Area Bonjour communication between the SDG agent and controller takes place through the network underlay. Based on policies, the SDG agent forwards the endpoint announcements or queries to the Catalyst Center. After discovering

a service, the endpoints can establish direct unicast communication through the fabric overlay in the same virtual network. The inter-virtual network unicast communication takes place through the Fusion router or external Firewall system. This communication is subject to the configured overlay IP routing and Security Group Tag (SGT) policies.

SD-Access enabled network example

In a Cisco SD-Access enabled network, the Fabric Edge SDG Agent provides unicast-based mDNS gateway function to locally attached wired and wireless endpoints, while the Policy Extended Node enables first-hop mDNS gateway function without flooding across the Layer 2 broadcast domain.

Non-SD-Access network example

Traditional networks without SD-Access do not provide VRF-aware mDNS service routing or eliminate Layer 2 flooding, resulting in less secure and less scalable service discovery.

SD-Access network analogy

A Cisco SD-Access wired and wireless network is like a gated community where each resident (endpoint) can securely discover and communicate with services without unnecessary traffic flooding the neighborhood.

BGP EVPN networks

Explains how BGP EVPN-based technology provides flexible Layer 3 segmentation and Layer 2 extension overlay networks, supporting secure and segmented mDNS service routing in enterprise environments.

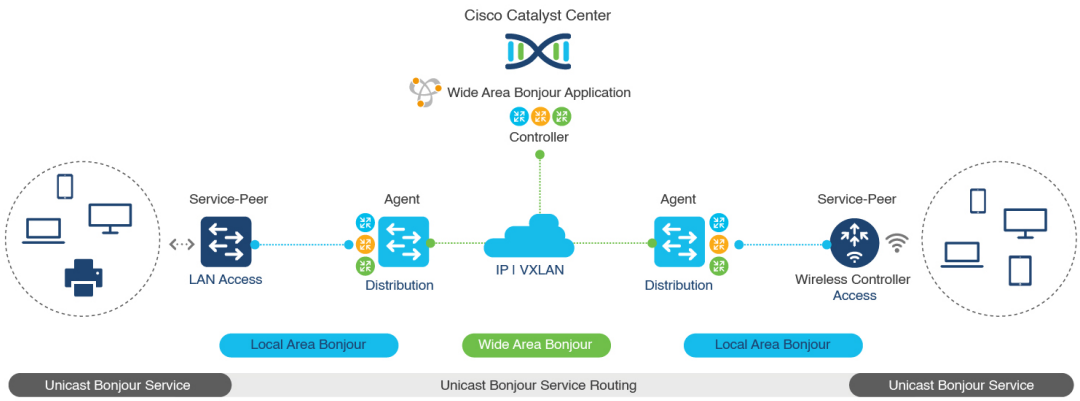
A BGP EVPN network is an overlay network technology that

- provides flexible Layer 3 segmentation and Layer 2 extension overlays
- enables secure and segmented mDNS service routing using VRF and EVPN VXLAN-aware Wide Area Bonjour service routing, and
- eliminates mDNS flooding and solves service reachability challenges for Layer 3 segmented routed networks in the fabric.

Overlay service routing in BGP EVPN networks

The BGP EVPN leaf switch in the distribution layer supports overlay Wide Area Bonjour service routing for BGP EVPN-enabled traditional Layer 2 wired access switches and traditional wireless local mode enterprise networks interconnected through various types of Layer 2 networks and Layer 3 segmented VRF-enabled networks.

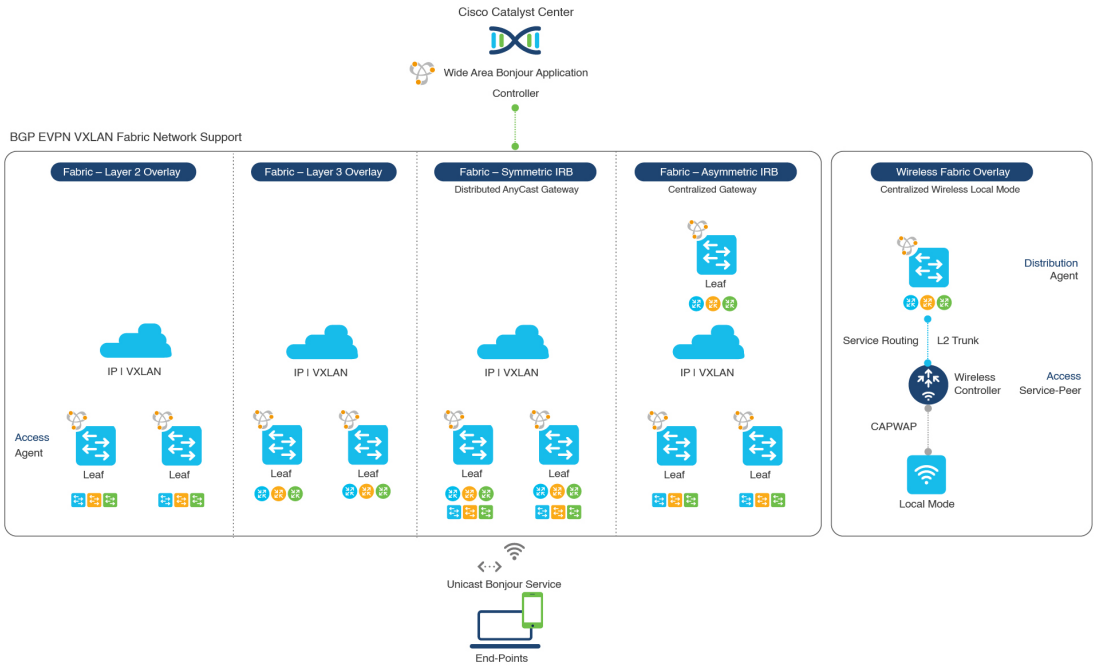
Figure 50: Overlay Bonjour service for a BGP EVPN-enabled enterprise network



Cisco DNA Service for Bonjour supports all industry-standard overlay network designs, enabling end-to-end unicast-based mDNS service routing and preventing flooding and service boundary limitations across wired and wireless networks.

The following figure illustrates various BGP EVPN VXLAN reference overlay network design alternatives. This network design enables end-to-end mDNS service discovery and distribution based on overlay network policies.

Figure 51: BGP EVPN VXLAN wired and wireless design alternatives



The Cisco Catalyst and Cisco Nexus 9000 Series Switches can be deployed in Layer 2 or Layer 3 leaf roles supporting mDNS service routing for a broad range of overlay networks. In any role, mDNS communication is limited locally and supports end-to-end unicast-based service routing across Wide Area Bonjour domain:

- **Layer 2 leaf SDG agent**: The Cisco Catalyst or Cisco Nexus switches can be deployed as Layer 2 leaf supporting end-to-end bridged network with IP gateway within or beyond BGP EVPN VXLAN fabric network. By default, mDNS is flooded as Broadcast, Unknown Unicast, Multicast (BUM) over the fabric-enabled core network. This mDNS flooding may impact network performance and security. The Layer 2 leaf, enabled as SDG agent, prevents mDNS flooding over VXLAN and supports unicast-based service routing.
- **Layer 3 leaf SDG agent**: The Cisco Catalyst or Cisco Nexus switches can be deployed as SDG agent supporting Layer 3 overlay network in BGP EVPN VXLAN fabric. The IP gateway and mDNS service boundary is terminated at the SDG agent switches and remote services can be discovered or distributed through centralized Catalyst Center.
- **Local mode wireless**: The centralized wireless local mode network can be terminated within or outside the EVPN VXLAN fabric domain to retain network segmentation and service discovery for wireless endpoints. The Cisco Catalyst 9800 Series Wireless Controller in service peer mode can build unicast service routing with distribution layer IP and SDG agent Cisco Catalyst switch to discover services from BGP EVPN VXLAN fabric overlay network.
- Catalyst Center: Catalyst Center supports Wide Area Bonjour capability to dynamically discover and distribute mDNS services based on Layer 2 or Layer 3 Virtual Network ID (VNID) policies to route the mDNS services between SDG agent switches in the network.

For more information about BGP EVPN networks, see *Cisco DNA Service for Bonjour Configuration Guide, Cisco IOS XE Bengaluru 17.6.x (Catalyst 9600 Switches)*.

BGP EVPN overlay network example

The overlay Bonjour service for a BGP EVPN-enabled enterprise network demonstrates how mDNS service routing is supported for both wired and wireless access switches interconnected through Layer 2 and Layer 3 segmented VRF-enabled networks.

Non-EVPN overlay network limitations

Traditional Layer 2 networks without BGP EVPN overlays may experience mDNS flooding and lack segmentation, resulting in service reachability challenges.

Overlay network analogy

A BGP EVPN overlay network is like a highway system that enables secure and segmented traffic flow, preventing congestion and ensuring efficient service delivery across different routes.

230 Configuring Local and Wide Area Bonjour Domains

Topics:

- Cisco DNA Service for Bonjour Solution Overview
- Configure local and wide area Bonjour domains
- Verify local area bonjour in multicast DNS mode for LAN and wireless networks
- Additional references for DNA Service for Bonjour

Guides the setup and configuration of Bonjour service discovery across local and wide area networks, including service routing and redundancy protocols.

Cisco DNA Service for Bonjour Solution Overview

Describes the Cisco DNA Service for Bonjour solution that enables Apple devices to discover and connect to network services across subnets in enterprise environments. Explains how the service facilitates seamless device connectivity while maintaining network security and performance.

Restrictions for Cisco DNA service for Bonjour solution

Consider these restrictions when implementing Cisco DNA Service for Bonjour Solution.

Review platform compatibility requirements before deployment. Cisco Service Discovery Gateway (SDG) and Wide Area Bonjour gateway function is supported on Cisco Catalyst Switch and Cisco ISR 4000 series routers. See Solution components on page 3053 for the complete list of supporting platforms, software versions and license levels.

- Use the new **mdns-sd gateway** method for Bonjour configuration policies. Cisco IOS supports classic and new method of building local Bonjour configuration policies. The classic method is based on **service-list mdns-sd** CLI whereas the new method is based on **mdns-sd gateway**. We recommend using the new **mdns-sd gateway** method since the classic configuration support will be deprecated in near future releases.
- Plan for manual migration when converting from classic to new method CLI. The classic to new method CLI migration is manual procedure to convert the configuration.
- Understand that Bonjour service policies on Cisco SDG Gateways are effective between local VLANs. In addition to these, a specific egress policy controls the type of services to be exported to the controller. The Layer 2 Multicast-DNS Bonjour communication between two end-points on same broadcast domain is transparent to gateway.
- Disable mDNS Snooping on Cisco WLC controller for wireless networks. To enable end-to-end Wide Area Bonjour solution on Wireless networks, the Cisco WLC controller must not enable mDNS Snooping function. The upstream IP gateway on the dedicated Cisco Catalyst switch must have the Bonjour gateway function enabled for wireless clients.
- Enable AP Multicast with unique Multicast group on Cisco Wireless LAN Controller. Cisco Wireless LAN Controller must enable AP Multicast with unique Multicast group. Without AP joining WLC Multicast group the mDNS messages will not be processed between client and gateway switch. Multicast on Client SSID or VLAN is optional for other multicast applications and not mandatory or required for Bonjour solution.
- Note the limitations of Cisco Catalyst 9800 WLC as mDNS Gateway. Cisco Catalyst 9800 WLC can be configured as mDNS Gateway. In this mode, the Cisco Catalyst 9800 WLC supports Local-Area Bonjour gateway solution limited to Wireless only networks. Cisco Catalyst 9800 does not support Wide Area Bonjour. For end-to-end Wired and Wireless Bonjour support, we recommend using upstream Cisco Catalyst Switch as IP and Bonjour gateway.

How Cisco Wide Area Bonjour services work

Describes how the Cisco wide area Bonjour solution follows a client-server model for service announcement and discovery in IP networks.

The Cisco Wide Area Bonjour solution follows a client-server model. The SDG Agent functions as a client and the Cisco Wide Area Bonjour application Cisco Catalyst Center functions as a server.

Summary

The key components involved in the Cisco Wide Area Bonjour service workflow are:

- Endpoint devices (Source): Send service announcements to the SDG Agent specifying what services they offer
- SDG Agent: Functions as a client, listens to announcements and matches them against configured policies
- Endpoint device (Receiver): Sends Bonjour queries to discover available services using mDNS protocol
- Cisco Wide Area Bonjour application: Functions as a server for service routing and distribution

Workflow

These stages describe the workflow of service announcement and discovery in the IP network:

1. Endpoint devices announce services to the network through the SDG Agent.
   - The endpoint devices (Source) in the Local Area Bonjour domain send service announcements to the SDG Agent and specify what services they offer. For example, _airplay._tcp.local, _raop._tcp.local, _ipp._tcp.local, and so on.
   - The SDG Agent listens to these announcements and matches them against the configured Local Area SDG Agent policies. If the announcement matches the configured policies, the SDG Agent accepts the service announcement and routes the service to the controller.
2. Endpoint devices discover services available in the network through the SDG Agent.
   - The endpoint device (Receiver) connected to the Local Area SDG Agent sends a Bonjour query to discover the services available, using the mDNS protocol.
   - If the query conforms to configured policies, SDG Agent responds with the services obtained from appropriate service routing via the Wide Area Bonjour Controller.
3. Wide Area Bonjour multi-tier policies control the service announcements and queries.

   The various policies that can be used to control the Bonjour announcements and queries are classified as these:
   - **Local Area SDG Agent Filters**: Enforced on the SDG Agent in Layer-2 Network Domain. These bi-directional policies control the Bonjour announcements or queries between the SDG Agents and the Bonjour endpoints.
   - **Wide Area SDG Agent Filters**: Enforced on the SDG Agent for export control to the Controller. This egress unidirectional policy controls the service routing from the SDG Agent to the controller.
   - **Cisco Wide Area Bonjour Policy**: Enforced on Controller for global service discovery and distribution. Policy enforcement, between the controller and the IP network is bi-directional.

Cisco wide area Bonjour supported network design

Describes the network design considerations and architectures that support Cisco wide area Bonjour implementations across distributed enterprise environments.

Traditional wired and wireless networks

Explains the various LAN network designs that Cisco DNA Service for Bonjour supports in enterprise deployments.

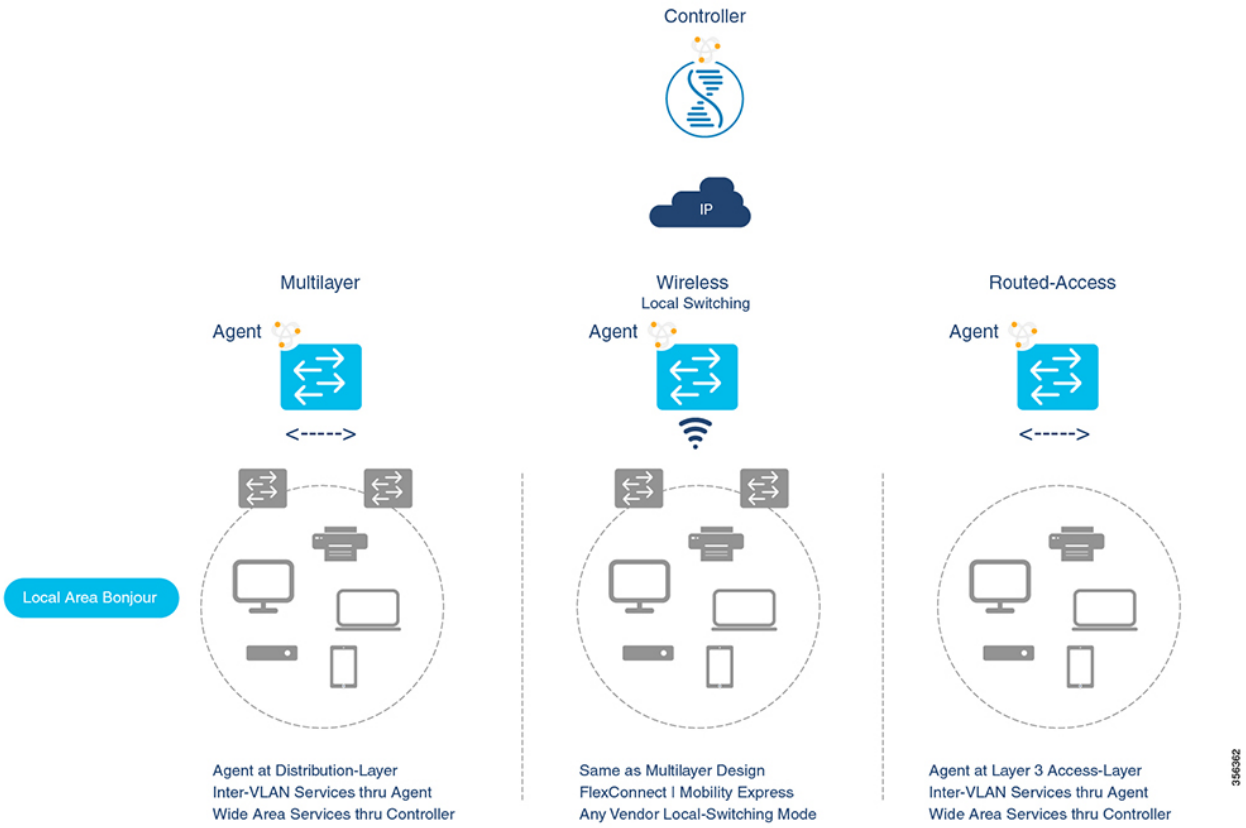
Traditional wired and wireless networks are enterprise network architectures that

- support various LAN network designs commonly deployed in the enterprise with Cisco DNA Service for Bonjour
- utilize SDG Agents that provide Bonjour gateway functions as IP gateways for endpoints, and
- enable consistent Bonjour gateway functions across both wired and wireless network deployments.

Network deployment modes

The SDG Agent providing Bonjour gateway functions is typically an IP gateway for wired end-points that could be residing in the distribution layer in multilayer network designs, or in the access layer in routed access network designs.

The following figure shows various topologies which are explained further in the section.



Cisco DNA Service for Bonjour supports these network deployment modes:

- **Multilayer LAN**: In this deployment mode, the Layer 2 Access switch provides the transparent bridging function of Bonjour services to Distribution-layer systems that act as the IP gateway and SDG Agent. There is no additional configuration or new requirement to modify the existing Layer-2 trunk settings between the Access and Distribution Layer Cisco Catalyst Switches.
- **Routed Access**: In this deployment mode, the first-hop switch is an IP gateway boundary and therefore, it must be combined with the SDG Agent role.

The Cisco DNA Service for Bonjour also supports various Wireless LAN network designs commonly deployed in the Enterprise. The SDG Agent provides consistent Bonjour gateway functions for the wireless endpoints as in wired networks. In general, the IP gateway of the wireless clients is also a Bonjour gateway. However, the placement of the SDG Agent may vary depending on the Wireless LAN deployment mode.

Cisco SD Access wired and wireless networks

Describes how Wide Area Bonjour integrates with Cisco SD-Access fabric-enabled wired and wireless networks using SDG Agents and Controllers.

Cisco SD Access wired and wireless networks are fabric-enabled network architectures that

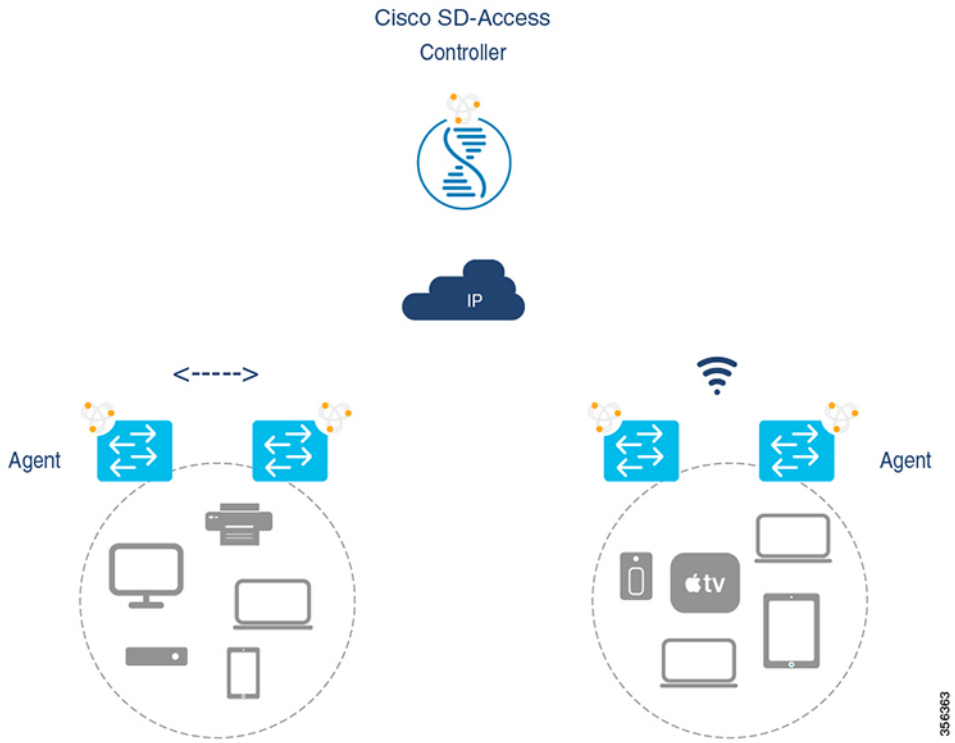
- configure the Fabric Edge switch as the SDG Agent for both wired and wireless endpoints
- require Wide Area Bonjour policies to align with SD-Access network policies regarding Virtual Networks and SGT policies, and
- enable service discovery communication through fabric underlay while maintaining data communication through fabric overlay.

Wide Area Bonjour logical components

Wide Area Bonjour uses two logical components in a network:

- SDG Agent: The Fabric Edge switch is configured as the SDG Agent, and the configuration is added only after the SD-Access is configured.
- Wide Area Bonjour Controller: The Wide Area Bonjour application in the Cisco Catalyst Center acts as the Controller.

The Wide Area Bonjour communication between the SDG Agent and the Controller takes place through the network underlay. The SDG Agent forwards the endpoint announcements or queries to the Controller through the fabric underlay. After discovering a service, a Bonjour-enabled application establishes direct unicast communication with the discovered device through the fabric overlay. This communication is subject to any configured routing and SDG policies.



Local and wide area Bonjour policies

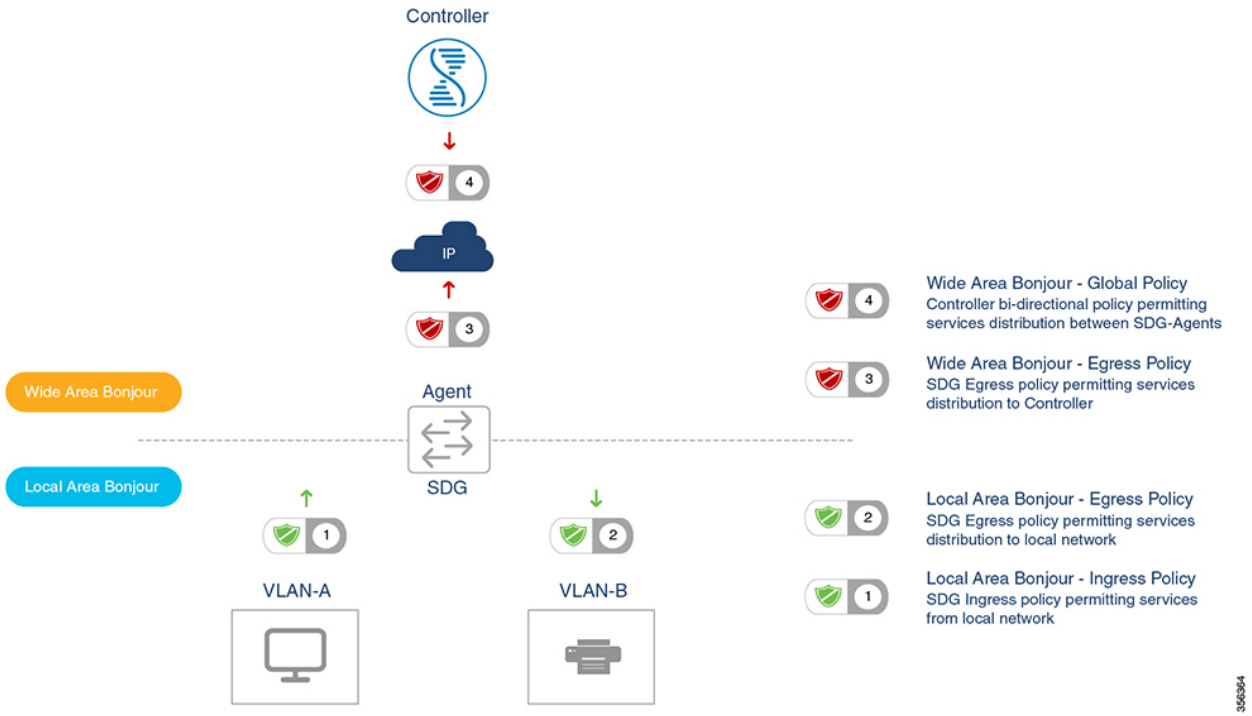
Describes Cisco Wide Area Bonjour policy functions that enable policy-based Bonjour services discovery and distribution in two-tier domains.

A Cisco Wide Area Bonjour policy is a network configuration mechanism that

- enables policy-based Bonjour services discovery and distribution in two-tier domains
- divides functionality into four unique functions for comprehensive service management, and
- allows network administrators to identify Bonjour services and set discovery boundaries limited to local or global based on requirements.

Policy enforcement points

The policy enforcement occurs at the SDG Agent level and in Cisco Catalyst Center Wide Area Bonjour application. This illustration shows enforcement point and direction of all four types of Bonjour policies:



Local area Bonjour policies

Explains the simplified and scalable Cisco IOS Bonjour policy structure with the new configuration mode.

A local area Bonjour policy is a network configuration framework that

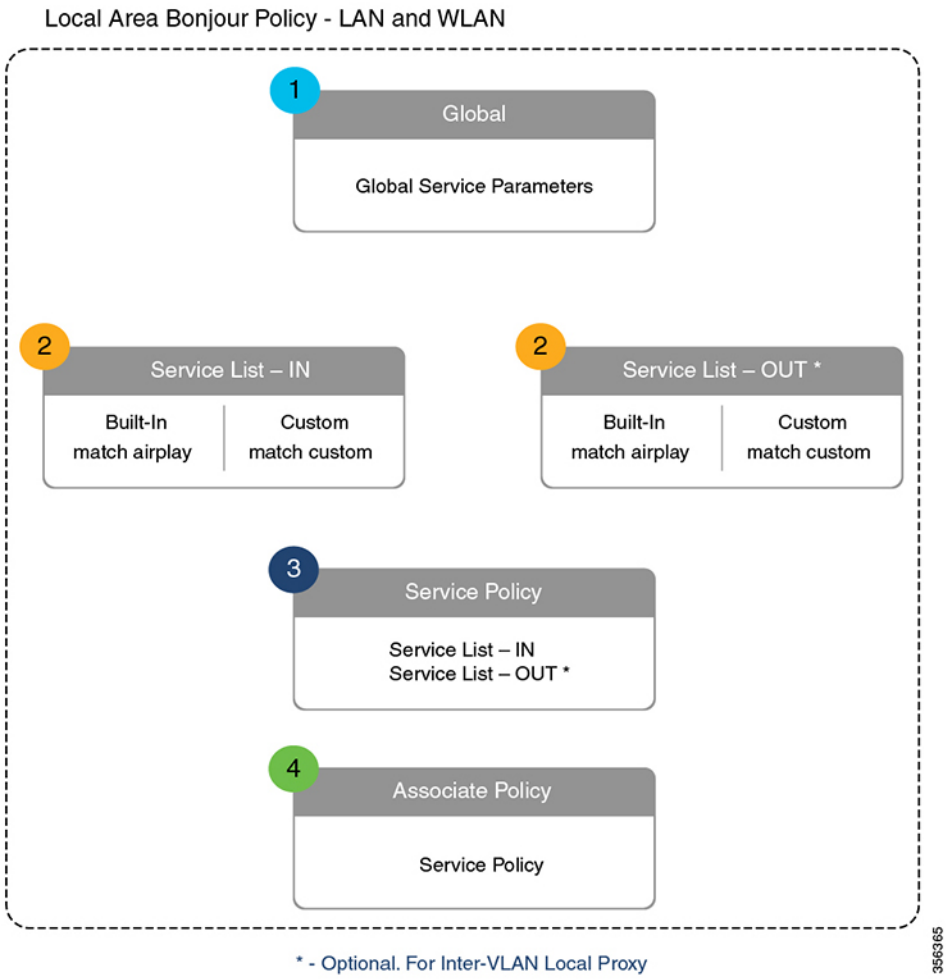
- enables service distribution with intuitive user-friendly service-types instead of individual mDNS PoinTeR (PTR) records types
- provides built-in service-types for common Enterprise services and allows custom service-type creation for network administrators, and
- requires a mandatory three-step configuration process for Local Area Bonjour domain management.

Local area Bonjour policy configuration

The Cisco IOS Bonjour policy structure is greatly simplified and scalable with the new configuration mode. The services can be enabled with intuitive user-friendly service-type instead individual mDNS PoinTeR (PTR) records types, for example select AirPlay that automatically enables video and audio service support from Apple TV or equivalent capable devices. Several common types of services in Enterprise can be enabled with built-in service-types. If built-in service type is limited, network administrator can create custom service-type and enable the service distribution in the network.

The policy configuration for the Local Area Bonjour domain is mandatory, and is a three step process. This illustration shows the step-by-step procedure to build the Local-Area Bonjour policy, and apply to enable the gateway function on selected local networks:

Figure 52: Local Area Bonjour Policy Hierarchy



To configure local area bonjour policies, enable mDNS globally. For the device to receive mDNS packets on the interface, configure mDNS gateway on the interface. Create a service-list by using filter options within IT allow services into or OUT of a device or interface. After enabling mDNS gateway globally and on the interface, you can apply filters (IN-bound filtering or OUT-bound filtering) on service discovery information by using **service-policy** commands.

Built-in service lists

The Cisco IOS software includes built-in list of services that may consist of one more Bonjour service-type. A single service-list may contain more than one service-type entries with default rule to accept service announcement from service-provider and the service query request from receiver end-points. If selected service-type contains more than one Bonjour service-types (PTR), then a service announcement or a service query is honored when the announcement/query is for any one of these included Bonjour service-types. For example, Apple Time Capsule Data service-type consists of both_adisk and _afpovertcp built-in PTRs, however if any end-point announces or requests for only _afpovertcp service, then SDG Agent will successfully classify and process the announcement or request. The service-list contains implicit-deny for all un-defined built-in or custom services entries.

This table illustrates complete list of built-in Bonjour services that can be used to create policies in local area Bonjour.

Table 201: Cisco IOS Built-In Bonjour Service Database

Service	Service Name	mDNS PTRs
Airplay	airplay	_airplay._tcp.local

Service	Service Name	mDNS PTRs
Apple TV	apple-tv	_airplay._tcp.local _raop._tcp.local
Audinate	audinate	_dante-safe._udp.local _dante-upgr._udp.local _netaudio-arc._udp.local _netaudio-chan._udp.local _netaudio-cmc._udp.local _netaudio-dbc._udp.local
AirServer Mirroring Service	airserver	_airplay._tcp.local _airserver._tcp.local
Apple AirTunes	airtunes	_raop._tcp.local
Amazon Fire TV	amazon-fire-tv	_amzn-wplay._tcp.local
Apple AirPrint	apple-airprint	_ipp._tcp.local _universal._sub._ipp._tcp.local
Apple TV 2	apple-continuity	_companion-link._tcp.local
Apple File Share	apple-file-share	_afpovertcp._tcp.local
Apple HomeKit	apple-homekit	_homekit._ipp.local _hap._tcp.local
Apple iTunes Library	apple-itunes-library	_atc._tcp.local
Apple iTunes Music	apple-itunes-music	_daap._tcp.local
Apple iTunes Photo	apple-itunes-photo	_dpap._tcp.local
Apple KeyNote Remote Control	apple-keynote	_keynotecontrol._tcp.local _keynotepair._tcp.local
Apple Remote Desktop	apple-rdp	_afpovertcp._tcp.local _net-assistant._tcp.local
Apple Remote Event	apple-remote-events	_eppc._tcp.local
Apple Remote Login	apple-remote-login	_sftp-ssh._tcp.local _ssh._tcp.local
Apple Screen Share	apple-screen-share	_rfb._tcp.local
Google Expeditions	google-expeditions	_googexpeditions._tcp.local
Apple Time Capsule Data	apple-timecapsule	_adisk._tcp.local _afpovertcp._tcp.local

Service	Service Name	mDNS PTRs
Apple Time Capsule Management	apple-timecapsule-mgmt	_airport._tcp.local
Apple MS Window File Share	apple-windows-fileshare	_smb._tcp.local
Fax	fax	_fax-ipp._tcp.local
Google ChromeCast	google-chromecast	_googlecast._tcp.local _googlerpc._tcp.local _googlezone._tcp.local
Apple HomeSharing	homesharing	_home-sharing._tcp.local
Apple iTunes Data Sync	itune-wireless-devicesharing2	_apple-mobdev2._tcp.local
Multifunction Printer	multifunction-printer	_ipp._tcp.local _scanner._tcp.local _fax-ipp._tcp.local
Phillips Hue Lights	phillips-hue-lights	_hap._tcp.local
Printer – Internet Printing Protocol	printer-ipp	_ipp._tcp.local
Printer – IPP over SSL	printer-ipps	_ipps._tcp.local
Linux Printer – Line Printer Daemon	printer-lpd	_printer._tcp.local
Printer Socket	printer-socket	_pdl-datastream._tcp.local
Roku Media Player	roku	_rsp._tcp.local
Scanner	scanner	_scanner._tcp.local
Spotify Music Service	spotify	_spotify-connect._tcp.local
Web-Server	web-server	_http._tcp.local
WorkStation	workstation	_workstation._tcp.local

Service Status Timer Management

The Bonjour service-provider end-points may announces one or more services in the network combining mDNS records and time-to-live (TTL) service timers for each record. The TTL value provides assurance of end-point availability and serviceability in the network. The SDG Agents ensures that IT contains up to date information in its local and updates global services in Controller based on TTL and other events in Local Area Bonjour domain. The network administrator must configure the service status timer where service-provider endpoint discovery is permitted.

Wide Area Bonjour policies

Describes Wide Area Bonjour policies that enable service routing over IP networks without network boundaries.

A Wide Area Bonjour policy is a network service control mechanism that

- requires controller bound service export policy to control routing local services and discover remote services from Cisco Catalyst Center

- enables SDG-Agent to build trusted communication channels with remote service responses implicitly permitted, and
- operates unidirectionally requiring only egress service policy towards the controller.

Policy structure and configuration

The Wide Area Bonjour policy hierarchy and structure is identical as described in Local Area Bonjour Policy structure section. This section provides step-by-step reference configuration to build and enforce the policy to enable the successful communication with Wide Area Bonjour App in Cisco Catalyst Center.

Policy configuration components:

- **Service List**: Network administrators must create new controller bound egress service lists for the Wide Area Bonjour domain. In most common network deployment models, the Wide Area Bonjour service list may contain same service-types as the Local Area Bonjour to implement common services between both domains. Based on requirements, certain services can be limited to Local Area and prevent routed in Wide Area Domain, then by default only allowed service list entries are permitted and rest are dropped with implicit deny rule.
- **Ingress Policy Direction**: The ingress service policy for Wide Area Bonjour domain is not required and cannot be associated to the controller.
- **Egress Policy Direction**: The Bonjour policy structure between Local Area and Wide Area is consistent, however the enforcement point is different. Cisco recommends configuring separate Service-List and Service-Policy for Wide Area Bonjour domain as this helps building unique policy sets for each domain.
- **Conditional Egress Service List**: The Wide Area Bonjour egress service list configuration can be customized to conditionally route the service or query request to the Cisco Catalyst Center. With this alternative configuration setting, network administrators can route the service or query the request in Wide Area Bonjour domain from specific local source VLAN networks instead globally from the entire system.

Service timer management:

The Cisco Catalyst Center centralizes services information from large scale distributed SDG-Agents across the network. To maintain scale and performance of the controller, services routing information is transmitted and synchronized periodically by each SDG-Agent network device. To protect system and network performance, the scheduler base service information exchange allows graceful and reliable way to discover and distribute Bonjour services across Wide Area Bonjour domain.

> **Note**
>
> In most large-scale network environments, the default Bonjour service timers on SDG-Agents are fine-tuned and may not need any further adjustments. Cisco recommends retaining the interval timer values to default and adjust only based on any user experience issue and consider modified parameters do not introduce scale and performance impact.

Default mDNS service configurations

Explains the default mDNS service configurations that accelerate solution adoption, increase user productivity, and reduce operation overhead through a default service policy.

Default mDNS service configurations are an intuitive approach to configuring mDNS services that

- accelerates solution adoption, increases user productivity, and reduces operation overhead through a default service policy
- creates a service list with default service-types that is automatically enforced in the ingress or egress direction, and

- allows you to define a custom policy and service list with custom-defined service types.

Default mDNS service configurations details

Starting with Cisco IOS XE Bengaluru 17.6.1, the default mDNS service configuration is introduced. The default service configuration contains a default service policy that creates a service list with default service-types that is automatically enforced in the ingress or egress direction.

Additionally, you can define a custom policy and service list with custom-defined service types, and enforce it in the ingress or egress direction.

Figure 53: Default mDNS service configurations



HSRP-aware mDNS service-routing

Explains Hot Standby Router Protocol-aware (HSRP-aware) mDNS Service-Routing that is supported between Service Peers and SDG agents in a multilayer network.

HSRP-Aware mDNS Service-Routing is a network protocol feature that

- maintains uninterrupted service-routing sessions between Service Peers and SDG agents during primary SDG agent failures
- enables automatic establishment of sessions with new primary SDG agents and cache information resyncing during changeouts, and
- uses HSRP virtual IP addresses on management VLANs to provide automatic gateway selection and rapid switchover capabilities.

HSRP-Aware mDNS Service-Routing configuration

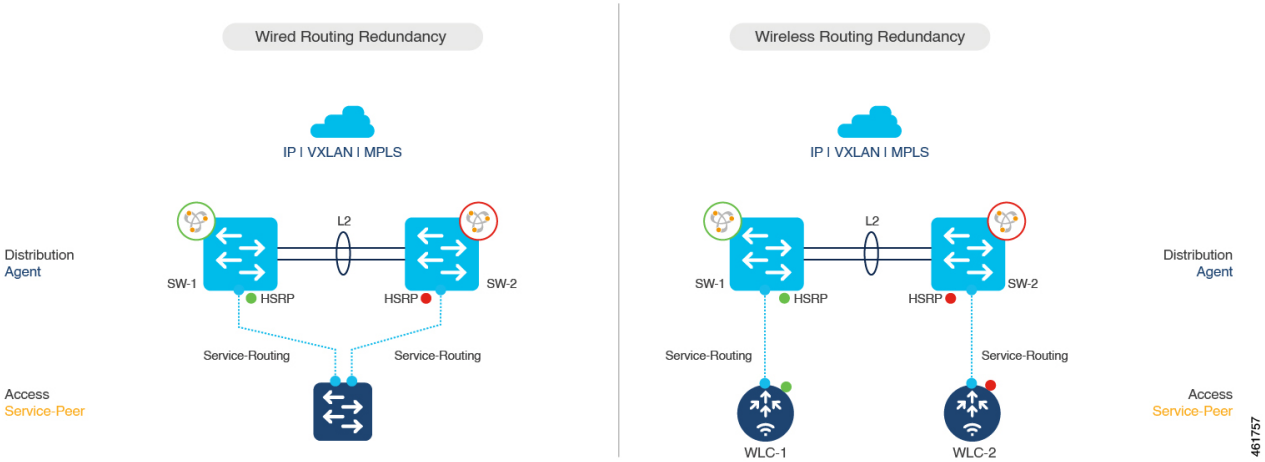
Starting from Cisco IOS XE Bengaluru 17.6.1, Hot Standby Router Protocol-aware (HSRP-aware) mDNS Service-Routing is supported between Service Peers and SDG agents in a multilayer network. During a changeover, that is when the primary SDG agent fails and the secondary SDG agent becomes the new primary, the service-routing session between the Service Peer and the SDG agent remains uninterrupted. The new primary SDG agent establishes a session with the Service Peer and cache information is resynced.

The HSRP virtual IP address of the management VLAN is enabled on the SDG agent using the **standby** *group_number* **IP** *ip_address* command. The HSRP virtual IP address needs to configured on the Service Peer as the IP address of the SDG agent.

> **Note**
>
> The HSRP virtual IP address must be reachable and in active state during a changeover.

This figure illustrates a wired and wireless network that supports HSRP-aware mDNS Service-Routing:

Figure 54: HSRP-Aware mDNS Service-Routing



The HSRP offers these advantages:

- Automatic gateway selection.
- Rapid switchover.
- Reduces service convergence.

mDNS service-gateway SSO support

Explains mDNS Stateful Switchover (SSO) support on network devices configured in Service Peer role.

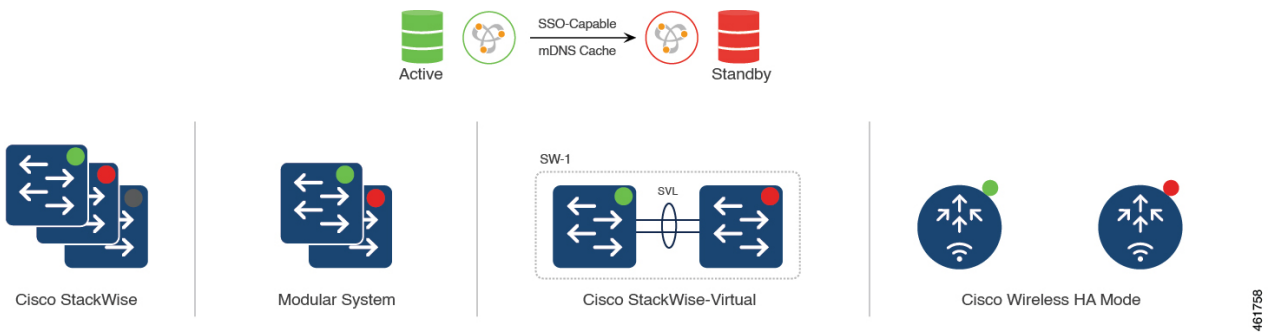
mDNS service-gateway SSO support is a high availability feature that

- enables mDNS Stateful Switchover (SSO) on network devices configured in Service Peer role
- synchronizes cache information between active and standby devices, and
- ensures continuous mDNS service discovery when the active device fails.

Service-gateway SSO operation

Starting from Cisco IOS XE Bengaluru 17.6.1, mDNS Stateful Switchover (SSO) is supported on network devices configured in Service Peer role. In SSO-enabled devices, one device is selected as an active device and the other as a standby device. The cache information learnt by the active device is synced with the standby device. When the active device fails, the standby device becomes the new active device and continues the mDNS service discovery process.

Figure 55: MDNS Service-Gateway SSO



Configure local and wide area Bonjour domains

Describes how to configure local and wide area Bonjour domains to enable service discovery and communication across different network segments and geographic locations.

How to configure multicast DNS mode for LAN and wired networks

Explains how to configure multicast DNS mode for Local Area Bonjour on LAN and wired networks, providing step-by-step configuration procedures for enabling this network discovery protocol.

This section provides information about how to configure Local Area Bonjour in multicast DNS mode.

Enable mDNS gateway on the device

Configure mDNS gateway on the device to enable multicast DNS functionality.

Enable mDNS gateway functionality on the device to allow multicast DNS service discovery across network segments.

mDNS gateway enables devices to discover and communicate with services across different network segments using multicast DNS protocols.

Before you begin

Follow these steps to configure mDNS on the device:

Procedure

1. **enable**

   Example

   ```
   Device> enable
   ```

   Enables privileged EXEC mode.

   Enter your password, if prompted.

2. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

3. **mdns-sd gateway**

   Example

   ```
   Device(config)# mdns-sd gateway
   ```

   Enables mDNS on the device and enters mDNS gateway configuration mode.

   Enter the following commands in mDNS gateway configuration mode to enable the respective functionalities:

   - **air-print-helper**: Enables IOS devices like iPADs to discover and use older printers that support Bonjour
   - **cache-memory-max**: Configures the percentage memory for cache
   - **ingress-client**: Configures Ingress Client Packet Tuners
   - **rate-limit**: Enables rate limiting of incoming mDNS packets
   - **service-announcement-count**: Configures maximum service advertisement count
   - **service-announcement-timer**: Configures advertisements announce timer periodicity
   - **service-query-count**: Configures maximum query count
   - **service-query-timer**: Configures query forward timer periodicity

   The following CLIs are effective only in service-peer mode:

   - **query-response**
   - **sdg-agent**
   - **service-announcement-count**
   - **service-announcement-timer**
   - **service-mdns-query**
   - **service-query-count**
   - **service-query-timer**
   - **service-receiver-purge**
   - **active-response**

   > **Note**
   >
   > For **cache-memory-max**, **ingress-client**, **rate-limit**, **service-announcement-count**, **service-announcement-timer**, **service-query-count**, and **service-query-timer** commands, you can retain the default value of the respective parameter for general deployments. Configure a different value, if required, for a specific deployment.

4. **exit**

   Example

   ```
   Device(config-mdns-sd)# exit
   ```

Exits mDNS gateway configuration mode.

---

mDNS gateway is now enabled on the device and ready to facilitate multicast DNS service discovery across network segments.

Create custom service definition (GUI)

Configure custom service definition using GUI.

Create a custom service definition to define specific mDNS services that can be discovered and used within your network environment.

Custom service definitions allow you to specify particular service types that devices on your network can advertise and discover through mDNS (multicast Domain Name System). This is useful for managing which services are available for discovery in your network infrastructure.

Before you begin

Follow these steps to create custom service definition using GUI:

Procedure

---

1. Choose **Configuration** > **Services** > **mDNS** > **Service Policy** > **Service Definition**.
2. Click **Add**.
3. Enter the **Service Definition Name** and **Description**.
4. Enter the **Service Type** and click the **+** icon.
5. Click **Apply to Device**.

The custom service definition is created and applied to the device. The new service definition will be available for use in mDNS service policies and can be discovered by devices on the network.

Create custom service definition

Configure custom service definition using commands.

Create a custom service definition to provide an admin-friendly name to one or more mDNS service types or PTR Resource Record Names.

Service definition is a construct that provides an admin friendly name to one or more mDNS service types or PTR Resource Record Name. By default, a few built-in service definitions are already predefined and available for admin to use. In addition to built-in service definitions, admin can also define custom service definitions.

Procedure

---

1. **enable**

Example

```
Device> enable
```

Enables privileged EXEC mode.

Enter your password, if prompted.

2. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

3. **mdns-sd service-definition** *service-definition-name*

   Example

   ```
   Device(config)# mdns-sd service-definition CUSTOM1
   ```

   Configures mDNS service definition.

   > **Note**
   >
   > All the created custom service definitions are added to the primary service list. Primary service list comprises of a list of custom and built-in service definitions.

4. **service-type** *string*

   Example

   ```
   Device(config-mdns-ser-def)# service-type _custom1._tcp.local
   ```

   Configures mDNS service type.

5. Repeat step 4 to configure more than one service type in the custom service definition.
6. **exit**

   Example

   ```
   Device(config-mdns-ser-def)# exit
   ```

   Exit mDNS service definition configuration mode.

The custom service definition is created and added to the primary service list, making it available for use in mDNS configurations.

Create a service list (GUI)

Configure a service list using the GUI interface.

Create service lists to control mDNS service discovery by specifying which services are allowed or denied in the network.

Service lists provide a mechanism to filter mDNS services based on service types and message types, helping you manage service discovery traffic in your network.

Procedure

1. Choose **Configuration** > **Services** > **mDNS** > **Service Policy** > **Service List**.
2. Click **Add**.
3. Enter the **Service List Name** and choose the direction from the **Direction** drop-down list.
4. Click **Add Service**.
5. Choose the service from the **Available Services** drop-down list and the message type from the **Message Type** drop-down list.
6. Click **Save**.
7. Click **Apply to Device**.

The service list is created and applied to the device, enabling mDNS service filtering based on your configured parameters.

Create service list

Configure mDNS service list using commands.

Create an mDNS service list to organize service definitions and control service discovery traffic.

mDNS service list is a collection of service definitions. Use service lists to manage which services are allowed IN or OUT of specific network segments.

Before you begin

Follow these steps to create a service list:

Procedure

1. **enable**

   Example

   ```
   Device> enable
   ```

   Enables privileged EXEC mode.

   Enter your password, if prompted.

2. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

3. **mdns-sd service-list** *service-list-name* {**IN** | **OUT**}

   Example

   ```
   Device(config)# mdns-sd service-list VLAN100-list in
   ```

   Configures mDNS service list.

4. **match** *service-definition-name* [**message-type** {**any** | **announcement** | **query**}]

Example

```
Device(config-mdns-sl-in)# match PRINTER-IPPS message-type announcement
```

Matches the service to the message type. Here, *service-definition-name* refers to the names of services, such as, airplay, airserver, airtunes, and so on.

> **Note**
>
> To add a service, the service name must be part of the primary service list.
>
> If the mDNS service list is set to IN, the applicable command syntax is: **match** *service-definition-name* [**message-type** {**any** | **announcement** | **query**}].
>
> If the mDNS service list is set to OUT, the applicable command syntax is: **match** *service-definition-name* [**message-type** {**any** | **announcement** | **query**}] [**location-filter** *location-filter-name*] [**source-interface** {*mDNS-VLAN-number* | *mDNS-VLAN-range*}].

5. **exit**

Example

```
Device(config-mdns-sl-in)# exit
```

Exits mDNS service list configuration mode.

The mDNS service list is created and configured with the specified service definitions and message types.

Create service policy (GUI)

Create service policy using GUI.

Create a service policy to manage mDNS services and control service discovery between network segments.

Service policies define how mDNS services are handled across different network locations by specifying input and output service lists.

Procedure

---

1. Choose **Configuration** > **Services** > **mDNS** > **Service Policy** > **Service Policy**.
2. Click **Add**.
3. Enter the **Service Policy Name**.
4. Choose the service list input from the **Service List Input** drop-down list.
5. Choose the service list output from the **Service List Output** drop-down list.
6. Choose the location from the **Location** drop-down list.
7. Click **Apply to Device**.

The service policy is created and applied to the device. The mDNS service discovery is now controlled according to the configured policy parameters.

Create service policy

Configure service policy using commands.

Create a service policy to specify allowed Bonjour service announcements or queries that should be processed IN ingress or egress directions.

A Service Policy that is applied to an interface specifies the allowed Bonjour service announcements or the queries of specific service types that should be processed, IN ingress direction or egress direction or both. For this, the service policy specifies two service-lists, one each for ingress and egress directions. IN the Local Area Bonjour domain, the same service policy can be attached to one or more Bonjour client VLANs; however, different VLANs may have different service policies.

Before you begin

Follow these steps to configure service policy with service lists:

Procedure

1. **enable**

   Example

   ```
   Device> enable
   ```

   Enables privileged EXEC mode.

   Enter your password, if prompted.

2. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

3. **mdns-sd service-policy** *service-policy-name*

   Example

   ```
   Device(config)# mdns-sd service-policy mdns-policy1
   ```

   Configures mDNS service policy.

4. **service-list** *service-list-name* {**IN** | **OUT**}

   Example

   ```
   Device(config-mdns-ser-pol)# service-list VLAN100-list in
   ```

   ```
   Device(config-mdns-ser-pol)# service-list VLAN300-list out
   ```

   Configures service lists for IN and OUT directions.

5. **exit**

Example

```
Device(config-mdns-ser-pol)# exit
```

Exits mDNS service policy configuration mode.

The mDNS service policy is created and configured with service lists for ingress and egress directions.

Associate service policy to an interface

Configure mDNS gateway on an interface and associate service policy to enable mDNS functionality.

Associate a service policy to an interface to enable mDNS (multicast DNS) functionality and govern Bonjour announcements and queries.

mDNS configuration on the device requires associating service policies to interfaces to control Bonjour service behavior on VLANs.

Before you begin

Follow these steps to configure mDNS on the device:

Procedure

---

1. **enable**

   Example

   ```
   Device> enable
   ```

   Enables privileged EXEC mode.

   Enter your password, if prompted.

2. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

3. **interface** *interface-name*

   Example

   ```
   Device(config)# interface Vlan 601
   ```

   Enters interface mDNS configuration mode and enables interface configuration.

4. **mdns-sd gateway**

   Example

   ```
   Device(config-if)# mdns-sd gateway
   ```

Configures mDNS gateway on the interface.

Enter these commands in the interface mDNS gateway configuration mode to enable the respective functionalities:

- **active-query**: Sets the time interval for SDG agent to refresh the active status of connected Bonjour client services. The timer value ranges from 60 to 3600 seconds.

  > **Note**
  >
  > This configuration is mandatory only on VLANs whose Bonjour policy is configured to accept Bonjour service announcements from connected Bonjour clients. If the VLAN is configured to only accept Bonjour queries but not Bonjour service announcements, this configuration is optional.

- **service-instance-suffix**(Optional) : Appends the service instance suffix to any announced service name that is forwarded to the controller.
- **service-mdns-query** [**PTR** | **all**]: Configures mDNS query request message processing for the specified query types. This command is applicable when the controller is in service-peer mode.

  > **Note**
  >
  > By default, the **service-mdns-query** command allows only PTR queries. If you need to respond to all (PTR, SRV, and TXT) queries, you need to execute this command:
  >
  > **service-mdns-query all**

- **service-policy** *policy-name*: Attaches the specified service policy to the VLAN. Bonjour announcements, and queries received by and sent from the VLAN are governed by the policies configured in the service policy. This configuration is mandatory for all VLANs.

  > **Note**
  >
  > Service policies can only be attached at interface level.

- **transport** [**all** | **ipv4** | **ipv6**] (Optional): Configures BCP parameter.

  It is recommended to use **transport ipv4** command, except in those networks where the Bonjour clients send only IPv6 announcements and queries.

5. **exit**

Example

```
Device(config-if-mdns-sd)# exit
```

Exits mDNS gateway configuration mode.

The mDNS gateway is configured on the interface with the service policy attached. The interface can now process Bonjour announcements and queries according to the configured policies.

Local area Bonjour in multicast DNS mode for wireless networks

Describes the configuration requirements for enabling local area Bonjour in multicast DNS mode within wireless network environments. This process ensures seamless communication between wireless endpoints and SDG Agent switches.

Local area Bonjour in multicast DNS mode is a service discovery framework that relies on mDNS messages transmitted over UDP port 5353 to facilitate communication between network endpoints.

- Operates using service announcements and queries sent to mDNS IPv4 address 224.0.0.251 and IPv6 address FF02::FB.
- Functions as a Layer 2 protocol, requiring clients to reside within the same Layer 2 domain for service discovery.
- Requires transparent mDNS transmission support from intermediate WLCs to bridge wireless endpoints and SDG Agents.

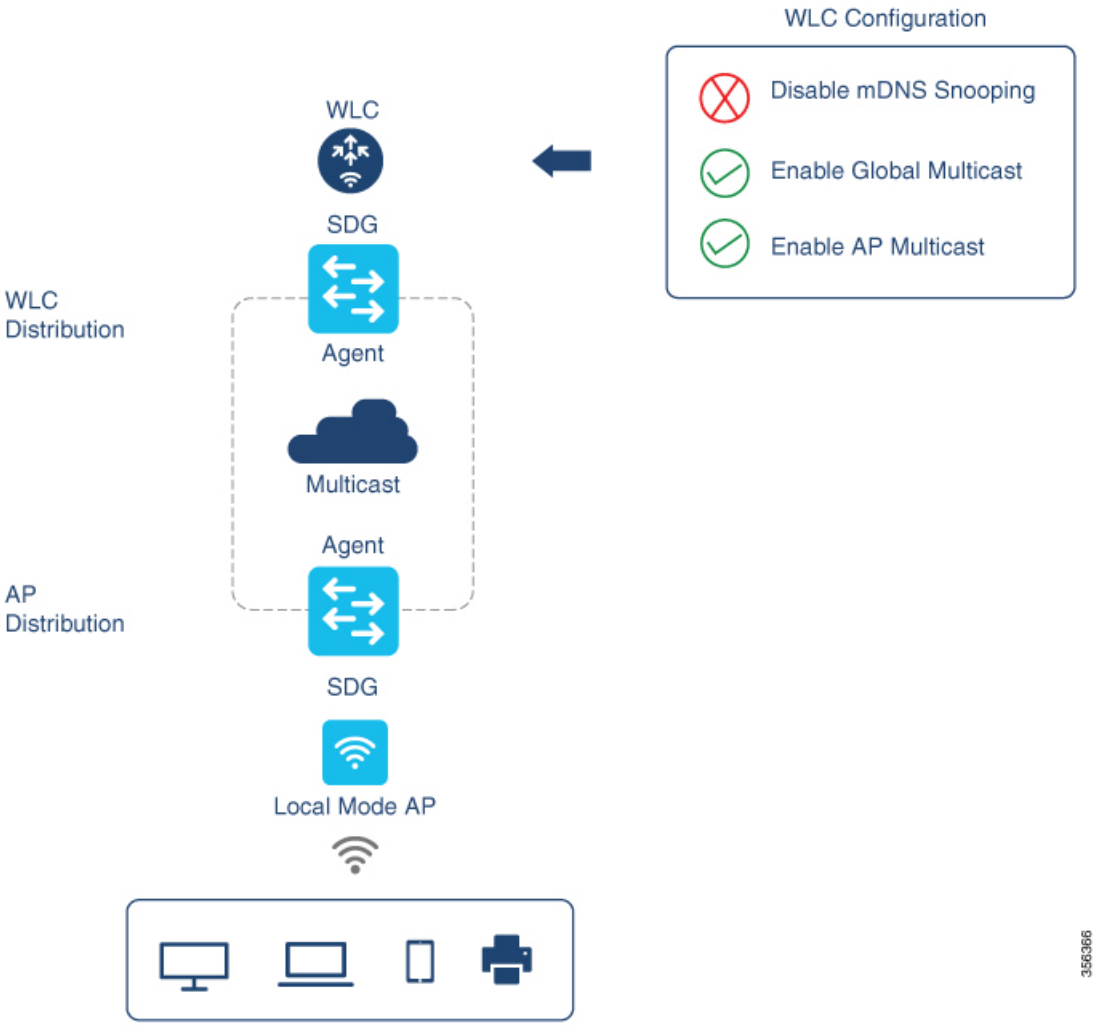
Wireless Network Configuration Requirements

To enable mDNS communication between wireless endpoints and a Cisco Catalyst switch acting as an SDG Agent, perform the following configuration steps:

1. Disable mDNS Snooping on Cisco AireOS based WLCs.
2. Enable the mDNS Gateway feature on Cisco Catalyst 9800 series WLCs.
3. Set the AP Multicast Mode to Multicast.

The following figure illustrates the prerequisite configuration for wireless networks to enable communication between SDG-Agent switches and wireless endpoints.

Figure 56: Wireless Network Prerequisite Configuration



The configuration of local area Bonjour on a switch that acts as the SDG Agent in a wireless network involves the same set of procedures that are used to configure local area Bonjour on a switch that acts as the SDG Agent in a wired network.

The Bonjour protocol operates on service announcements and queries. Each query or advertisement is sent to the mDNS IPv4 address 224.0.0.251 and IPv6 address FF02::FB. The mDNS messages are carried over well-known industry standard UDP port 5353, over both Layer 3 transport types.

The Layer 2 address used by the Bonjour protocol is link-local multicast address and therefore it's only forwarded to the same Layer 2 network. As multicast DNS (mDNS) is limited to a Layer 2 domain, for a client to discover a service, it has to be a part of the same Layer 2 domain. This isn't always possible in a large-scale deployment or enterprise.

To enable mDNS communication between Wireless endpoints and Cisco Catalyst switch that acts as an SDG Agent, the intermediate WLC must transparently allow the network to transmit and receive mDNS messages.

Hence, for a Multicast DNS Mode Wireless network deployment, disable the mDNS Snooping on Cisco AireOS based WLC and enable mDNS Gateway feature on Cisco Catalyst 9800 series WLC and set the AP Multicast Mode to Multicast.

The Cisco WLC and Access Points by default prevent the forwarding of Layer 2 or Layer 3 Multicast frames between Wireless and Wired network infrastructure. The forwarding is supported with stateful capabilities enabled using AP Multicast. The network administrator must globally enable Multicast and configure a unique Multicast Group to advertise in the network. This multicast group is only required for Cisco Access Points to enable Multicast over Multicast (MCMC) capabilities across the LAN network. The Bonjour solution doesn't require any Multicast requirements on Wireless Client VLAN; thus, it's optional and applicable only for other Layer 3 Multicast applications.

The core network must be configured with appropriate Multicast routing to allow the Access Points to join WLC Multicast Group. The Multicast configuration must be enabled on Cisco WLC management VLAN and on the Cisco Access Points of their respective distribution layer switch.

Associate service policy with wireless profile policy

Configure association of service policy with wireless profile policy to override the default mDNS service policy.

This task allows you to override the default mDNS service policy with a custom service policy of your choice.

A default mDNS service policy is already attached once the wireless profile policy is created. Use this procedure to override the default mDNS service policy with any of your service policy.

Before you begin

Follow these steps to associate service policy with wireless profile policy:

Procedure

1. **enable**

   Example

   ```
   Device> enable
   ```

   Enables privileged EXEC mode.

   Enter your password, if prompted.

2. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

3. **wireless profile policy** *profile-policy-name*

   Example

   ```
   Device(config)# wireless profile policy default-policy-profile
   ```

   Configures wireless profile policy.

4. **mdns-sd service-policy** *custom-mdns-service-policy*

   Example

   ```
   Device(config-wireless-policy)# mdns-sd service-policy
   custom-mdns-service-policy
   ```

   Associates an mDNS service policy with the wireless profile policy.

   The default mDNS service policy name is **default-mdns-service-policy**.

5. **exit**

Example

```
Device(config-wireless-policy)# exit
```

Exits wireless profile policy configuration mode.

The custom mDNS service policy is now associated with the wireless profile policy, overriding the default service policy.

Configure wide area Bonjour domain

Describes how to configure wide area Bonjour domain parameters on the controller, including service types to export from the SDG Agent using service-lists and egress policies.

The Wide Area Bonjour domain configuration specifies the parameters of the controller, that is the Wide Area Bonjour Application running on Cisco Catalyst Center, as well as the service types that need to be exported to it from the SDG Agent. Configuring Wide Area Bonjour Domain involves creating service-lists and service policy similar to those created in Local Area Bonjour configuration; however, only egress policy from SDG Agent to controller is applicable.

Associate service policy with the controller in wide area Bonjour domain

Configure service policy association with the controller in wide area Bonjour domain for global service management.

This task enables the association of service policies with controllers in Wide Area Bonjour domains to manage service discovery and distribution across the network.

In Wide Area Bonjour, the service policy is configured globally and does not get associated with a VLAN as in the case of Local Area Bonjour.

Before you begin

Follow these steps to configure service policy globally:

Procedure

---

1. **enable**

   Example

   ```
   Device> enable
   ```

   Enables privileged EXEC mode.

   Enter your password, if prompted.

2. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

3. **service-export mdns-sd controller** *controller name*

Example

```
Device(config)# service-export mdns-sd controller Cisco Catalyst
Center-BONJOUR-CONTROLLER
```

Specifies a name for the controller and enters service-export mode

4. **controller-address** *ipv4-address*

   Example

   ```
   Device(config-mdns-sd-se)# controller-address 199.245.1.7
   ```

   Specifies the controller address.

5. **controller-port** *port-number*

   Example

   ```
   Device(config-mdns-sd-se)# controller-port 9991
   ```

   Specifies the port number on which the controller is listening.

6. **controller-source-interface** *interface-name*

   Example

   ```
   Device(config-mdns-sd-se)# controller-source-interface Loopback0
   ```

   Specifies the source-interface for the controller.

7. **controller-service-policy** *service-policy-name* **OUT**

   Example

   ```
   Device(config-mdns-sd-se)# controller-service-policy policy1 OUT
   ```

   Specifies the service policy to be used by the controller.

   > **Note**
   >
   > Only OUT policy is applicable for Wide Area Bonjour.

8. **exit**

   Example

   ```
   Device(config-mdns-sd)# exit
   ```

   Exits controller service export configuration mode.

9. **mdns-sd gateway**

Example

```
Device(config)# mdns-sd gateway
```

Enters mDNS gateway configuration mode.

10. **ingress-client query-suppression enable**

Example

```
Device(config-mdns-sd)# ingress-client query-suppression enable
```

Enables ingress query suppression for better scale and performance.

11. **exit**

Example

```
Device(config-mdns-sd)# exit
```

Exits mDNS gateway configuration mode.

The service policy is successfully associated with the controller in the Wide Area Bonjour domain, enabling global service discovery management and ingress query suppression.

Configure HSRP-aware mDNS service-routing on SDG

Lists the configuration guides for hot standby router protocol-aware (HSRP-aware) mDNS service-routing on SDG.

Configure hot standby router protocol-aware (HSRP-aware) mDNS service-routing on SDG using the referenced configuration guides for Catalyst switch platforms.

For information, see these guides:

- Software Configuration Guide, Cisco IOS XE Bengaluru 17.6.x (Catalyst 9300 Switches)
- Software Configuration Guide, Cisco IOS XE Bengaluru 17.6.x (Catalyst 9400 Switches)
- Software Configuration Guide, Cisco IOS XE Bengaluru 17.6.x (Catalyst 9600 Switches)

Configure HSRP-aware mDNS service-routing on service-peer

Configure hot standby router protocol-aware (HSRP-aware) mDNS service-routing on service-peer using commands.

Configure HSRP-aware mDNS service-routing on service-peer to enable multicast DNS gateway functionality with hot standby router protocol awareness.

Use this procedure when you need to configure mDNS gateway in service-peer mode with HSRP awareness to provide redundancy and high availability for mDNS services.

Before you begin

Follow these steps to configure hot standby router protocol-aware mDNS service-routing on service-peer:

Procedure

---

1. **configure terminal**

Example

```
Device# configure terminal
```

Enters global configuration mode.

2. **mdns-sd gateway**

   Example

   ```
   Device(config)# mdns-sd gateway
   ```

   Enables mDNS on the device and enters mDNS gateway configuration mode.

3. **mode service-peer**

   Example

   ```
   Device(config-mdns-sd)# mode service-peer
   ```

   Enables mDNS gateway in service-peer mode.

4. **source-interface vlan** *vlan-interface-number*

   Example

   ```
   Device(config-mdns-sd)# source-interface vlan 23
   ```

   Configures source interface to communicate between SDG Agent and service-peer.

   > **Note**
   >
   > *vlan-interface-number*- The valid range is from1 to 4094.

5. **SDG-agent** *IP-address*

   Example

   ```
   Device(config-mdns-sd)# sdg-agent 9.6.16.10
   ```

   Configures SDG agent IPv4 or IPv6 address.

   > **Note**
   >
   > The *IP-address* refers to the HSRP-enabled IP address.

6. **end**

Example

```
Device(config-mdns-sd)# end
```

Exits server group configuration mode and enters privileged EXEC mode.

The device is configured with HSRP-aware mDNS service-routing on service-peer mode, enabling multicast DNS gateway functionality with hot standby router protocol support.

Verify local area bonjour in multicast DNS mode for LAN and wireless networks

Describes how to verify Local Area Bonjour in multicast DNS mode for both LAN and wireless networks to ensure proper service discovery functionality.

This section shows how to verify Local Area Bonjour in Multicast DNS mode for LAN and Wireless networks.

Verify SDG-agent status

Provides SDG-agent status verification information.

Use the sample outputs to verify SDG-agent status and understand the command output format for various mDNS service discovery commands.

This is a sample output of the **show mdns-sd service-list***service-list-name* {**in** | **out**} command.

```
Name            Direction  Service    Message-Type     Source
=============================================================
VLAN100-list     In         Printer   Announcement      -
                 In         Airplay   Query             -
                 In         CUSTOM1   Any               -
VLAN300-list     Out        Printer   Announcement     Vl200
```

This is a sample output of the **show mdns-sd service-definition***service-definition-name***service-type** {*custom* | *built-in*} command.

```
Service                 PTR                          Type
=====================================================================
apple-tv                _airplay._tcp.local          Built-In
                         _raop._tcp.local
apple-file-share       _afpovertcp._tcp.local       Built-In
CUSTOM1                _custom1._tcp.local          Custom
CUSTOM2                _customA._tcp.local          Custom
                       _customA._tcp.local
```

This is a sample output of the **show mdns-sd***service-policy-name***interface***interface-name* command.

```
Name   Service-List-In   Service-List-Out
================================================
mdns-policy-1  VLAN100-list   VLAN300-list
mdns-policy-2  VLAN400-list   VLAN400-list
```

This is a sample output of the **show mdns-sd summary** command.

```
mDNS Gateway: Enabled
Mode: Service Peer
Service Announcement Periodicity(in seconds): 30
Service Announcement Count: 50
Service Query Periodicity(in seconds): 15
Service Query Count: 50
Active Response Timer (in seconds): Disabled
ANY Query Forward: Disabled
SDG Agent IP: 9.8.57.10
Active Query Periodicity (in minutes): 30
mDNS Query Type: PTR only
Transport Type: IPv4
mDNS AP service policy: default-mdns-service-policy
```

This is a sample output of the **show mdns-sd sp-SDG statistics** command.

```
mDNS SP Statistics
last reset time: 07/27/21 15:36:33
Messages sent:
Query : 122
ANY query : 35
Advertisements : 12
Advertisement Withdraw : 1
Service-peer cache clear : 0
Resync response : 3
Srvc Discovery response : 0
Keep-Alive : 2043
Messages received:
Query response : 0
ANY Query response : 0
Cache-sync : 9
Get service-instance : 0
Srvc Discovery request : 0
Keep-Alive Response : 2042
```

Verify wide area Bonjour controller status

Provides commands and sample outputs to verify wide area Bonjour controller status.

Verify wide area Bonjour controller status using show commands to display controller summary, export summary, statistics, and detailed information.

This is a sample output of the **show mdns controller summary** command.

```
Device# show mdns controller summary

Controller Summary
====================================
 Controller Name  :   Cisco Catalyst Center-BONJOUR-CONTROLLER
 Controller IP    :   10.104.52.241
 State            :   UP
 Port             :   9991
 Interface        :   Loopback0
 Filter List      :   policy1
 Dead Time        :   00:01:00
```

This is a sample output of the **show mdns controller export-summary** command.

```
Device# show mdns controller export-summary

Controller Export Summary
=========================
 Controller IP    :   10.104.52.241
 State            :   UP
 Filter List      :   policy1
 Count            :   100
 Delay Timer      :   30 seconds
 Export           :   300
 Drop             :   0
 Next Export      :   00:00:01
```

This is a sample output of the **show mdns controller statistics** command.

```
Device# show mdns controller statistics

Total BCP message sent           : 47589
  Total BCP message received       : 3
  Interface WITHDRAW messages sent : 0
  Clear cache messages sent        : 0
  Total RESYNC state count         : 0
  Last successful RESYNC           : Not-Applicable

  Service Advertisements:
   IPv6 advertised                 : 0
   IPv4 advertised                 : 300
   Withdraws sent                  : 0
   Advertisements Filtered         : 0
   Total service resynced          : 0

  Service Queries:
   IPv6 queries sent               : 0
   IPv6 query responses received   : 0
   IPv4 queries sent               : 0
   IPv4 query responses received   : 0
```

This is a sample output of the **show mdns controller detail** command.

```
Device# show mdns controller detail

Controller : Cisco Catalyst Center-BONJOUR-CONTROLLER
 IP : 10.104.52.241, Dest Port : 9991, Src Port : 0, State : UP
 Source Interface : Loopback0, MD5 Disabled
 Hello Timer 0 sec, Dead Timer 0 sec, Next Hello 00:00:00
 Uptime 00:00:00
Service Announcement :
 Filter : policy1
 Count 100, Delay Timer 30 sec, Pending Announcement 0, Pending Withdraw 0
 Total Export Count 300, Next Export in 00:00:16
Service Query :
 Query Suppression Disabled
 Query Count 50, Query Delay Timer 15 sec, Pending 0
 Total Query Count 0, Next Query in 00:00:01
```

Verify mDNS cache configurations

Provides verification commands for mDNS cache configurations.

This reference helps you verify mDNS cache configurations using show commands that display cache information from both Active and Standby devices.

These show commands display cache from both Active and Standby devices using the chassis option:

```
Device# show mdns-sd cache chassis active R0

------------------------------------------------------------ PTR Records
------------------------------------------------------------
RECORD-NAME                                  TTL      TYPE      ID
CLIENT-MAC       RR-RECORD-DATA
------------------------------------------------------------------------------------------------------------
_home-sharing._tcp.local                     4500     WLAN      1
0205.2c23.0001   AP6B8B4567-sta00001._home-sharing._tcp.local

------------------------------------------------------------ SRV Records
------------------------------------------------------------
RECORD-NAME                                  TTL      TYPE      ID
CLIENT-MAC       RR-RECORD-DATA
------------------------------------------------------------------------------------------------------------
AP6B8B4567-sta00001._home-sharing._tcp.local   4500     WLAN      1
0205.2c23.0001   0 0 5353 AP6B8B4567-sta00001.local

------------------------------------------------------------ A/AAAA Records
------------------------------------------------------------
RECORD-NAME                                  TTL      TYPE      ID
CLIENT-MAC       RR-RECORD-DATA
------------------------------------------------------------------------------------------------------------
AP6B8B4567-sta00001.local                    4500     WLAN      1
0205.2c23.0001   9.2.57.106

------------------------------------------------------------ TXT Records
------------------------------------------------------------
RECORD-NAME                                  TTL      TYPE      ID
CLIENT-MAC       RR-RECORD-DATA
------------------------------------------------------------------------------------------------------------
AP6B8B4567-sta00001._home-sharing._tcp.local   4500     WLAN      1
0205.2c23.0001   [14]'model=MacMini'
```

Note

Alternatively, you can issue the **show mdns-sd cache** command to display the cache from the Active controller.

```
Device# show mdns-sd cache chassis standby R0

------------------------------------------------------------ PTR Records
------------------------------------------------------------
RECORD-NAME                                  TTL      TYPE      ID
CLIENT-MAC       RR-RECORD-DATA
------------------------------------------------------------------------------------------------------------
_home-sharing._tcp.local                     4500     WLAN      1
0205.2c23.0001   AP6B8B4567-sta00001._home-sharing._tcp.local

------------------------------------------------------------ SRV Records
```

```
-------------------------------------------------------------------
RECORD-NAME                                    TTL      TYPE      ID
CLIENT-MAC       RR-RECORD-DATA
-------------------------------------------------------------------------------------------------------------
AP6B8B4567-sta00001._home-sharing._tcp.local   4500     WLAN      1
0205.2c23.0001   0 0 5353 AP6B8B4567-sta00001.local

------------------------------------------------------------- A/AAAA Records
-------------------------------------------------------------------
RECORD-NAME                                    TTL      TYPE      ID
CLIENT-MAC       RR-RECORD-DATA
-------------------------------------------------------------------------------------------------------------
AP6B8B4567-sta00001.local                      4500     WLAN      1
0205.2c23.0001   9.2.57.106

------------------------------------------------------------- TXT Records
-------------------------------------------------------------------
RECORD-NAME                                    TTL      TYPE      ID
CLIENT-MAC       RR-RECORD-DATA
-------------------------------------------------------------------------------------------------------------
AP6B8B4567-sta00001._home-sharing._tcp.local   4500     WLAN      1
0205.2c23.0001   [14]'model=MacMini'
```

Verify additional mDNS cache configurations

Lists commands to verify additional mDNS cache configurations from Active and Standby databases.

This reference provides commands to verify mDNS cache configurations from both Active and Standby database states.

Commands for Active DB verification

To verify the cache from the Active DB, use these commands:

```
show mdns-sd cache ap-mac 0a0b.0cf0.000e chassis active R0
show mdns-sd cache client-mac 0269.fe06.0023 chassis active R0
show mdns-sd cache detail chassis active R0
show mdns-sd cache glan-id <> chassis active R0
show mdns-sd cache glan-id <> detail chassis active R0
show mdns-sd cache location-group <> chassis active R0
show mdns-sd cache location-group <> detail chassis active R0
show mdns-sd cache mdns-ap <> detail chassis active R0
show mdns-sd cache mdns-ap <> chassis active R0
show mdns-sd cache rlan-id <> detail chassis active R0
show mdns-sd cache rlan-id <> chassis active R0
show mdns-sd cache type TXT chassis active R0
show mdns-sd cache type A-AAAA detail chassis active R0
show mdns-sd cache wired chassis active R0
show mdns-sd cache wired detail chassis active R0
show mdns-sd cache wlan-id 10 chassis active R0
show mdns-sd cache wlan-id 1 detail chassis active R0
```

Commands for Standby DB verification

To verify the cache from the Standby DB, use these commands:

```
show mdns-sd cache ap-mac <> chassis standby R0
show mdns-sd cache client-mac <> chassis standby R0
```

```
show mdns-sd cache detail chassis standby R0
show mdns-sd cache glan-id <> chassis standby R0
show mdns-sd cache glan-id <> detail chassis standby R0
show mdns-sd cache location-group <> chassis standby R0
show mdns-sd cache location-group <> detail chassis standby R0
show mdns-sd cache mdns-ap <> detail chassis standby R0
show mdns-sd cache mdns-ap <> chassis standby R0
show mdns-sd cache rlan-id <> detail chassis standby R0
show mdns-sd cache rlan-id <> chassis standby R0
show mdns-sd cache type [A-AAAA|PTR|SRV|TXT] chassis standby R0
show mdns-sd cache type [A-AAAA|PTR|SRV|TXT] detail chassis standby R0
show mdns-sd cache wired chassis standby R0
show mdns-sd cache wired detail chassis standby R0
show mdns-sd cache wlan-id <> chassis standby R0
show mdns-sd cache wlan-id <> detail chassis standby R0
```

Verify local area Bonjour configuration for LAN and wireless networks

Provides verification of local Area Bonjour configuration for LAN and wireless networks.

This reference provides sample output of the **show run** command to verify local area Bonjour configuration for LAN and wireless networks.

```
mdns-sd gateway

mdns-sd service-definition custom1
 service-type _airplay._tcp.local
 service-type _raop._tcp.local


mdns-sd service-list list1 IN
 match custom1
mdns-sd service-list list2 OUT
 match custom1


mdns-sd service-policy policy1
service-list list1 IN
service-list list2 OUT


service-export mdns-sd controller Cisco Catalyst Center-CONTROLLER-POLICY
controller-address 99.99.99.10
controller-service-policy policy1 OUT
controller-source-interface Loopback0
```

Additional references for DNA Service for Bonjour

Provides supplementary reference materials and feature history information for local and wide area Bonjour implementations.

This reference provides additional resources for DNA service for Bonjour, including related documentation and MIB information.

Related Topic	Document Title
CISCO Wide Area Bonjour Application on Cisco Catalyst Center User Guide	CISCO Wide Area Bonjour Application on Cisco Catalyst Center User Guide, Release 1.3.1.0

MIBs

MIB	MIBs Link
CISCO-SDG-MDNS-MIB	This MIB module defines objects describing the statistics of 63 local area and wide area MDNS SDG agent. Statistics could be 64 either global or per interface specific.

Feature history for Cisco DNA service for Bonjour

Lists release information about Cisco DNA service for Bonjour features and platform support.

This table provides release information about the feature or features described in this module. This table lists only the software release that introduced support for a given feature in a given software release train. Unless noted otherwise, subsequent releases of that software release train also support that feature.

Release	Modification
Cisco IOS 15.2(6) E2	Cisco DNA Service for Local Area Bonjour and Wide Area Bonjour was introduced on the following platforms: • Cisco Catalyst 2960-X Series Switches • Cisco Catalyst 2960-XR Series Switches
Cisco IOS 15.5(1)SY4	Cisco DNA Service for Local Area Bonjour and Wide Area Bonjour was introduced on Cisco Catalyst 6800 Series Switches.
Cisco IOS XE 3.11.0 E	Cisco DNA Service for Local Area Bonjour and Wide Area Bonjour was introduced on the following platforms: • Cisco Catalyst 4500-E Series Switches • Cisco Catalyst 4500-X Series Switches

Release	Modification
Cisco IOS XE Gibraltar 16.11.1	Cisco DNA Service for Local Area Bonjour and Wide Area Bonjour was introduced on the following platforms: • Cisco Catalyst 3650 Series Switches • Cisco Catalyst 3850 Series Switches • Cisco Catalyst 9300 Series Switches • Cisco Catalyst 9400 Series Switches • Cisco Catalyst 9500 Series Switches • Cisco Catalyst 9500 Series Switches - High Performance • Cisco Catalyst 9600 Series Switches • Cisco Catalyst 9800 Series Wireless Controllers • Cisco 5500 Series Wireless Controllers • Cisco 8540 Wireless Controllers • Cisco 4000 Series Integrated Services Routers (ISR)
Cisco IOS XE Amsterdam 17.1.1	Cisco DNA Service for Local Area Bonjour and Wide Area Bonjour was introduced on Cisco Catalyst 9200 Series Switches.
Cisco IOS XE Amsterdam 17.2.1	Introduced Cisco DNA Service for Bonjour support for the following: • SD-Access network • Unicast mode for LAN network
Cisco IOS XE Amsterdam 17.3.2a	Introduced Cisco DNA Service for Bonjour support for the following: • Multilayer networks • Location grouping in wired networks • mDNS AP group in wireless networks

Release	Modification
Cisco IOS XE Bengaluru 17.6.1	Introduced support for the following features for Local Area Bonjour in Unicast Mode for LAN networks: • Default mDNS Service Configurations • HSRP-Aware mDNS Service-Routing • mDNS Service-Gateway SSO Support

231 Configuring Local Area Bonjour for Wireless Local Mode

Topics:

- Local area Bonjour for wireless local mode
- Wireless AP multicast
- Configure local area Bonjour for wireless local mode
- Verify mDNS gateway configuration

Local area Bonjour for wireless local mode

Explains Local Area Bonjour functionality for wireless local mode, detailing prerequisites, operational restrictions, and fundamental concepts for implementation planning.

Local Area Bonjour for wireless local mode is a network functionality that

- introduces unicast mode function in Local Area Bonjour network domain
- enables enhanced gateway function at the first hop of Wired and Wireless networks to communicate directly with any industry standard RFC 6762 compliant Multicast DNS (mDNS) end point in Layer 2 Unicast mode, and
- introduces new service-peer mode expanding classic single-gateway controller to end-to-end service-routing with upstream SDG agent switch to enable unicast-mode, increased scale, performance and resiliency in the network.

Prerequisites for local area Bonjour for wireless local mode

Outlines prerequisites for local area Bonjour for wireless local mode.

Ensure that the Cisco Catalyst 9800 series controller is successfully configured and operational before implementing Cisco Local Area Bonjour for local mode wireless networks. This is necessary for the controller to be deployed in service-peer mode

- Ensure that the targeted controller for the service-peer role has the required Cisco IOS-XE software version. See *Supported SDG Agents with Supported Licenses and Software Requirements* table in Cisco DNA Service for Bonjour Solution Overview chapter.
- Ensure that the controller runs a valid Cisco DNA-Advantage license.
- Ensure that the upstream distribution-layer Cisco Catalyst switch in SDG agent mode runs a valid Cisco DNA-Advantage license.
- Ensure that the controller is interconnected as Layer 2 trunk in static 802.1Q mode, when Layer 2 Unicast service-routing is running between SDG agent in distribution-layer and the controller service-peer.
- Ensure that the controller has IP reachability to upstream Cisco Catalyst 9000 series switches in SDG agent mode over same the IPv4 wireless management subnet.
- Ensure that global multicast is enabled on the controller and AP is set to multicast mode. All local mode APs must join the multicast group in the network to successfully process mDNS messages.

Restrictions for local area Bonjour for wireless local mode

Outlines restrictions for local area Bonjour for wireless local mode.

- Controller management port is not supported for service-routing with upstream Catalyst SDG Agent switch.
- The controller in service peer mode supports location-based service for access points in local mode and FlexConnect central switching mode.
- The controller supports location-based capabilities only between wireless connected service provider and the receiver.
- The controller does not support service-routing configuration using GUI.

Local area Bonjour for wireless local mode

Explains local area Bonjour for wireless local mode.

Local area Bonjour for wireless local mode is a network service that

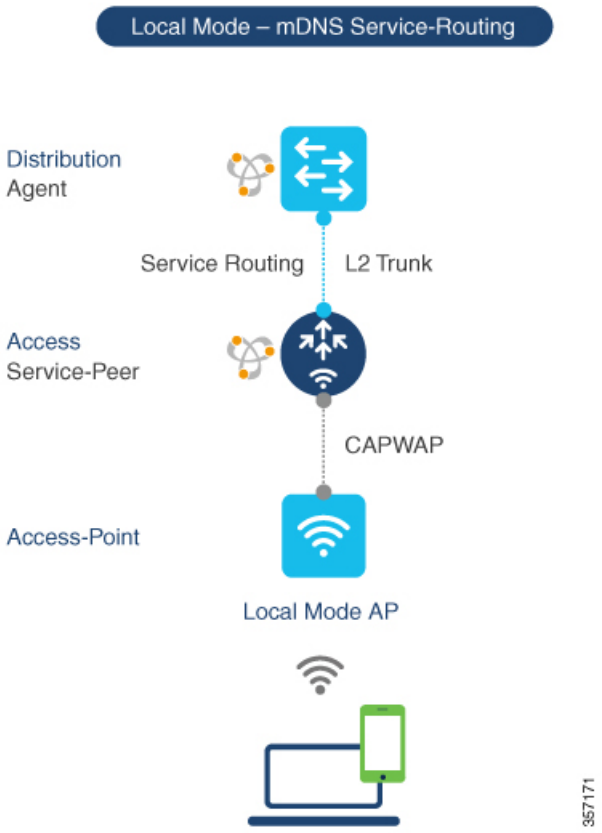
- enables distributed Bonjour gateway function at the network edge through unified Cisco IOS-XE operating system across Cisco Catalyst 9000 series LAN switches and Cisco Catalyst 9800 series controller
- supports mDNS gateway function to wireless users in local mode with service-routing peering to upstream Cisco Catalyst 9000 series switch for network-wide services discovery and distribution, and
- eliminates mDNS flooding over Layer 2 trunk ports through unicast based service-routing between the controller and upstream SDG-Agent switch.

Local area Bonjour capabilities

The traditional wireless controller supported mDNS snooping function with various advancements for wireless networks. As the enterprise requirements expands, IT drives the IT organization to introduce new network deployment models, supporting mobile devices and distributed zero-configuration services with increased scale, granular security control and resiliency for mission critical networks. With end-to-end Wide Area Bonjour service-routing, the new solution enables service-oriented enterprise networks with intuitive user-experience.

The controller platform supporting mDNS gateway function builds service-routing peering with upstream Cisco Catalyst 9000 series switch for network-wide services discovery and distribution based on IT-managed granular policies and locations. The unicast based service-routing between the controller in service-peer mode and upstream SDG-Agent switch provides increased bandwidth and eliminates mDNS flood over wireless networks and Layer 2 trunk to upstream network.

Figure 57: Cisco Catalyst 9800 series controller local area Bonjour for wireless local mode



Wireless AP multicast

Describes the configuration of wireless AP multicast settings through GUI and CLI interfaces, including IP network multicast configuration procedures.

Wireless AP multicast is a network feature that

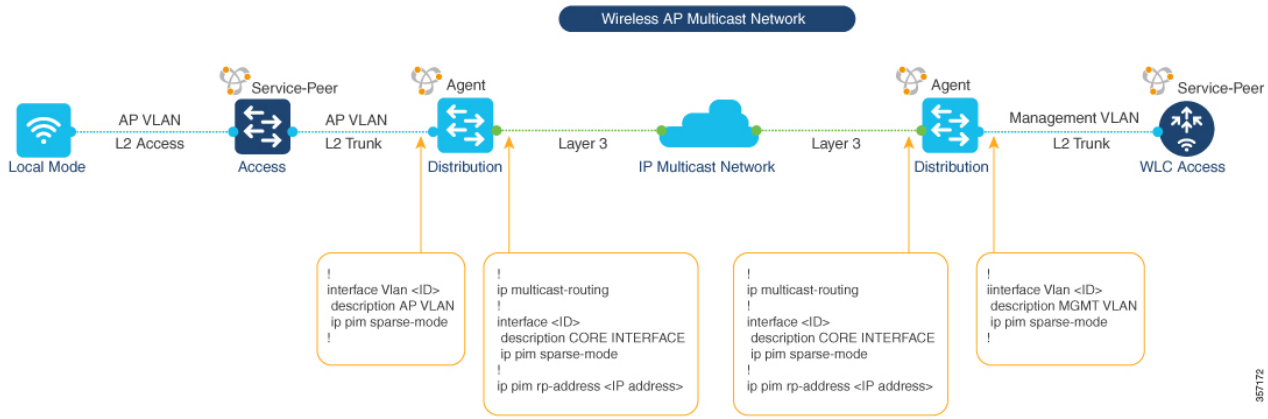
- enables forwarding of Layer 2 or Layer 3 multicast frames between wireless and wired network infrastructure with stateful capabilities
- requires configuration of a unique AP multicast group on the controller to advertise in the IP core network, and
- supports mDNS message processing over a wireless network when multicast is enabled.

Wireless AP multicast requirements

The controller and AP by default prevents forwarding of Layer 2 or Layer 3 Multicast frames between wireless and wired network infrastructure. The forwarding is supported with stateful capabilities enabled using AP multicast. To allow mDNS message processing over a wireless network, multicast must be enabled and unique AP multicast group must be configured on the controller to advertise in IP core network. This AP multicast group is only required for APs to enable Multicast over Multicast (MCMC) capabilities in the network. The Bonjour solution do not require any other multicast requirements on wireless client VLAN; thus, it is optional and applicable only for other Layer 3 multicast applications.

End-to-end wireless multicast configuration is required to ensure wireless APs successfully join the controller-announced multicast group.

Figure 58: Multicast routing in IP core network



Configure wireless AP multicast (GUI)

Configure wireless AP multicast using GUI.

Configure wireless AP multicast settings to enable multicast communication on a controller in service-peer mode.

This procedure configures wireless AP multicast on a controller in service-peer mode.

Procedure

1. Choose **Configuration** > **Services** > **Multicast**.
2. Set the **Global Wireless Multicast Mode** to **Enabled**.
3. From the AP Capwap Multicast drop-down list, select **Multicast**.

4. Enter a unique IP address at **AP Capwap IPv4 Multicast group Address**.
5. Click **Apply**.
6. Click **Save**.

The wireless AP multicast configuration is applied and saved on the controller.

Configure wireless AP multicast (CLI)

Configure wireless AP multicast using commands.

This task configures wireless AP multicast to enable communication between access points and the controller using multicast addresses.

This procedure configures wireless AP multicast on a controller in service-peer mode.

Procedure

---

1. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. **wireless multicast**

   Example

   ```
   Device(config)# wireless multicast
   ```

   Enable global IP multicast processing.

3. **wireless multicast** *IPv4-multicast-address*

   Example

   ```
   Device(config)# wireless multicast 239.254.254.1
   ```

   Enables AP CAPWAP mode to Multicast with unique IPv4 multicast address configurations.

4. **exit**

   Example

   ```
   Device(config-mdns-sd)# exit
   ```

   Exits mDNS gateway configuration mode.

Wireless AP multicast is configured on the controller, enabling multicast communication between access points.

Configure multicast in IP network (CLI)

Configure multicast in IP network using commands.

Configure IP Multicast under AP VLAN, Management VLAN and IP core interfaces on upstream Catalyst LAN distribution-layer switch to enable multicast traffic processing in the network infrastructure.

This procedure configures IP Multicast under AP VLAN, Management VLAN and IP core interfaces on upstream Catalyst LAN distribution-layer switch.

Procedure

1. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. **IP multicast-routing**

   Example

   ```
   Device(config)# ip multicast-routing
   ```

   Enables IP multicast processing.

3. **interface** *interface-id*

   Example

   ```
   Device(config)# interface TenGigabitEthernet 1/0
   ```

   Selects an interface that is connected to hosts and network devices on which PIM can be enabled.

4. **IP PIM sparse-mode**

   Example

   ```
   Device(config-if)# ip pim sparse-mode
   ```

   Enables IP Multicast on Layer 3 interfaces of distribution and core layer network switches:

   - **AP VLAN**- Enables IP multicast on SVI interface on VLAN assigned to wireless APs of wireless AP distribution layer switch.
   - **Management VLAN**- Enables IP multicast on SVI interface on VLAN assigned to controller management VLAN of wireless distribution layer switch.
   - **Layer 3 Interface**- Enable IP multicast routing on all core network devices and Layer 3 interfaces.

5. **exit**

Example

```
Device(config-if)# exit
```

Exits interface configuration mode.

6. **IP PIM RP-address** *RP-address*

Example

```
Device(config)# ip pim rp-address 239.254.254.100
```

Configures IP Multicast RP address on core and distribution network switches. IP network may have alternate multicast routing method.

IP multicast routing is enabled on the device with PIM sparse-mode configured on the specified interface and the RP address configured for multicast group management.

Configure local area Bonjour for wireless local mode

Provides comprehensive mDNS configuration procedures, including service policies, custom service definitions, WLAN gateway setup, service-routing, and location-based mDNS on service-peers via GUI and CLI.

This section provides configuration guidelines to implement Cisco Catalyst 9800 series controller as mDNS gateway and enable service-peer mode to enable service-routing with upstream distribution-layer Cisco Catalyst 9000 series switch in SDG-Agent mode to build Local Area Bonjour.

Configure mDNS service policy (GUI)

Configure mDNS service policy using GUI.

Configure mDNS service policy to control multicast DNS service discovery traffic on wireless networks.

The mDNS service policy consists of creating a service-list to permit built-IN or user-defined custom service-types, associate service-list to a service-policy to enforce IN ingress or egress direction and apply the service-policy to targeted Wireless Profile. This configuration is common on the controller IN service peer or single-gateway solution for wireless networks.

This procedure configures mDNS Service-Policy on a controller IN service-peer mode.

Procedure

---

1. Choose **Configuration** > **Services** > **mDNS**.
2. Set the mDNS Gateway button to **Enabled**.
3. Click **Service Policy** Tab.
4. Click **Service List** and click **Add**.

   This activates the Service List window.

5. IN the **Service List Name** field, enter a unique name with alphanumeric value.
6. From the **Direction** drop-down list, select service list policy direction. Use **IN** for ingress or **OUT** for egress mDNS message matching policy.
7. Click **+Add Services** to add mDNS service-types IN selected service list.
8. From the **Available Services** drop-down list, select built-IN or custom mDNS service-type.

9. From the **Message Type** drop-down list, select **Announcement** to accept service advertisement or **Query** to permit service discovery from the network. Default message-type is **any**.
10. Click **Save** button to add mDNS service-type entry.

> **Note**
>
> Repeat Step-7 to Step-9 to add more mDNS service-types IN selected service list.

11. Click **Apply to Device**.

    This creates a new mDNS Service List for selected direction.

> **Note**
>
> Repeat Step-5 to Step-11 for bi-directional service list.

12. Click **Service-Policy** tab.
13. Click **+Add** to create new mDNS service-policy.
14. IN the **Service Policy Name** field, enter a unique mDNS service policy name.
15. From the **Service List Input** drop-down list, select ingress mDNS service list input to enforce mDNS policies on ingress direction from wireless networks.
16. From the **Service List Output** drop-down list, select mDNS policies on egress direction to wireless networks.
17. Click **Apply to Device**.

    This creates a new mDNS service policy.

18. Choose **Configuration** > **Tags & Profiles** > **Policy**
19. Choose or create a new **Policy Profile**.
20. Click **Advanced** tab.
21. From the **mDNS Service Policy** drop-down list, select an mDNS service policy.

    Refer to Cisco Catalyst 9800 Series Configuration Guide to configure other policy profile parameters.

22. Click **Apply to Device** button.

    This creates a new policy profile or updates an existing policy profile with mDNS service policy.

23. Click **Save**.

The mDNS service policy is successfully configured and applied to the policy profile, enabling controlled multicast DNS service discovery on the wireless network.

Configure mDNS service policy using CLI

Configure mDNS service policy using CLI commands.

Configure mDNS service policy to build and apply service-policies on target wireless profile in service-peer mode.

This procedure builds and applies service-policies on target wireless profile in service-peer mode.

Procedure

1. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. **mdns-sd service-list** *service-list-name* **in out**

   Example

   ```
   Device(config)# mdns-sd service-list VLAN100-LIST-IN in
   Device(config)# mdns-sd service-list VLAN100-LIST-OUT out
   ```

   Configures mDNS service-list to classify one or more service-types. Unique service-list is required to process incoming mDNS message and outbound response to requesting end points.

3. **match service-definition-name message-type any announcement query**

   Example

   ```
   Device(config)# mdns-sd service-list VLAN100-LIST-IN in
   Device(config-mdns-sl-in)# match APPLE-TV
   Device(config-mdns-sl-in)# match PRINTER-IPPS message-type announcement
   ```

   Matches inbound service-list. The controller validates to accept or drop incoming mDNS service-type (for example, Apple TV) advertisement or query matching message type. The service-list contains implicit deny at the end. Default message-type is “any”.

4. **match service-definition-name message-type any announcement query**

   Example

   ```
   Device(config)# mdns-sd service-list VLAN100-LIST-OUT out
   Device(config-mdns-sl-in)# match APPLE-TV
   Device(config-mdns-sl-in)# match PRINTER-IPPS
   ```

   Matches an outbound service-list. The controller provides local service proxy function by responding matching service-type to the requesting end points. For example, the Apple-TV and Printer learnt from VLAN 100 will be distributed to receiver in same VLAN 100. The service-list contains implicit deny at the end.

   The message-type for outbound service-list is not required.

5. **exit**

   Example

   ```
   Device(config-mdns-sl-in)# exit
   ```

   Returns to global configuration mode.

6. **mdns-sd service-policy** *service-policy-name*

Example

```
Device(config)# mdns-sd service-policy VLAN100-POLICY
```

Creates a unique mDNS service-policy.

7. **service-list** *service-list-name* **in out**

Example

```
Device(config-mdns-ser-policy)# service-list VLAN100-LIST-IN in
Device(config-mdns-ser-policy)# service-list VLAN100-LIST-OUT out
```

Configure mDNS service-policy to associate service-list for each direction.

8. **exit**

Example

```
Device(config-mdns-ser-policy)# exit
```

Exits mDNS service policy configuration mode.

9. **wireless profile policy** *policy-name*

Example

```
Device(config)# wireless profile policy WLAN-PROFILE
```

Configures unique wireless profile policy name to associate mDNS service-policy.

10. **mdns-sd service-policy** *service-policy*

Example

```
Device(config-wireless-policy)# mdns-sd service-policy VLAN100-POLICY
```

Associates mDNS service-policy to configured VLAN IDs.

> **Note**
>
> This step requires wireless profile policy to be administratively shutdown prior association service-policy and re-activate with no shutdown to make service-policy effective.

11. **exit**

Example

```
Device(config-mdns-sd)# exit
```

Exits mDNS gateway configuration mode.

The mDNS service policy is configured and associated with the wireless profile policy. The controller can now process incoming mDNS messages and provide outbound responses based on the configured service lists and policy.

Configuring custom service definition (GUI)

Configure custom service definition using GUI.

Create custom mDNS service definition with matching mDNS PTR records to enable end mDNS service-routing in the network.

The Cisco IOS-XE supports various built-in well-known mDNS service-definition types mapping to key mDNS PTR records to user-friendly names. For example, built-in Apple-TV service-type is associated with _airplay. _tcp.local and _raop. _tcp.local PTR records to successfully enable service in the network. The network administrator can create custom service-definition with matching mDNS PTR records to enable end mDNS service-routing in the network.

This procedure configures custom mDNS service definition and applies it to policy.

Procedure

1. Choose **Configuration** > **Services** > **mDNS**.
2. Set the mDNS Gateway button to **Enabled**.
3. Click **Service Policy** Tab.
4. Click **Add** to create new custom mDNS service-list definition.

   This activates Service Definition window.
5. In the **Service Definition Name** filed, enter a unique aplhanumeric value.
6. (Optional) In the **Description** field, enter a description for the service definition.
7. In the **Service Type** field, enter single mDNS PoinTeR (PTR) record entry in *_<service-type>. _<protocol>.local* regular expression format. For example, _airplay. _tcp.local
8. Click **+** to add custom mDNS service-type in selected definition list.

   Note

   Repeat Steps 7 and Step 8 to add more custom service-type in selected definition list.
9. Click **Apply**.
10. Perform steps give in *Configuring mDNS Service Policy (GUI)* by selecting built-in or custom service-type to configure service list.
11. Click **Save**.

Custom mDNS service definition is configured and applied to the service policy.

Configure custom service definition (CLI)

Configure custom service definition using commands to discover mDNS services from local wireless networks.

This task creates custom service-definition configuration to discover mDNS services from local wireless networks.

Use custom service definitions when you need to configure mDNS service discovery for services that are not predefined in the system.

Procedure

1. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. **mdns-sd service-definition-name** *service-definition-name*

   Example

   ```
   Device(config)# mdns-sd service-definition APPLE-CLASSROOM
   ```

   Creates unique service-definition name for custom service-types.

3. **service-type** *custom-mDNS-PTR*

   Example

   ```
   Device(config-mdns-ser-def)# service-type _classroom._tcp.local
   ```

   Configure an regular-expression string for custom mDNS PoinTeR(PTR) record.

4. **exit**

   Example

   ```
   Device(config-mdns-ser-def)# exit
   ```

   Returns to global configuration mode.

The custom service definition is created and configured for mDNS service discovery on local wireless networks.

Configure mDNS gateway on WLAN (GUI)

Configure mDNS gateway on WLAN using GUI.

Configure mDNS gateway to activate mDNS processing for wireless clients associated with a WLAN.

The mDNS gateway activation on targeted WLAN is required to start processing incoming mDNS messages from associated wireless clients. To activate mDNS gateway the WLAN must be administratively shutdown and re-enable thus it may require network downtime planning.

This procedure configures custom mDNS gateway and required policies.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click to enable the mDNS Gateway on existing **WLAN** row of Catalyst 9800 controller. Click **+ Add** button to create new WLAN if required. Refer to Catalyst 9800 Series Wireless Controller Configuration Guide for step-by-step WLAN configuration.

3. Click **Advanced** tab.
4. From the **mDNS Mode** drop-down list, select **Gateway** to activate mDNS Gateway on the selected WLAN.
5. Click **Apply to Device**.
6. Click **Save**.

The mDNS gateway is configured and activated on the WLAN, enabling mDNS message processing for associated wireless clients.

Configure mDNS gateway on WLAN (CLI)

Configure mDNS gateway on WLAN using commands.

This task implements mDNS gateway functionality on a targeted WLAN to enable multicast DNS service discovery across network segments.

This procedure implements mDNS gateway on a targeted WLAN of the controller in service-peer mode.

Procedure

1. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. **WLAN** *profile-name WLAN-ID SSID-name*

   Example

   ```
   Device(config)# wlan WLAN-PROFILE 1 blizzard
   ```

   Creates a unique WLAN.

3. **mdns-sd-interface gateway**

   Example

   ```
   Device(config-wlan)# mdns-sd-interface gateway
   ```

   Configure mDNS gateway on targeted WLAN.

   > **Note**
   >
   > This step requires wireless profile policy to be administratively shutdown prior association service-policy and re-activate with no shutdown to make service-policy effective.

4. **exit**

Example

```
Device(config-wlan)# exit
```

Returns to global configuration mode.

The mDNS gateway is configured on the WLAN and will enable multicast DNS service discovery functionality once the wireless profile policy is properly activated.

Configure service-routing on service-peer

Configure service-routing on service-peer to extend mDNS service discovery and distribution boundary beyond single controller to global IP network.

Configure the controller in service peer mode to extend mDNS service discovery and distribution boundary beyond single controller to global IP network using unicast based service-routing.

The controller deployed in Service-Peer mode extends mDNS service discovery and distribution boundary beyond single controller to global IP network using on unicast based service-routing. The controller service peer must establish IP based unicast service-routing with Cisco Catalyst 9000 series switch in distribution layer network for global service-routing.

This procedure configures the controller in service peer mode.

Procedure

---

1. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. **mdns-sd gateway**

   Example

   ```
   Device(config)# mdns-sd gateway
   ```

   Enables mDNS and enters in mDNS gateway configuration mode. The following optional parameters are available:

   - **active-query**: Periodic mDNS query to refresh dynamic cache.
   - **active-response**: Periodic active mDNS response instead per request processing.
   - **mode**: Set Catalyst 9800 in service-peer mode.
   - **SDG-agent**: Unicast service-routing with targeted SDG-Agent.
   - **service-announcement-count**: Configures maximum advertisements in service-routing to SDG-Agent.
   - **service-announcement-timer**: Configures advertisements announce timer periodicity in service-routing to SDG-Agent.
   - **service-query-count**: Configures maximum queries in service-routing to SDG-Agent.
   - **service-query-timer**: Configures query forward timer periodicity in service-routing to SDG-Agent.
   - **service-type-enumeration**: Configures service enumeration.

- **source-interface**: Configures the source interface. If the source interface is configured, it will be used for all mDNS transactions. By default, wireless management interface will be used.
- **transport**: Use IPv4 (default) or IPv6 transport for mDNS messaging to end points.

> **Note**
>
> For **rate-limit**, **service-announcement-count**, **service-announcement-timer**, **service-query-count** and **service-query-timer** commands, you can retain the default value of the respective parameter for general deployments. Configure a different value, if required, for a specific deployment.

3. **mode service-peer**

   Example

   ```
   Device(config-mdns-sd)# mode service-peer
   ```

   Configure mDNS gateway in service-peer mode.

4. **SDG-agent IPv4 Address**

   Example

   ```
   Device(config-mdns-sd)# sdg-agent 10.0.2.254
   ```

   Configure SDG Agent IPv4 address. Typically, the management VLAN gateway address. If FHRP mode, then use FHRP Virtual-IP address of management VLAN.

5. **exit**

   Example

   ```
   Device(config-mdns-sd)# exit
   ```

   Returns to global configuration mode.

The controller is configured in service peer mode with unicast based service-routing to extend mDNS service discovery beyond the single controller to the global IP network.

Configure location-based mDNS on service-peer (GUI)

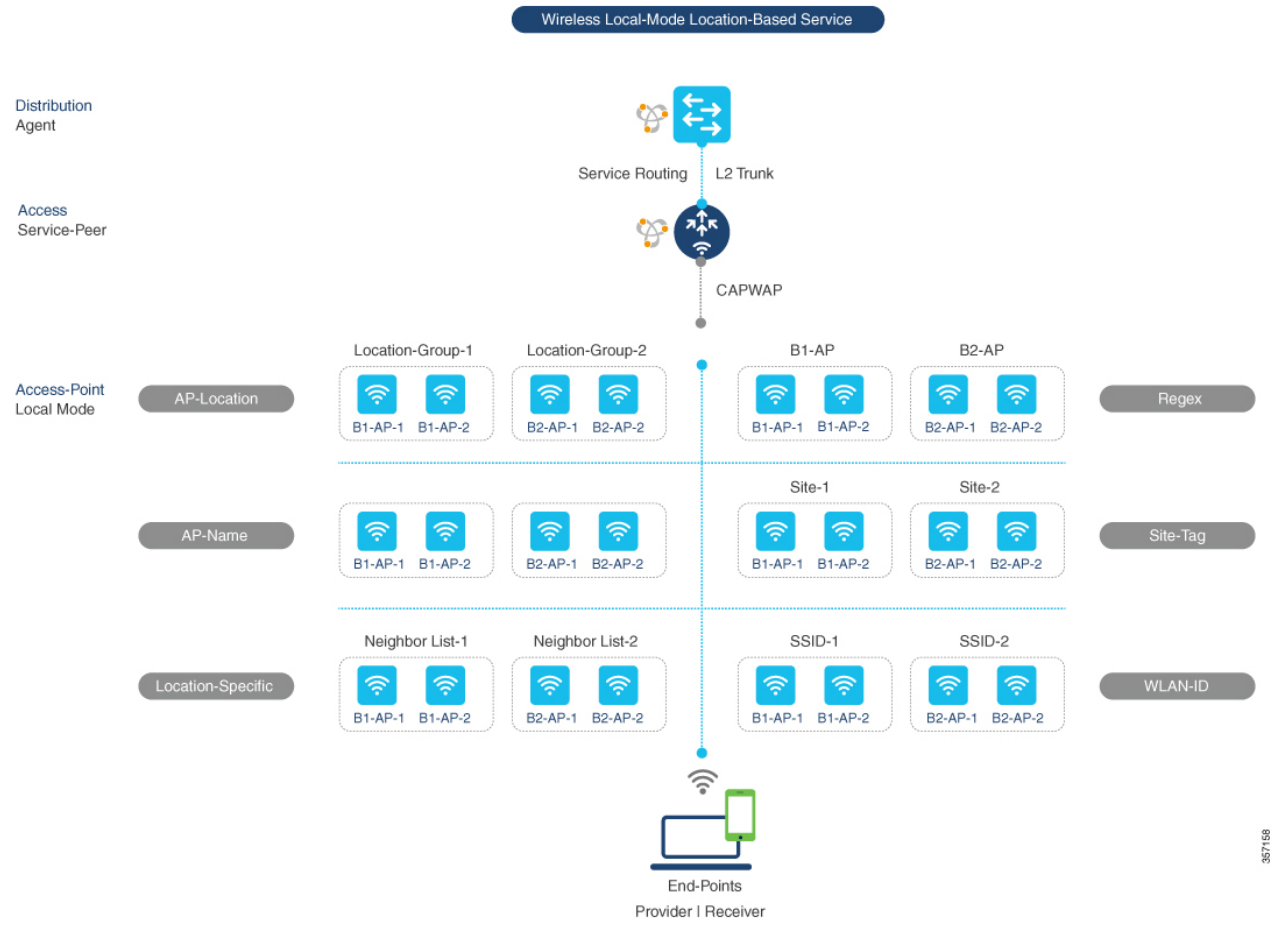
Configure location-based mDNS on service-peer using GUI.

This task configures location-based mDNS service policy to enable service discovery and distribution between wireless service provider and receiver endpoints using policy-based service distributions IN wireless networks.

Cisco Catalyst 9800 series controller supports location-based mDNS service discovery and distribution between wireless service provider and receiver endpoints. The location-based mDNS service support can be implemented using multiple supporting AP classification methods to implement policy-based service distributions IN wireless networks. The location-based mDNS service is effective and supported on wireless APs IN Local-Mode or FlexConnect Central Switching modes.

The figure given below illustrates various LSS based mDNS service mode discovery and distribution support:

Figure 59: Location-Based mDNS gateway



Before you begin

Follow these steps to configure location-based mDNS service policy:

Procedure

1. Choose **Configuration** > **Services** > **mDNS**.
2. Set the mDNS Gateway button to **Enabled**.
3. Click **Service Policy** Tab.
4. Click **Service List** and click **Add**.

   This activates the Service List window.
5. IN the **Service List Name** field, enter a unique name with alphanumeric value.
6. From the **Direction** drop-down list, select service list policy direction. Use **IN** for ingress or **OUT** for egress mDNS message matching policy.
7. Click **+Add Services** to add mDNS service-types IN selected service list.
8. From the **Available Services** drop-down list, select built-IN or custom mDNS service-type.
9. From the **Message Type** drop-down list, select **Announcement** to accept service advertisement or **Query** to permit service discovery from the network. Default message-type is **any**.
10. Click **Save** button to add mDNS service-type entry.

> **Note**
>
> Repeat Step-7 to Step-9 to add more mDNS service-types IN selected service list.

11. Click **Apply to Device**.

    This creates a new mDNS Service List for selected direction.

> **Note**
>
> Repeat Step-5 to Step-11 for bi-directional service list.

12. Click **Service-Policy** tab.
13. Click **+Add** to create new mDNS service-policy.
14. IN the **Service Policy Name** field, enter a unique mDNS service policy name.
15. From the **Service List Input** drop-down list, select ingress mDNS service list input to enforce mDNS policies on ingress direction from wireless networks.
16. From the **Service List Output** drop-down list, select mDNS policies on egress direction to wireless networks.
17. Click **Apply to Device**.

    This creates a new mDNS service policy.

18. Choose **Configuration** > **Tags & Profiles** > **Policy**
19. Choose or create a new **Policy Profile**.
20. Click **Advanced** tab.
21. From the **mDNS Service Policy** drop-down list, select an mDNS service policy.

    Refer to Cisco Catalyst 9800 Series Configuration Guide to configure other policy profile parameters.

22. Click **Apply to Device** button.

    This creates a new policy profile or updates an existing policy profile with mDNS service policy.

23. Click **Save**.

The location-based mDNS service policy is configured and applied to the policy profile, enabling policy-based mDNS service discovery and distribution IN the wireless network.

Configure location-based mDNS on service-peer (CLI)

Configure location-based mDNS on service-peer using commands.

This procedure implements LSS based mDNS service discovery and distribution between wireless endpoints on the targeted WLAN of the controller in service-peer mode.

Location-based mDNS filtering allows for targeted service discovery within specific network segments or geographic locations, improving network efficiency and security by limiting service advertisements to relevant areas.

Procedure

1. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

2. **mdns-sd service-policy** *service-policy-name*

   Example

   ```
   Device(config)# mdns-sd service-policy VLAN100-POLICY
   ```

   Creates a unique mDNS service-policy.

3. **location AP-location AP-name LSS regex site-tag SSID**

   Example

   ```
   Device(config-mdns-ser-policy)# location ap-location
   ```

   Creates a unique mDNS service-policy.

   - **AP-location**: Enables mDNS service discovery and distribution between wireless service provider and receiver connected to one or more AP configured in the same location name. The mDNS services from non-matching AP location is automatically filtered.
   - **AP-name**: Enables mDNS service discovery and distribution between wireless service provider and receiver connected to single AP matching same AP name. The mDNS services from non-matching AP name is automatically filtered.
   - **LSS**: Enables mDNS service discovery and distribution between wireless service provider and receiver connected to same and neighboring one or more AP based on RRM. The mDNS services from non-matching AP neighbor-list is automatically filtered.
   - **regex**: Enables mDNS service discovery and distribution between wireless service provider and receiver connected to one or more AP configured within matching AP name or AP Location name using regular-expression string. The mDNS services from non-matching AP names is automatically filtered.
   - **site-tag**: Enables mDNS service discovery and distribution between wireless service provider and receiver connected to one or more AP configured same site tag name. The mDNS services from non-matching site tag is automatically filtered.
   - **SSID**: Enables mDNS service discovery and distribution between wireless service provider and receiver connected to one or more AP configured same SSID name. The mDNS services from non-matching SSID is automatically filtered.

4. **exit**

   Example

   ```
   Device(config-mdns-ser-policy)# exit
   ```

Exits mDNS service policy configuration mode.

The location-based mDNS service policy is configured and will filter mDNS services based on the specified location criteria.

Verify mDNS gateway configuration

Lists verification procedures and reference commands to confirm proper mDNS gateway configuration and operational status.

This section provides guidelines to verify various Local Area Bonjour domain mDNS service configuration parameters, cache records, statistics and more on the controller in service peer mode.

Command or Action	Purpose
show mdns-sd cache AP-MAC client-MAC detail glan-ID mdns-AP rlan-ID statistics type UDN wired wlan-ID	Displays available mDNS cache records supporting multiple variables providing granular source details: • **AP-MAC**: Displays one or more mDNS service instance cache records discovered from provided AP MAC address. • **client-MAC**: Displays one or more mDNS service instance(s) cache records discovered from service provider wireless client MAC address. • **detail**: Displays mDNS record detail information combined with client and network attributes and other service parameters. • **glan-ID**: Displays one or more mDNS service instance(s) cache records discovered from provided Wired Guest LAN ID MAC address. • **mdns-AP**: Displays one or more mDNS service instance(s) cache records discovered from provided Wireless mDNS AP MAC address. • **rlan-ID**: Displays one or more mDNS service instances(s) cache records discovered from provided Wired Remote LAN ID. Range 1-128. • **statistics**: Displays detail global bi-directional mDNS statistics for IPv4 and IPv6 transports with packet processing count for each mDNS record-type. • **type**: Displays one or more service-instance(s) cache records matching mDNS record-type, i.e., A-AAAA, PTR, SRV and TXT. • **UDN**: Displays one or more mDNS service instance(s) cache records discovered from segmented Wireless service provider in User-Defined-Group (UDN) or shared-services. • **wired**: Displays one or more mDNS service instance(s) cache records discovered from upstream Layer 2 wired network. • **wlan-ID**: Displays one or more mDNS service instance(s) cache records discovered from matching provided wlan-ID. Range 1-4096.
show mdns-sd statistics debug flexconnect glan-ID rlan-ID wired wlan-ID	Displays detailed mdns statistics processed bi-directionally by system on each mDNS gateway enabled VLAN configured mDNS in Unicast mode. The expanded keyword of mDNS statistics can provide detail view on interface, policy, service-list and services.
show mdns-sd summary	Displays brief information about mDNS gateway and key configuration status on all VLANs and interfaces of the system.

Verify Catalyst WLC service-peer configuration

This section provides guidelines to verify service peer service configuration and statistics.

Command or Action	Purpose
show mdns-sd sp-SDG statistics	Displays mDNS service-routing statistics between Catalyst 9800 service-peer and upstream SDG Agent switch for global service discovery and distribution.
show mdns-sd summary	Displays brief information about mDNS gateway and key configuration status and parameters of the system.

Reference

Reference for mDNS gateway.

Table 202:

Related Topic	Document Title
DNA Service for Bonjour Deployment on Cisco Catalyst 9600 Switch	Cisco Catalyst 9600 Series Switch Software Configuration Guide, Release 17.5.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9500 Switch	Cisco Catalyst 9500 Series Switch Software Configuration Guide, Release 17.5.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9400 Switch	Cisco Catalyst 9400 Series Switch Software Configuration Guide, Release 17.5.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9300 Switch	Cisco Catalyst 9300 Series Switch Software Configuration Guide, Release 17.5.X
Cisco Wide Area Bonjour Application on Cisco Catalyst Center User Guide	Cisco Wide Area Bonjour Application on Cisco Catalyst Center User Guide, Release 2.2.2

232 Configuring Local Area Bonjour for Wireless FlexConnect Mode

Topics:

- Local area Bonjour for wireless FlexConnect mode
- Local area Bonjour for wireless FlexConnect mode

Guides through the setup and configuration of Local Area Bonjour for Wireless FlexConnect Mode, including prerequisites, multicast Domain Name System gateway alternatives, and verification procedures.

Local area Bonjour for wireless FlexConnect mode

Introduces core concepts of Local Area Bonjour in Wireless FlexConnect environments, explains restrictions and prerequisites, and compares mDNS gateway alternatives to facilitate effective service discovery and network design understanding.

Local area Bonjour for wireless FlexConnect mode is a network service feature that

- introduces unicast mode function in Local Area Bonjour network domain
- enables enhanced gateway function at the first hop of Wired and Wireless networks to communicate directly with any industry standard RFC 6762 compliant Multicast DNS (mDNS) end point in Layer 2 Unicast mode, and
- provides service-peer mode expanding single-gateway to end-to-end service-routing with upstream SDG-Agent switch to enable unicast-mode, increased scale, performance and resiliency in the network.

Restrictions for local area Bonjour for wireless FlexConnect mode

Outlines restrictions for local area Bonjour for wireless FlexConnect mode.

In FlexConnect mode network deployments, the mDNS gateway and service-peer mode on the controller must not be configured and must be in disabled state.

Prerequisites for local area Bonjour for wireless FlexConnect mode

Outlines prerequisites for local area Bonjour for wireless FlexConnect mode.

The Cisco Catalyst 9800 series controller must be successfully configured and operational before implementing Cisco Local Area Bonjour for FlexConnect mode wireless networks.

Here are the prerequisites to enable successful mDNS gateway solution for Wireless FlexConnect:

- Ensure that the targeted Layer 2 Catalyst 9000 Series Ethernet switch is configured in service-peer role and running the required Cisco IOS-XE software version.
- Ensure that the Catalyst 9000 Series Ethernet switch runs a valid Cisco DNA-Advantage license.
- Ensure that the upstream distribution-layer Cisco Catalyst switch for Wired and FlexConnect Local Switching Wireless networks is configured in SDG-Agent mode and runs a valid Cisco DNA-Advantage license.

mDNS gateway alternatives for wireless FlexConnect mode

Explains mDNS gateway alternatives for wireless FlexConnect mode.

mDNS gateway alternatives for wireless FlexConnect mode are implementation methods that

- address evolving business and technical requirements in Enterprise networks
- enable service discovery and distribution in FlexConnect Local Switching based wireless networks, and
- provide two distinct approaches based on the operating network environment.

Implementation methods

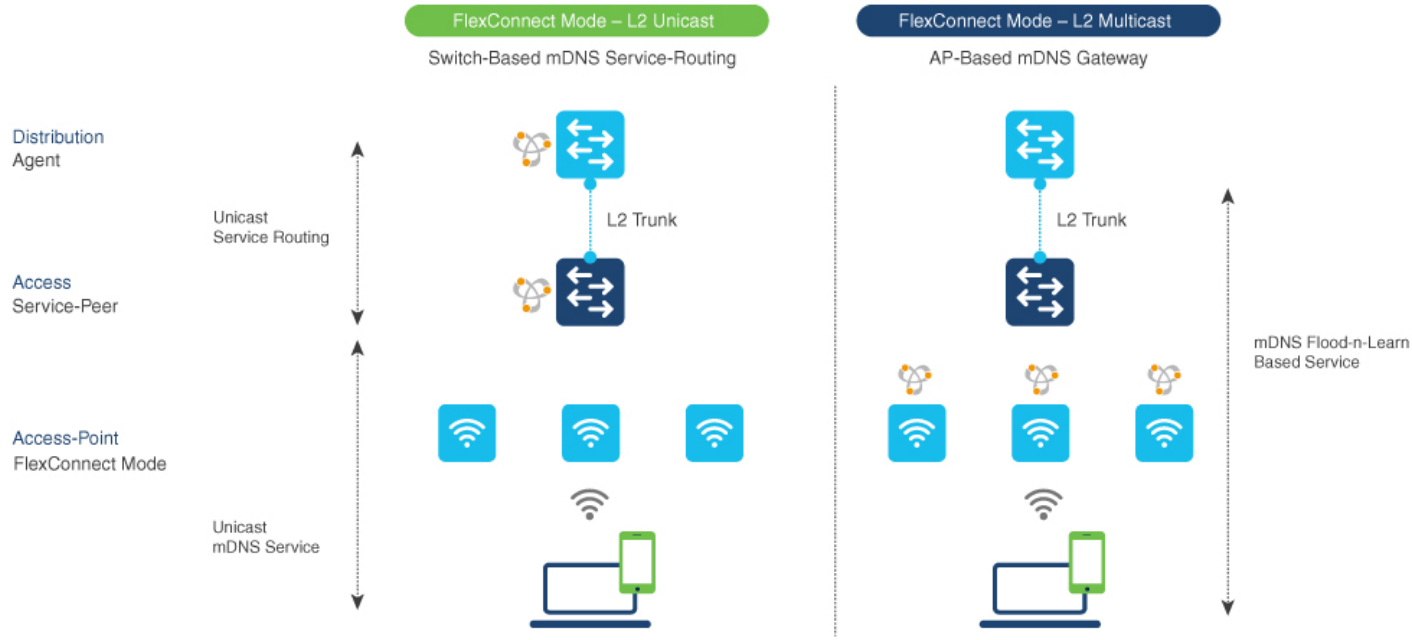
Based on the operating network environment, the mDNS gateway for FlexConnect mode wireless network can be implemented in one of the methods to address service discovery and distribution:

- **Switch Based mDNS Gateway**–In Layer 2 access, the Cisco Catalyst 9000 series Ethernet switch must be implemented as mDNS gateway in Service-Peer role. The key benefits are:

- Replaces flood-n-learn with the new enhanced Unicast-based mDNS communication with FlexConnect mode wireless users.
- Eliminates mDNS flood with Unicast service-routing to LAN distribution. The Unicast service-routing between LAN distribution and Layer 2 access layer switches forms Local Area Bonjour domain to enable policy and location-based service discovery and distribution. The Unicast based service-routing over Layer 2 trunk eliminates mDNS flood-free and enables service-oriented wireless networks.
- Eliminates the requirement to forward wired network traffic to wireless Access Points improving wireless scale, performance, and network reliability.

- **AP Based mDNS Gateway**–The Cisco FlexConnect mode wireless access points can alternatively be implemented as mDNS gateway when connected to unsupported LAN access switch. In this method, the mDNS service discovery and distribution follows flood-n-learn mechanism over the Layer 2 wireless network. To implement AP based mDNS gateway, see the Multicast Domain Name System chapter.

Figure 60: mDNS gateway alternatives for FlexConnect mode



Local area Bonjour for wireless FlexConnect mode

Explains local area Bonjour for wireless FlexConnect mode.

Local area Bonjour for wireless FlexConnect mode is a mDNS gateway function that

- supports various advancements for broad range of wireless networks with distributed zero-configuration services
- enables distributed Bonjour gateway function at network edge through common unified Cisco IOS-XE operating system across Cisco Catalyst 9000 series LAN switches and Cisco Catalyst 9800 series controller
- provides end-to-end Wide Area Bonjour service-routing for service-oriented enterprise networks with intuitive user-experience.

mDNS gateway modes

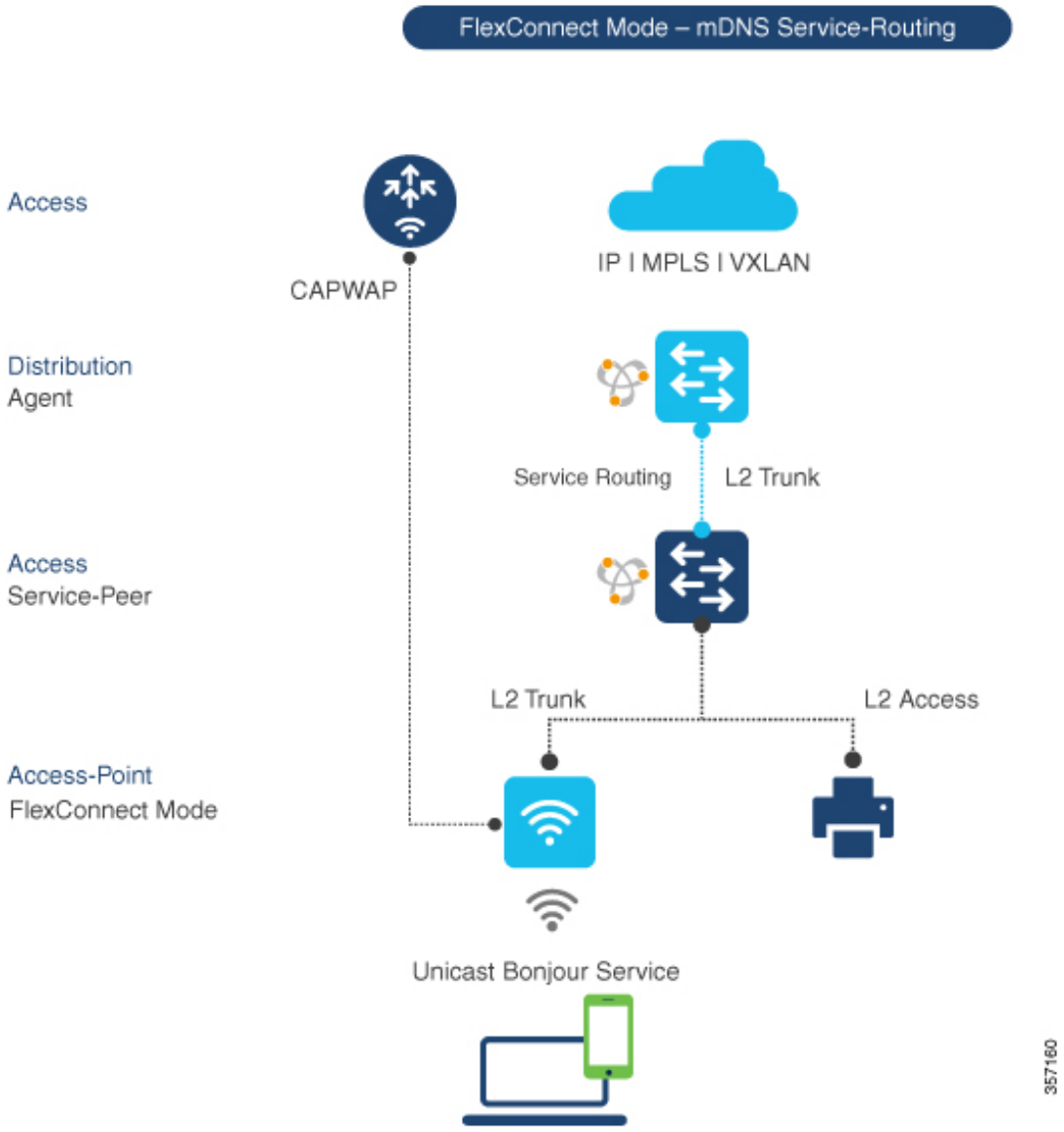
The Cisco Catalyst 9000 series switches in the Layer 2 access layer and Layer 3 distribution layer must be configured in specific mDNS gateway modes to enable Unicast-based mDNS service-routing between wired and FlexConnect Local Switching mode wireless users within the same Layer 2 network block:

- **Service-Peer** - The Layer 2 access switch connecting wireless access point in FlexConnect Local Switching mode must be configured with mDNS gateway in Service-Peer mode. Each Layer 2 access switch provides mDNS gateway function between locally attached wired and FlexConnect mode wireless users. The Unicast-based mDNS service discovery and distribution within same or different VLANs is supported with bi-directional mDNS policies on single Layer 2 access switch.
- **SDG Agent** - The mDNS flood-n-learn based method in Layer 2 network is replaced with simple Unicast based service-routing between Layer 2 access switch in Service-Peer mode and upstream distribution-layer in mDNS gateway SDG Agent mode. The Unicast based mDNS service-routing eliminates mDNS flood over Layer 2 trunk ports providing increased bandwidth, enhanced security, location-based services, and flood control management in wired and FlexConnect wireless network.

Controller mDNS gateway function architecture

This figure illustrates how the controller connected to wireless access points support mDNS gateway function to wireless users in FlexConnect Local Switching mode.

Figure 61: Cisco Catalyst 9800 series controller local area Bonjour for wireless - FlexConnect mode



Local area Bonjour for wireless FlexConnect mode

Details the end-to-end configuration of Local Area Bonjour in Wireless FlexConnect mode, providing step-by-step CLI-based procedures for gateway modes, service policies, location filtering, custom service definitions, service routing, wired integration, and verification methods.

Local area Bonjour for wireless FlexConnect mode is a network configuration that enables service discovery and routing across distributed wireless infrastructure by implementing Cisco Catalyst 9000 series Ethernet switches as mDNS gateways with service-peer and SDG Agent mode capabilities to build service-routing connections with upstream distribution-layer switches.

Configuration guidelines

This configuration enables service-peer and SDG Agent mode to enable service-routing with upstream distribution-layer Cisco Catalyst 9000 series switch in SDG Agent mode to build Local Area Bonjour.

Configure mDNS gateway mode (CLI)

Configure mDNS gateway mode using CLI commands to enable mDNS gateway and Service-Peer mode on Layer 2 access switch and SDG Agent mode on Layer 3 distribution layer switch.

This task enables mDNS gateway functionality on network switches to facilitate multicast DNS service discovery across network segments.

Use this procedure when you need to enable mDNS gateway and Service-Peer mode on Layer 2 access switch and SDG Agent mode on Layer 3 distribution layer switch.

Procedure

1. **enable**

   Example

   ```
   Device# enable
   ```

   Enables privileged EXEC mode.

   Enter your password, if prompted.

2. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

3. **mdns-sd gateway**

   Example

   ```
   Device(config)# mdns-sd gateway
   ```

   Enables mDNS on the Layer 2 Catalyst switch and enters the mDNS gateway configuration mode.

   (Optional) You can configure additional parameters:

   - **air-print-helper**: Enables communication between Apple iOS devices like iPhone or iPad to discover and use older printers that does not support driverless AirPrint function.
   - **cache-memory-max**: Configures the percentage memory for cache.
   - **ingress-client**: Configures Ingress client packet tuners.
   - **rate-limit**: Enables rate limiting of incoming mDNS packets.
   - **service-announcement-count**: Configures maximum advertisements.
   - **service-announcement-timer**: Configures advertisements announcement timer periodicity.
   - **service-query-count**: Configures maximum queries.
   - **service-query-timer**: Configures query forward timer periodicity.
   - **service-type-enumeration**: Configures service enumeration.

> **Note**
>
> For **cache-memory-max**, **ingress-client**, **rate-limit**, **service-announcement-count**, **service-announcement-timer**, **service-query-count**, **service-query-timer**, and **service-type-enumeration** commands, you can retain the default value of the respective parameter for general deployments. Configure a different value, if required, for a specific deployment.

4. **mode {service-peer | SDG-agent}**

   Example

   ```
   Device(config-mdns-sd)# mode service-peer
   ```

   ```
   Device(config-mdns-sd)# mode sdg-agent
   ```

   Configure mDNS gateway in one of these modes based on the system settings:

   - **service-peer**- Enables Layer 2 Catalyst access switch in mDNS Service-Peer mode.
   - **SDG-agent**- Default. Enables Layer 3 distribution layer Catalyst switch in SDG Agent mode to peer with central Cisco Catalyst Center controller for Wide Area Bonjour service routing.

5. **exit**

   Example

   ```
   Device(config-mdns-sd)# exit
   ```

   Exits mDNS gateway configuration mode.

The mDNS gateway mode is configured on the switch. The switch can now facilitate mDNS service discovery across network segments based on the configured mode.

Configure mDNS service policy (CLI)

Configure mDNS service policy using commands to enable multicast DNS service discovery and proxy functionality on Catalyst switches.

Configure mDNS service policy to enable multicast DNS service discovery and proxy functionality on Catalyst switches, allowing service advertisement and discovery across VLANs in both Service-Peer and SDG agent modes.

You need to perform this configuration to enable mDNS service policy on Catalyst switches. This configuration is required in Service-Peer mode for Layer 2 Catalyst switches and SDG agent mode for Layer 3 Catalyst switches.

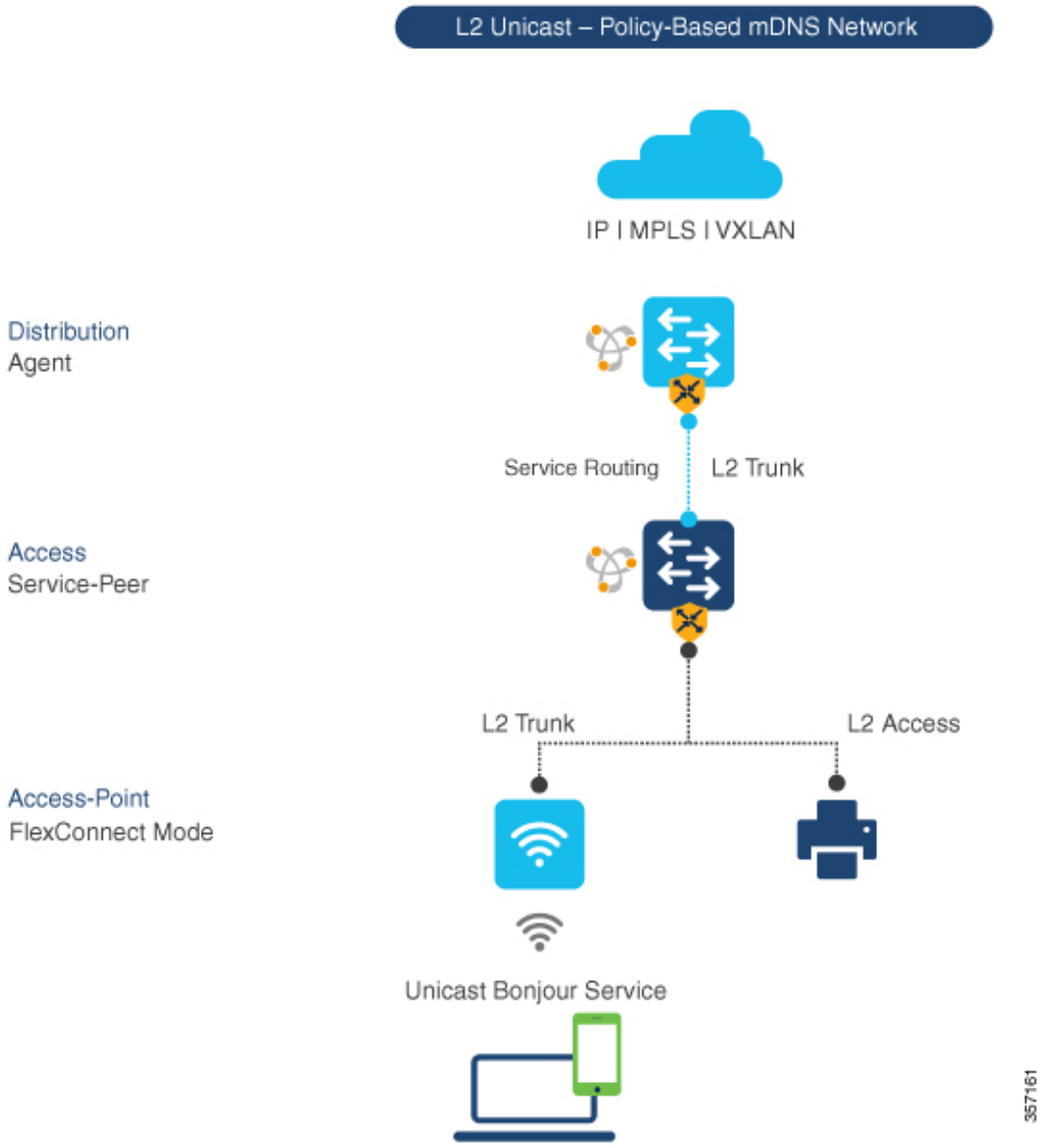
You need to perform the following to configure an mDNS service policy:

1. Create service-list to permit built-in or user-defined custom service types.
2. Associate service-list to a service-policy to enforce ingress or egress direction.
3. Apply the service policy to the new VLAN configuration mode.

> **Note**
>
> You will need this configuration in Service-Peer mode for Layer 2 Catalyst switch and SDG agent mode for Layer 3 Catalyst switch.

The figure shows how to configure mDNS policies on Catalyst switch in Service-Peer and SDG agent modes.

Figure 62: mDNS service policy configuration on Catalyst switch in Service-Peer and SDG agent modes



This procedure builds and applies service-policies on target VLAN in service-peer and SDG agent modes.

Before you begin

Follow these steps to configure mDNS service policy using CLI commands:

Procedure

1. **enable**

   Example

   ```
   Device# enable
   ```

   Enables privileged EXEC mode.

   Enter your password, if prompted.

2. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

3. **mdns-sd service-list** *service-list-name* {**in** | **out**}

   Example

   ```
   Device(config)# mdns-sd service-list VLAN100-LIST-IN in
   ```

   ```
   Device(config)# mdns-sd service-list VLAN100-LIST-OUT out
   ```

   Configure mDNS service-list to classify one or more service types. Unique service-list is required to process incoming mDNS message and outbound response to request locally connected wired or FlexConnect wireless end points.

4. **match** *service-definition-name* [**message-type** {**any** | **announcement** | **query**}]

   Example

   ```
   Device(config)# mdns-sd service-list VLAN100-LIST-IN in
   ```

   ```
   Device(config-mdns-sl-in)# match APPLE-TV
   ```

   ```
   Device(config-mdns-sl-in)# match PRINTER-IPPS message-type announcement
   ```

   Matches inbound service-list.

   The Catalyst switch validates to accept or drop incoming mDNS service-type (such as, Apple TV) advertisement or query matching message type from locally connected wired or FlexConnect wireless end points. The service-list contains implicit deny at the end.

   The default message-type used is **any**.

5. **match** *service-definition-name* [**message-type** {**any** | **announcement** | **query**}]

Example

```
Device(config)# mdns-sd service-list VLAN100-LIST-OUT out
```

```
Device(config-mdns-sl-in)# match APPLE-TV
```

```
Device(config-mdns-sl-in)# match PRINTER-IPPS
```

Matches outbound service-list.

The Catalyst switch provides local service proxy function by responding matching service-type to the requesting end point(s). For example, the Apple-TV and Printer learnt from VLAN 100 will be distributed to FlexConnect wireless receiver in same VLAN 100. The service-list contains implicit deny at the end.

The message-type for outbound service-list is not required.

6. **mdns-sd service-policy** *service-policy-name*

Example

```
Device(config)# mdns-sd service-policy VLAN100-POLICY
```

Creates unique mDNS service-policy in global configuration mode.

7. **service-list** *service-list-name* **{in | out}**

Example

```
Device(config)# mdns-sd service-policy VLAN100-POLICY
```

```
Device(config-mdns-ser-policy)# service-list VLAN100-LIST-IN in
```

```
Device(config-mdns-ser-policy)# service-list VLAN100-LIST-OUT out
```

Configures mDNS service-policy to associate service-list for each direction.

8. **VLAN configuration** *ID*

Example

```
Device(config)# vlan configuration 100
```

Enables wired or wireless FlexConnect user VLAN configuration for advanced service parameters. One or more VLANs can be created for the same settings.

Here, *ID* refers to the VLAN configuration ID. The range is from 101 to 110 and 200. This range allows to configure consecutive and non-consecutive VLAN ID(s).

9. **mdns-sd gateway**

Example

```
Device(config-vlan)# mdns-sd gateway
```

Enables mDNS gateway on configured wired or FlexConnect wireless user VLAN ID(s).

10. **service-policy** *service-policy-name*

    Example

    ```
    Device(config-vlan-mdns)# service-policy VLAN100-POLICY
    ```

    Associates mDNS service-policy to the configured wired or FlexConnect wireless user VLAN ID(s).

11. **exit**

    Example

    ```
    Device(config-vlan-mdns)# exit
    ```

    Exits mDNS gateway configuration mode.

The mDNS service policy is successfully configured and applied to the specified VLAN. The Catalyst switch can now process mDNS service advertisements and queries according to the configured service lists and policy rules.

Configure mDNS Location-Filter (CLI)

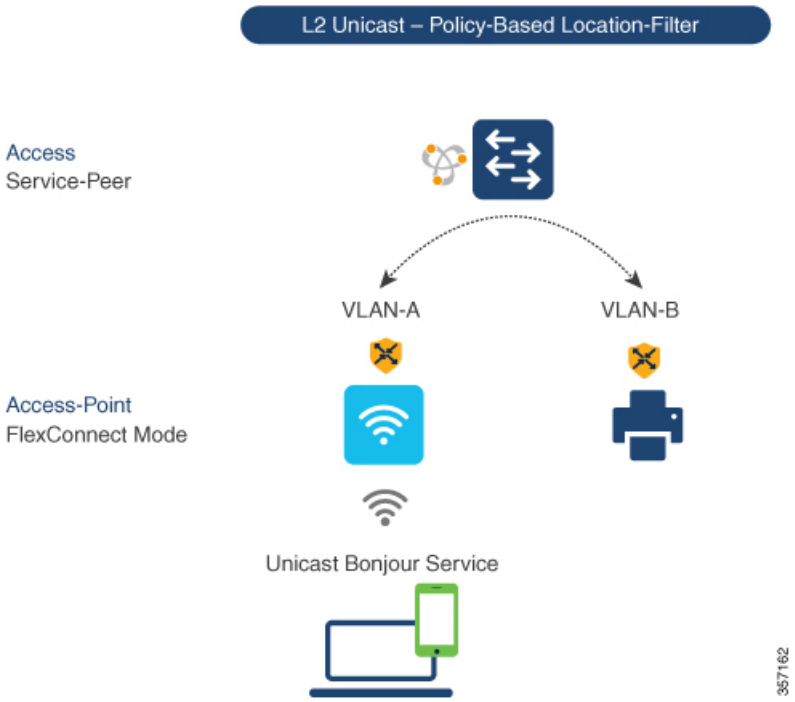
Configure mDNS location-filter using commands to allow service discovery and distribution between locally configured VLAN IDs associated to FlexConnect wireless user networks.

Configure mDNS location-filter to enable local service proxy on Cisco Catalyst switch in Service-Peer mode and discover mDNS services between local wired and wireless FlexConnect user VLANs.

Optionally, you can configure mDNS location-filter to allow service discovery and distribution between locally configured VLAN IDs associated to FlexConnect wireless user networks.

The following figure illustrates and references location-filter policy on Catalyst switch in Service-Peer mode permitting to discover and distribute mDNS services between wired and FlexConnect wireless user VLANs.

Figure 63: Catalyst Service-Peer mDNS Location-Filter configuration



Before you begin

Follow these steps to enable local service proxy on Cisco Catalyst switch in Service-Peer mode and discover mDNS services between local wired and wireless FlexConnect user VLANs:

Procedure

1. **enable**

   Example

   ```
   Device# enable
   ```

   Enables privileged EXEC mode.

   Enter your password, if prompted.

2. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

3. **mdns-sd location-filter** *location-filter-name*

   Example

   ```
   Device(config)# mdns-sd location-filter LOCAL-PROXY
   ```

   Configures a unique location-filter in global configuration mode.

4. **match location-group {all | default | ID} VLAN** [*ID*]

   Example

   ```
   Device(config-mdns-loc-filter)# match location-group default vlan 100
   ```

   ```
   Device(config-mdns-loc-filter)# match location-group default vlan 101
   ```

   Configures the match criteria to mutually distribute the permitted services between grouped VLANs. For example, mDNS services can be discovered and distributed using the Unicast mode between wireless FlexConnect user VLAN ID 100 and wired user VLAN ID 101.

5. **mdns-sd service-list** *service-list-name* **{in | out}**

   Example

   ```
   Device(config)# mdns-sd service-list VLAN100-LIST-OUT out
   ```

   Configures the mDNS service-list to classify one or more service types.

   The **service-list** configuration is required to process any incoming or outgoing mDNS messages.

6. **match** *service-definition-name* [**message-type {any | announcement | query}**]

   Example

   ```
   Device(config)# mdns-sd service-list VLAN100-LIST-OUT out
   ```

   ```
   Device(config-mdns-sl-out)# match APPLE-TV location-filter LOCAL-PROXY
   ```

   Associates location-filter to one or more service types to enable local proxy between local VLANs. For example, the Apple-TV learnt from VLAN 100 and VLAN 101 will be distributed to receiver in VLAN 100.

   > **Note**
   >
   > You do not require a **message-type** for the outbound service-list.

7. **mdns-sd service-policy** *service-policy-name*

   Example

   ```
   Device(config)# mdns-sd service-policy VLAN100-POLICY
   ```

   Creates unique mDNS service-policy in global configuration mode.

8. **service-list** *service-list-name* **{in | out}**

Example

```
Device(config)# mdns-sd service-policy VLAN100-POLICY
```

```
Device(config-mdns-ser-policy)# service-list VLAN100-LIST-OUT out
```

Configures mDNS service-policy to associate service-list for each direction.

9. **VLAN configuration** *ID*

Example

```
Device(config)# vlan configuration 100
```

Enables VLAN configuration for advanced service parameters. You can create one or more VLANs with the same settings.

Here, *ID* refers to the VLAN configuration ID. The range is from 101 to 110 and 200. This range allows to configure consecutive and non-consecutive VLAN ID(s).

10. **mdns-sd gateway**

Example

```
Device(config-vlan-config)# mdns-sd gateway
```

Enables mDNS gateway on configured VLAN ID(s).

11. **service-policy** *service-policy-name*

Example

```
Device(config-vlan-mdns-sd)# service-policy VLAN100-POLICY
```

Associates mDNS service-policy to the configured VLAN ID(s).

12. **exit**

Example

```
Device(config-vlan-mdns-sd)# exit
```

Exits mDNS gateway configuration mode.

---

mDNS location-filter is successfully configured to enable local service proxy between the specified VLANs. The mDNS services can now be discovered and distributed between wired and FlexConnect wireless user VLANs.

Configure custom service definition (CLI)

Configure custom service definition using commands to enable end mDNS service-routing in the network.

Create custom service-definition with matching mDNS PTR records to enable end mDNS service-routing in the network.

The Cisco IOS-XE supports mapping of various built-in well-known mDNS service-definition types to key mDNS PTR records and user-friendly names. For example, built-in Apple-TV service-type is associated with _airplay. _tcp.local and

_raop. _tcp.local PTR records to successfully enable service in the network. Network administrators create custom service-definition with matching mDNS PTR records to enable end mDNS service-routing in the network.

Procedure

1. **enable**

   Example

   ```
   Device# enable
   ```

   Enables privileged EXEC mode.

   Enter your password, if prompted.

2. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

3. **mdns-sd service-definition** *service-definition-name*

   Example

   ```
   Device(config)# mdns-sd service-definition APPLE-CLASSROOM
   ```

   Creates a unique service-definition name for custom service-types.

4. **service-type** *custom-mDNS-PTR*

   Example

   ```
   Device(config-mdns-ser-def)# service-type _classroom. _tcp.local
   ```

   Configures a regular-expression string for custom mDNS PoinTeR(PTR) record.

5. **exit**

   Example

   ```
   Device(config-mdns-ser-def)# exit
   ```

   Exits mDNS gateway configuration mode.

The custom service-definition is configured and can be associated to the service-list to enable mDNS service-routing functionality in the network.

Configure service-routing on service-peer using CLI

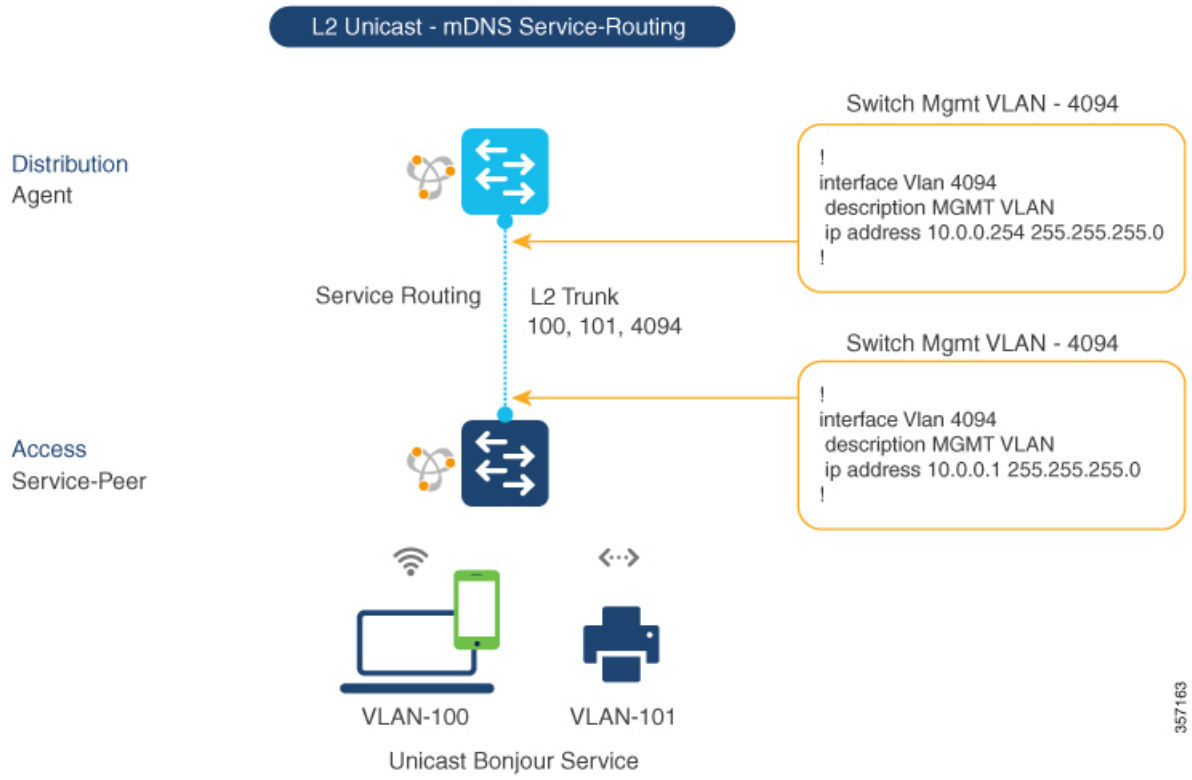
Configure service-routing on service-peer using CLI commands to establish communication with an upstream SDG Agent switch.

Configure service-routing on a Layer 2 Cisco Catalyst switch in Service-Peer mode to establish communication with an upstream distribution-layer switch in SDG Agent mode and enable mDNS trust interface settings.

The Layer 2 Cisco Catalyst switch in Service-Peer mode builds a service-routing with an upstream distribution-layer switch in the SDG Agent mode. To build service-routing, the Layer 2 Cisco Catalyst switch requires at least one interface with valid IP address to reach the upstream SDG Agent Catalyst switch. The switch management port is unsupported.

The following figure illustrates the topology to enable unicast-based service-routing over Layer 2 trunk between access-layer Catalyst switch in the Service-Peer mode and distribution-layer Catalyst switch in SDG Agent mode.

Figure 64: Catalyst Service-Peer Service-Routing configuration



Before you begin

Follow these steps to configure service-routing on Cisco Catalyst switch in Service-Peer mode and setup mDNS trust interface settings:

Procedure

1. **enable**

   Example

   ```
   Device# enable
   ```

   Enables Privileged EXEC mode. Enter your password, if prompted.

2. **configure terminal**

Example

```
Device# configure terminal
```

Enters the global configuration mode.

3. **VLAN configuration** *ID*

   Example

   ```
   Device(config)# vlan configuration 100
   ```

   Enables Wired and FlexConnect user VLAN configuration for advanced service parameters. One or more VLANs can be created for the same settings.

   Here, *ID* refers to the VLAN configuration ID. For example, *VLAN configuration 101-110, 200* range, allows to configure consecutive and non-consecutive VLAN ID(s).

4. **mdns-sd gateway**

   Example

   ```
   Device(config-vlan-config)# mdns-sd gateway
   ```

   Enables mDNS gateway on configured VLAN ID(s).

   To enable the respective functionalities, enter these commands in the mDNS gateway configuration mode:

   - **active-query timer [sec]**: Configure to enable refresh discovered services and their records with periodic mDNS Query message for permitted service types. The valid range is from 60 to 3600 seconds. The recommended value is 3600 seconds.
   - **service-mdns-query {PTR | srv | txt}**: Permits processing specific Query type. The default query type is PTR.
   - **transport {ipv4 | ipv6 | both}**: Permits processing for IPv4, IPv6, or both. It is recommended to use one network type to reduce redundant processing and respond with the same information over two network types. The default network type is IPv4.

5. **source-interface** *ID*

   Example

   ```
   Device(config-vlan-mdns-sd)# source-interface vlan 4094
   ```

   Selects the interface with a valid IP address to source service-routing session with the upstream Cisco Catalyst SDG Agent switch. Typically, the management VLAN interface can be used.

6. **SDG-agent** [*IPv4_address*]

   Example

   ```
   Device(config-vlan-mdns-sd)# sdg-agent 10.0.0.254
   ```

   Configures the SDG Agent IPv4 address, typically, the management VLAN gateway address. If FHRP mode, then use the FHRP virtual IP address of the management VLAN.

7. **exit**

Example

```
Device(config-vlan-mdns-sd)# exit
```

Exits the mDNS gateway configuration mode.

Service-routing is configured on the service-peer, establishing communication with the upstream SDG Agent switch and enabling mDNS gateway functionality on the specified VLAN.

Location-based mDNS

Describes the default per-switch mDNS discovery and distribution functionality for FlexConnect wireless users on Layer 2 Catalyst switches.

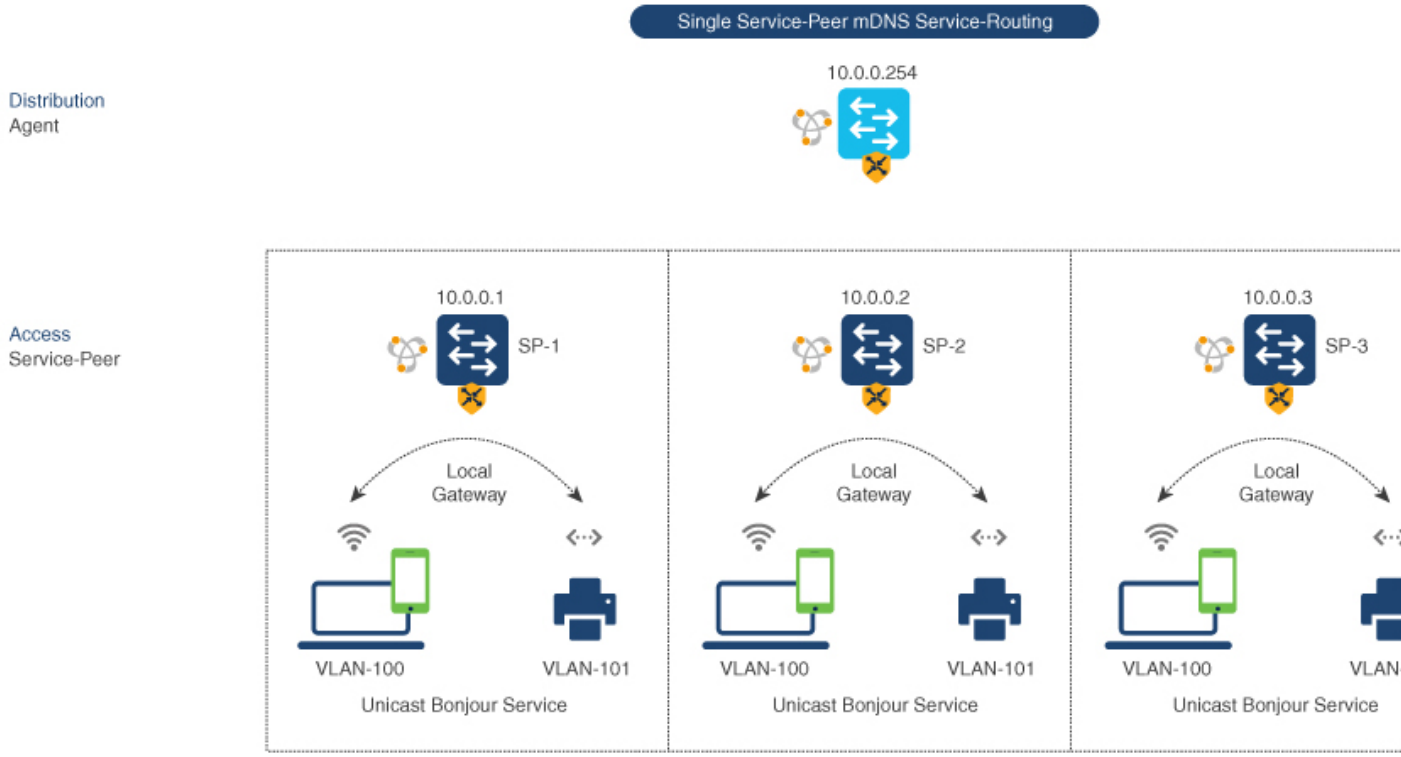
Location-based mDNS is a network service discovery mechanism that

- enables per-switch mDNS discovery and distribution for FlexConnect wireless users attached locally to Layer 2 Catalyst switches in Service-Peer mode by default
- supports user mobility across multiple Layer 2 Catalyst switches even when FlexConnect user VLANs are extended between switches, and
- requires mDNS service-policy configuration on the SDG Agent to accept policy-based service provider and receiver information from downstream Service-Peer access-layer switches.

Per-switch location-based FlexConnect configuration

The following figure shows the per-switch location-based FlexConnect configuration topology.

Figure 65: Per-Switch Location-Based FlexConnect configuration



> **Note**
>
> Configure the mDNS service policy on the distribution layer SDG Agent switch before proceeding to the next configuration step. For more information, see the Configuring mDNS Service Policy section.

Configure service-routing on SDG agent (CLI)

Configure service-routing on SDG agent using CLI commands to enable Unicast mode Bonjour service-routing with downstream Layer 2 access-layer switches.

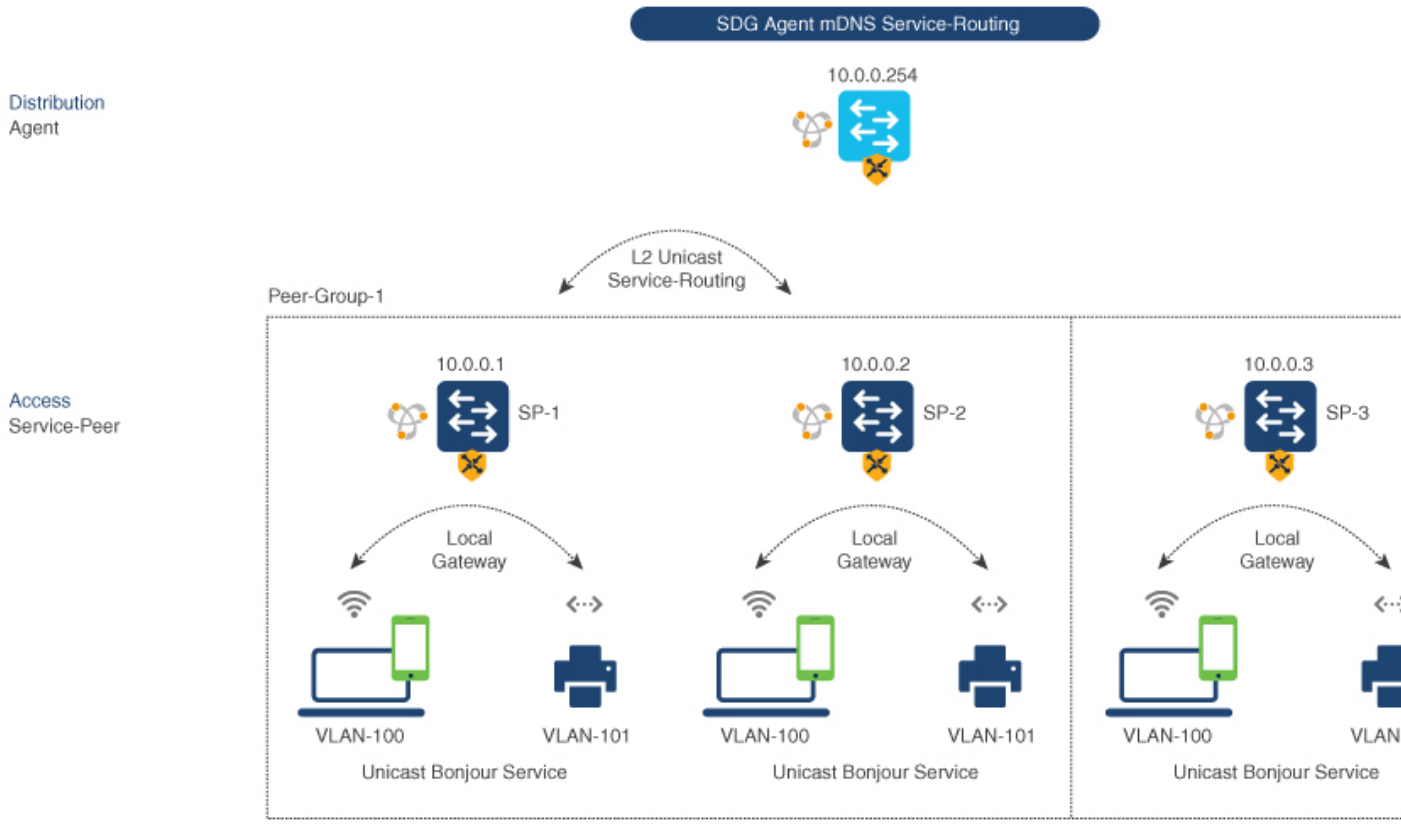
Configure service-routing on SDG agent to enable Unicast mode Bonjour service-routing with downstream Layer 2 access-layer Ethernet switches connected to FlexConnect wireless users.

The Cisco Catalyst 9000 series switches support SDG Agent mode automatically at the distribution layer and enables Unicast mode Bonjour service-routing with the downstream Layer 2 access-layer Ethernet switches connected to the FlexConnect wireless users. The SDG Agent must be configured with mDNS service-policy on wireless FlexConnect user VLAN to accept mDNS service cache from downstream Service-Peer switches.

This section provides step-by-step configuration guidelines to enable policy-based service discovery and distribution between locally paired Layer 2 access network switches in the Service-Peer mode.

The following figure illustrates unicast service-routing on SDG Agent and downstream Layer 2 access network switches in the Service-Peer mode.

Figure 66: Catalyst SDG agent Service-Routing configuration



Before you begin

> **Note**
>
> Configure the mDNS service policy on the distribution layer SDG Agent switch before proceeding to the next configuration step. For more information, see the Configuring mDNS Service Policy section.

Follow these steps to enable the mDNS service policy and peer-group on SDG Agent switch, and enable Unicast mode service-routing with Layer 2 access network switches in Service-Peer mode:

Procedure

1. **enable**

   Example

   ```
   Device# enable
   ```

   Enables privileged EXEC mode.

   Enter your password, if prompted.

2. **configure terminal**

   Example

   ```
   Device# configure terminal
   ```

   Enters global configuration mode.

3. **mdns-sd service-peer group** *service-peer-group-name*

   Example

   ```
   Device(config)# mdns-sd service-peer group group_1
   ```

   Configures a unique Service-Peer group.

4. **peer-group** [*ID*]

   Example

   ```
   Device(config-mdns-svc-peer)# peer-group 1
   ```

   Assigns a unique peer-group ID to the Service-Peers pair permitting mDNS service discovery and distribution within the assigned group list.

   The valid peer-group range is from 1 to 1000 for each SDG Agent switch.

5. **service-policy** *service-policy-name*

   Example

   ```
   Device(config-mdns-svc-peer-grp)# service-policy VLAN100-POLICY
   ```

Associates an mDNS service policy to accept service advertisements and query from the paired Service-Peers.

6. **service-peer** [*IPv4_address*] **location-group** {**all** | **default** | **ID**}

   Example

   ```
   Device(config-mdns-svc-peer-grp)# service-peer 10.0.0.1 location-group default
   ```

   ```
   Device(config-mdns-svc-peer-grp)# service-peer 10.0.0.2 location-group default
   ```

   Configures at least one Service-Peer to accept the mDNS service advertisement or query message. When a group has more than one Service-Peers, the SDG Agent provides Layer 2 Unicast mode routing between the configured peers.

   For example, the SDG Agent provides Unicast based service gateway function between three (10.0.0.1 and 10.0.0.2) Layer 2 Service-Peer switches matching the associated service-policy.

   The mDNS service information from the unpaired Layer 2 Service-Peer (10.0.0.3) cannot announce or receive mDNS services with the other grouped Service-Peers (10.0.0.1 and 10.0.0.2).

7. **exit**

   Example

   ```
   Device(config-mdns-svc-peer-grp)# exit
   ```

   Exits mDNS gateway configuration mode.

The SDG Agent is configured with service-routing to enable Unicast mode service-routing with downstream Layer 2 access network switches in Service-Peer mode.

Verify local area Bonjour in service-peer mode

Provides guidelines to verify various Local Area Bonjour domain mDNS service configuration parameters in service-peer mode.

This section provides guidelines to verify various Local Area Bonjour domain mDNS service configuration parameters, cache records, statistics and more on the controller in service-peer mode.

Command or Action	Purpose
show mdns-sd cache {all | interface | MAC | name | service-peer | static | type | VLAN}	Displays available mDNS cache records supporting multiple variables providing granular source details received from wired or wireless FlexConnect user VLANs. The variables are as follows: • all – Displays all available cache records discovered from multiple source connections of a system. • interface – Displays available cache records discovered from the specified Layer 3 interface. • MAC – Displays available cache records discovered from the specified MAC address. • name – Displays available cache records based on the service provider announced name. • service-peer – Displays available cache records discovered from the specified Layer 2 Service-Peer. • static – Displays locally configured static mDNS cache entry. • type – Displays available cache records based on the specific mDNS record type, such as, PTR, SRV, TXT, A or AAAA. • VLAN – Displays available cache records discovered from the specified Layer 2 VLAN ID in the Unicast mode.
show mdns-sd service-definition {name | type}	Displays built-in and user-defined custom service-definition that maps service name to the mDNS PTR records. The service-definition can be filtered by name or type.
show mdns-sd service-list {direction | name}	Displays inbound or outbound direction list of configured service-list to classify matching service-types for service-policy. The list can be filtered by name or specific direction.
show mdns-sd service-policy {interface | name}	Displays list of mDNS service-policy mapped with inbound or outbound service-list. The service-policy list can be filtered by an associated specified interface or name.

Command or Action	Purpose
show mdns-sd statistics {all | cache | debug | interface | service-list | service-policy | services | VLAN}	Displays detailed mDNS statistics processed bi-directionally by the system on each mDNS gateway enabled VLAN configured mDNS in Unicast mode. The expanded keyword for mDNS statistics can provide detailed view on interface, policy, service-list, and services. **Note** This command displays all mDNS packets received from directly connected (Local Mode) or Flex clients in WLAN.
show mdns-sd summary {interface | VLAN}	Displays brief information about mDNS gateway and key configuration status on all wired and wireless FlexConnect user VLANs, and interfaces of the system.

Verify local area Bonjour in SDG agent mode

Provides guidelines to verify various local area Bonjour domain mDNS service configuration parameters in SDG agent mode.

This section provides guidelines to verify various Local Area Bonjour domain mDNS service configuration parameters, cache records, statistics and more on the controller in SDG Agent mode

Command or Action	Purpose
show mdns-sd cache {all | interface | MAC | name | service-peer | static | type | VLAN | VRF}	Displays available mDNS cache records supporting multiple variables providing granular source details. The variables are as follows: • all – Displays all available cache records discovered from multiple source connections of a system. • interface – Displays available cache records discovered from the specified Layer 3 interface. • MAC – Displays available cache records discovered from the specified MAC address. • name – Displays available cache records based on the service provider announced name. • service-peer – Displays available cache records discovered from the specified Layer 2 Service-Peer. • static – Displays locally configured static mDNS cache entry. • type – Displays available cache records based on the specific mDNS record type, such as, PTR, SRV, TXT, A or AAAA. • VLAN – Displays available cache records discovered from the specified Layer 2 VLAN ID in the Unicast mode. • VRF – Displays per-VRF available cache records based on specific mDNS record type, i.e., PTR, SRV, TXT, A or AAAA.
show mdns-sd service-definition {name | type}	Displays built-in and user-defined custom service-definition that maps service name to the mDNS PTR records. The service-definition can be filtered by name or type.
show mdns-sd service-list {direction | name}	Displays inbound or outbound direction list of the configured service-list to classify matching service-types for service-policy. The list can be filtered by name or specific direction.
show mdns-sd service-policy {interface | name}	Displays list of mDNS service-policy mapped with inbound or outbound service-list. The service-policy list can be filtered by an associated specified interface or name.
show mdns-sd statistics {all | cache | debug | interface | service-list | service-policy | services | VLAN}	Displays detailed mDNS statistics processed bi-directionally by the system on each mDNS gateway enabled VLAN configured mDNS in Unicast mode. The expanded keyword for mDNS statistics can provide detailed view on interface, policy, service-list, and services.
show mdns-sd summary {interface | VLAN}	Displays brief information about mDNS gateway and key configuration status on all VLANs and interfaces of the system.

Reference

Reference for local area Bonjour.

Table 203:

Related Topic	Document Title
DNA Service for Bonjour Deployment on Cisco Catalyst 9600 Switch	Cisco Catalyst 9600 Series Switch Software Configuration Guide, Release 17.5.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9500 Switch	Cisco Catalyst 9500 Series Switch Software Configuration Guide, Release 17.5.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9400 Switch	Cisco Catalyst 9400 Series Switch Software Configuration Guide, Release 17.5.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9300 Switch	Cisco Catalyst 9300 Series Switch Software Configuration Guide, Release 17.5.X
Cisco Wide Area Bonjour Application on Cisco Catalyst Center User Guide	Cisco Wide Area Bonjour Application on Cisco Catalyst Center User Guide, Release 2.2.2

233 Configuration Example for Local Mode - Wireless and Wired

Topics:

- Overview

Demonstrates a practical configuration scenario for implementing local mode deployment across both wireless and wired network infrastructure components.

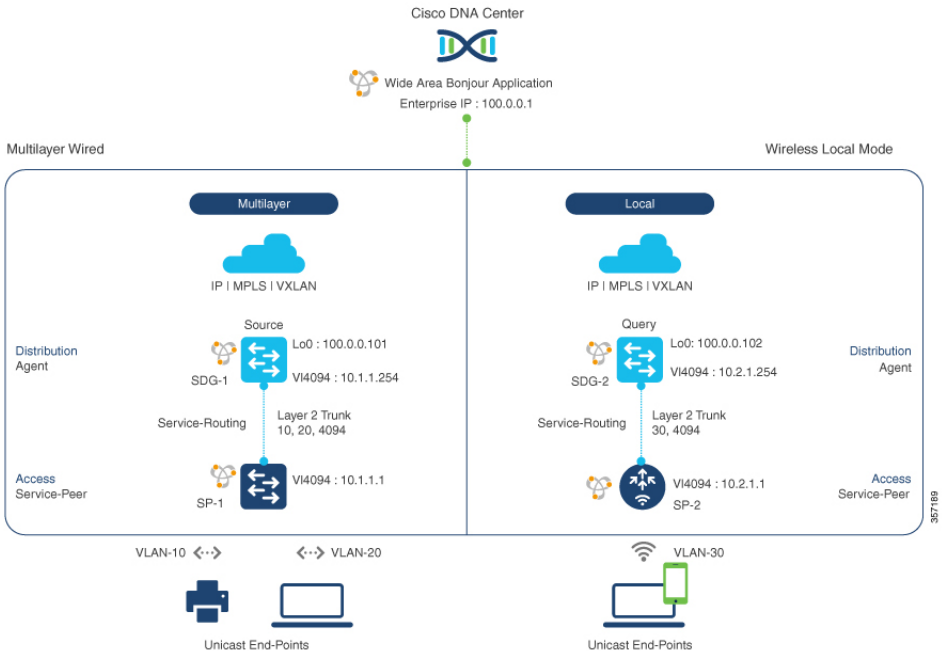
Overview

Introduces the framework for local mode deployments across multilayer wired and wireless environments, including wireless AP multicast configuration, Wide Area Bonjour examples with default and customized service lists, Cisco Catalyst Center traditional setups, verification procedures, and supporting reference material.

This chapter provides configuration guidelines to implement Wide Area Bonjour enabling end-to-end policy-based mDNS service discovery and distribution across multilayer wired and wireless local mode. The first hop mDNS gateway at Layer 2 access switch and the controller must be implemented in service peer mode and paired with LAN and wireless distribution-layer switch in SDG agent role. The network-wide distributed SDG agent must be paired with the Cisco Catalyst Center to enable mDNS service-routing across IP core network based on multiple services and network attributes.

The following figure illustrates unicast mode bonjour network environment with AirPrint capable printer and user computer (macOS, Microsoft Windows, etc.) connected to same Ethernet switch. The computers and mobile devices of the wireless user are associated to wireless AP in local mode across multi-hop IP boundary from printers.

Figure 67: Wide Area Bonjour Service-Routing Multilayer Wired and Wireless Local Mode



Configuring Wireless AP Multicast Mode

Configure wireless AP multicast mode.

This procedure configures wireless AP multicast on the controller for local mode APs and IP network.

The controller must be configured with unique IP multicast address for wireless AP in local mode to permit mDNS communication across wired and wireless networks.

Step	Controller Service Peer Configuration
Step-1 Enable global IP Multicast on Cisco Catalyst 9800 series controller.	``` ! wireless multicast ! ```
Step-2 Configure Wireless AP mode to Multicast with unique IP Multicast address.	``` ! wireless multicast 239.254.254.1 ! ```

The following table provides step-by-step IP multicast configuration guidelines on SDG agent (SDG-1 and SDG-2) at the distribution layer network.

Step	Switch SDG Agent Configuration	WLC SDG Agent Configuration
Step-1 Enable IP multicast-routing on distribution layer switches connecting Cisco Wireless Local Mode Access Point and Cisco Wireless LAN Controller.	``` ! ip multicast-routing ! ```	``` ! ip multicast-routing ! ```
Step-2 Configure IP PIM Rendezvous-Point (RP) on distribution layer switches.	``` ! ip pim rp-address 10.150.255.1 ! ```	``` ! ip pim rp-address 10.150.255.1 ! ```
Step-3 Enable IP PIM on SVI Interface of distribution layer switches connected Cisco Wireless Local Mode Access Point and Cisco WLC Management VLAN.	``` ! interface Vlan 101 description CONNECTED TO WIRELESS AP - LOCAL MODE ip pim sparse-mode ! ```	``` ! interface Vlan 4094 description CONNECTED TO WIRELESS MGMT - WLC ip pim sparse-mode ! ```
Step-4 Enable IP PIM on Layer 3 uplink Interface of distribution layer switches connected Cisco Wireless Local Mode Access Point and Cisco WLC Management VLAN.	``` ! interface range FortyGigabitEthernet 1/1/1 - 2 description CONNECTED TO IP CORE NETWORK ip pim sparse-mode ! ```	``` ! interface range FortyGigabitEthernet 1/1/1 - 2 description CONNECTED TO IP CORE NETWORK ip pim sparse-mode ! ```

Note

IP Multicast must be enabled in the Layer 3 core network to allow Cisco wireless APs in local mode to successfully join the WLC announced multicast group. For more information, refer to the Cisco online documentation to implement IP multicast networks.

Configuration Example for Default Service List and Policy in Wide Area Bonjour Between Multilayer Wired and Wireless Endpoints

Configuration example for default service list and policy in wide area bonjour between multilayer wired and wireless endpoints.

This section provides guidance on configuring Service-Peer, SDG Agent, and Cisco Catalyst Center, allowing the wired and wireless endpoints to dynamically discover default service list using Layer 2 unicast and policy.

Example: Wired and Wireless Access Layer Service Peer Configuration

Example for wired and wireless access layer service peer configuration.

The following table provides a sample configuration of wired and wireless controller access layer service peer.

Table 204: Configuring Wired and Wireless Access Layer Service Peer

Configuration Step	Sample Configuration: SP-1 Service-Peer Configuration	Sample Configuration: SP-2 Service-Peer Configuration
Step-1: Enable mDNS gateway and set the gateway mode. **Note** In wireless controller, service peer mode is enabled by default with mDNS gateway configuration.	```! mdns-sd gateway mode service-peer !```	```! mdns-sd gateway mode service-peer !```
Step-2: Activate unicast mDNS gateway and attach service policy on wired VLAN and wireless FlexConnect user VLAN of SP-1 and SP-2 Layer 2 access switch.	```! vlan configuration 10, 30 mdns-sd gateway service-policy LOCAL-AREA-POLICY active-query timer 3600 !```	```! vlan configuration 20, 30 mdns-sd gateway service-policy LOCAL-AREA-POLICY active-query timer 3600 !```
Step-3: Enable unicast service routing between wired and wireless service peer and SDG agent using wired management source VLAN ID and IP address.	```vlan configuration 10, 30 mdns-sd gateway source-interface vlan 4094 sdg-agent 10.1.1.254 !```	```! vlan configuration 20, 30 mdns-sd gateway source-interface vlan 4094 sdg-agent 10.1.1.254 !```

Example: Wired and Wireless Distribution Layer SDG Agent Configuration

Example for wired and wireless distribution layer SDG agent configuration.

The following table provides a sample configuration of distribution layer SDG agent.

Table 205: Configuring Wired and Wireles Distribution Layer SDG Agent

Configuration Step	Sample Configuration: SDG-1 – SDG Agent
Step-1: Enable mDNS gateway and set the gateway mode. The default mode is sdg-agent.	`! mdns-sd gateway !`
Step-2: Activate unicast mDNS gateway on wired VLAN and wireless user VLAN on SDG agents.	`! vlan configuration 10, 20, 30  mdns-sd gateway !`
Step-3: Configure the service peer-group and attach service-policy on the SDG agent distribution switch and enable service-routing between the assigned Service Peer switch group.	`! mdns-sd service-peer group  peer-group 1   service-policy LOCAL-AREA-POLICY  service-peer 10.1.1.1 location-group  default  service-peer 10.1.1.2 location-group  default !`
Step-4: Associate outbound service-list to a unique service-policy.	`! mdns-sd service-policy WIDE-AREA-POLICY  service-list WIDE-AREA-SERVICES-OUT !`
Step-5: Enable Wide Area Bonjour service-routing with service export configuration association controller IP Address, source interface for stateful connection, and mandatory egress policy for Wide Area service-routing.	`! service-export mdns-sd controller DNAC-CONTROLLER-POLICY  controller-address 100.0.0.1  controller-source-interface LOOPBACK  0  controller-service-policy WIDE-AREA-POLICY !`

Configuration Example for Customized Service List and Policy in Wide Area Bonjour Between Multilayer Wired and Wireless Endpoints

Configuration example for customized service list and policy in wide area Bonjour between multilayer wired and wireless endpoints.

This section provides guidance on configuring Service-Peer, SDG Agent and Cisco Catalyst Center, allowing the wired and wireless endpoints to dynamically discover printer using Layer 2 unicast and policy.

Example: Wired and Wireless Access Layer Service Peer Configuration

Example for wired and wireless access layer service peer configuration.

The following table provides a sample configuration of wired and wireless controller access layer service peer.

Table 206: Configuring Wired and Wireless Access Layer Service Peer

Configuration Step	Sample Configuration: Switch Service Peer	Sample Configuration: Wireless Controller Service Peer
Step-1: Enable mDNS gateway and set the gateway mode. **Note** In wireless controller, service peer mode is enabled by default with mDNS gateway configuration.	``` ! mdns-sd gateway mode service-peer ! ```	``` ! mdns-sd gateway ! ```
Step-2: Create unique mDNS inbound policy to permit ingress AirPrint service announcement on the Catalyst Switch and wireless controller in service peer mode.	``` ! mdns-sd service-list LOCAL-AREA-SERVICES-IN in match printer-ipp ! ```	``` ! mdns-sd service-list LOCAL-AREA-SERVICES-IN in match printer-ipp ! ```
Step-3: Create unique mDNS outbound policy to permit egress AirPrint service response on the Catalyst Switch and wireless controller in service peer mode	``` ! mdns-sd service-list LOCAL-AREA-SERVICES-OUT out match printer-ipp ! ```	``` ! mdns-sd service-list LOCAL-AREA-SERVICES-OUT out match printer-ipp ! ```
Step-4: Associate inbound and outbound service list to a unique service policy.	``` ! mdns-sd service-policy LOCAL-AREA-POLICY service-list LOCAL-AREA-SERVICES-IN service-list LOCAL-AREA-SERVICES-OUT ! ```	``` mdns-sd service-policy LOCAL-AREA-POLICY service-list LOCAL-AREA-SERVICES-IN service-list LOCAL-AREA-SERVICES-OUT ! ```

Step-5: Activate unicast mDNS gateway and attach service policy on wired VLAN and WLAN.
• **Switch**: Activate mDNS gateway per VLAN.
• **Controller**: Activate mDNS gateway per WLAN policy profile and SSID
	``
	!
	vlan configuration 10,
	20
	 mdns-sd gateways
	  service-policy
	LOCAL-AREA-POLICY
	  active-query timer
	3600
	!
	``
		``
		!
		wireless profile policy
		 WLAN-PROFILE
		 shutdown
		 mdns-sd service-policy
		 LOCAL-AREA-POLICY
		 no shutdown
		!
		wlan WLAN-PROFILE 1
		blizzard
		 shutdown
		 mdns-sd-interface
		gateway
		 no shutdown
		!
		``

Configuration Step	Sample Configuration: Switch Service Peer	Sample Configuration: Wireless Controller Service Peer
Step-6: (Optional) Enable service routing on wired service peer mDNS between local VLANs. Also, enable location-based wireless service on the controller. • **Switch**: Configure location filter group to discover and distribute between paired local VLAN. • **Controller**: Configure wireless location-based services.	```! mdns-sd location-filter LOCAL-PROXY match location-group default vlan 10 match location-group default vlan 20 ! mdns-sd service-list LOCAL-AREA-SERVICES-OUT OUT match printer-ipps location-filter LOCAL-PROXY !```	```! mdns-sd service-policy LOCAL-AREA-POLICY location ap-location !```
Step-7: Enable unicast service routing between wired and wireless service peer and SDG agent. • **Switch**: Configure SDG agent IP and wired management source VLAN ID and IP address. • **Controller**: Configure SDG Agent IP and wireless management source VLAN ID and IP address.	```vlan configuration 10, 20 mdns-sd gateways source-interface vlan 4094 sdg-agent 10.1.1.254 !```	```! mdns-sd gateway source-interface vlan 4094 sdg-agent 10.2.1.254 !```

Example: Wired and Wireless Distribution Layer SDG Agent Configuration

Example for wired and wireless distribution layer SDG agent configuration.

The following table provides a sample configuration of distribution layer SDG agent.

Table 207: Configuring Distribution Layer SDG Agent

Configuration Step	Sample Configuration: Wired SDG Agent	Sample Configuration: Wireless SDG Agent
Step-1: Enable mDNS gateway and set the gateway mode.	```! mdns-sd gateway !```	```! mdns-sd gateway !```
Step-2: Activate unicast mDNS gateway on wired VLAN and wireless user VLAN on respective SDG agents.	```! vlan configuration 10, 20 mdns-sd gateway !```	```! vlan configuration 30 mdns-sd gateway !```
Step-3: Create unique controller bound mDNS policy to permit egress AirPrint service discovery and distribution from Catalyst Switch in SDG agent mode. Inbound policy towards controller is not required.	```! mdns-sd service-list WIDE-AREA-SERVICES-OUT out match printer-ipp !```	```! mdns-sd service-list WIDE-AREA-SERVICES-OUT out match printer-ipp !```
Step-4: Associate outbound service-list to a unique service-policy.	```! mdns-sd service-policy WIDE-AREA-POLICY service-list WIDE-AREA-SERVICES-OUT !```	```! mdns-sd service-policy WIDE-AREA-POLICY service-list WIDE-AREA-SERVICES-OUT !```
Step-5: Enable Wide Area Bonjour service-routing with service export configuration association controller IP Address, source interface for stateful connection and mandatory egress policy for Wide Area service-routing.	```! service-export mdns-sd controller DNAC-CONTROLLER-POLICY controller-address 100.0.0.1 controller-source-interface LOOPBACK 0 controller-service-policy WIDE-AREA-POLICY !```	```! service-export mdns-sd controller DNAC-CONTROLLER-POLICY controller-address 100.0.0.1 controller-source-interface LOOPBACK 0 controller-service-policy WIDE-AREA-POLICY !```

Cisco Catalyst Center Traditional Multilayer Wired and Wireless Configuration

Cisco DNA center traditional multilayer wired and wireless configuration.

Configuring Service Filters for Traditional Multilayer Wired and Wireless - Local Mode (GUI)

Configure service filters for traditional multilayer wired and wireless in local mode using GUI.

This procedure implements global service filters, which permit the Cisco Wide Area Bonjour application to dynamically discover and distribute service information between trusted Cisco Catalyst SDG agent switches across the IP network.

Procedure

1. Navigate to the **Configuration** tab in the Wide Area Bonjour application.
2. From the sidebar, select the subdomain for which you want to create the service filter.
3. Check the **Service Filter** box.
4. Click **Service Filter** icon from the topology to view a list of the service filters for the selected domain. You can also manually edit existing service filters from this list.
5. Click **Create Service Filter**.
6. From the **Network Mode** drop-down list, choose **Traditional** (the default mode).
7. Enter a unique name for the service filter.
8. (Optional) Enter a description for the service filter.
9. Select one or more service types to permit announcements and queries.
10. Enable or disable service filters after creating them. By default, service filters are enabled.

Configuring Source SDG Agents in Traditional Multilayer Wired and Wireless - Local Mode (GUI)

Configure source SDG agents in traditional multilayer wired and wireless - local mode using GUI.

This procedure configures discovery of wired printer sources from the LAN distribution switches paired with Layer 2 Catalyst Switches in a service peer role. The wireless distribution switches paired with a controller in a service peer role receive query responses for wired printers and distribute the responses to querying devices over the wireless local mode network.

Procedure

1. Click **Add** in the upper-right portion of the Catalyst Center Policy screen.
2. Select the **Query SDG agent** radio button. By default, the **Source** radio button is selected.
3. From the **SDG Agent/IP** drop-down list, select an SDG agent (100.0.0.101) which announces the services, for example, Printer.
4. Select **Peer** from the **Service Layer** drop-down list.
5. Uncheck the box **Any**. By default, this is enabled.
6. Select the query **VLAN** (Vlan-10) to distribute services (Printer) from a specific network.
7. Enable or disable services from the selected query IPv4 subnet. By default, this is enabled.
8. Enable or disable services from the selected query IPv6 subnet. By default, this is enabled.
9. Enter the **service peer IPv4 address** (10.1.1.1).
10. Click the **+** icon to add more service peers, if any. Select **Any** to accept services from any peer on a selected VLAN.
11. (Optional) Click **Add Next** to add more source SDG agents. (Repeat the preceding steps.)
12. Click **DONE**.
13. Click **CREATE**.

Configuring Query SDG Agents in Traditional Multilayer Wired and Wireless - Local Mode (GUI)

Configure query SDG agents in traditional multilayer wired and wireless - local mode using GUI.

This procedure configures distributed services to query SDG agents connected to a controller in service peer mode, based on a policy.

Procedure

1. Click **Add** in the upper-right portion of the DNA-Center Policy screen.
2. Select the **Query SDG agent** radio button. By default, the **Source** radio button is selected.
3. From the **SDG Agent/IP** drop-down list, select an SDG agent (100.0.0.102) that receives queries for the services (Printer).
4. Select **Peer** from the **Service Layer** drop-down list.
5. Uncheck the box **Any**. By default, this is enabled.
6. Select the query **VLAN** (Vlan-30) to distribute services (Printer) to a specific network.
7. Enable or disable services from the selected query IPv4 subnet. By default, this is enabled.
8. Enable or disable services from the selected query IPv6 subnet. By default, this is enabled.
9. Enter the **service peer IPv4 address** (10.2.1.254).
10. Click the **+** icon to add more service-peers, if any. Select **Any** to accept services from any peer on a selected VLAN.
11. (Optional) Click **Add Next** to add more query agents. (Repeat the preceding steps.)
12. Click **DONE**.
13. Click **CREATE**.

Verifying Wide Area Bonjour Between Multilayer Wired and Wireless Local Mode

Verify wide area Bonjour between multilayer wired and wireless local mode.

This section provides step-by-step mDNS configuration and service discovery and distribution status based on applied policy on Wired Layer 2 access switch in service peer and SDG agent mode.

Verifying Wired Service-Peer Configuration

Verify wired service-peer configuration.

Use the following commands on the Cisco Catalyst switch in service peer (SP-1) mode to determine the operational status after applying configuration and discovering the AirPrint service from the local network.

```
Device# show mdns-sd summary vlan 10

VLAN: 10
========================================
mDNS Gateway: Enabled
mDNS Service Policy: LOCAL-AREA-POLICY
Active Query: Enabled
            : Periodicity 3600 Seconds
Transport Type: IPv4
Service Instance Suffix: Not Configured
mDNS Query Type: ALL
SDG Agent IP: 10.1.1.254
Source Interface: Vlan4094

Device# show mdns-sd service-policy name LOCAL-AREA-POLICY

Service Policy Name Service List IN Name Service List Out Name
================================================================================
LOCAL-AREA-POLICY        LOCAL-AREA-SERVICES-IN      LOCAL-AREA-SERVICES-OUT

Device# show mdns-sd cache vlan 10
```

Name	Type	TTL/ Remaining	Vlan-Id/ Interface-name	MAC Address	RR Record Data
_universal. _sub. _ipp. _tcp.local	PTR	4500/4486	Vl10	ac18.2651.03fe	Bldg-1-FL1-PRN. _ipp. _tcp.local
_ipp. _tcp.local	PTR	4500/4486	Vl10	ac18.2651.03fe	Bldg-1-FL1-PRN. _ipp. _tcp.local
Bldg-1-FL1-PRN. _ipp. _tcp.local	SRV	4500/4486	Vl10	ac18.2651.03fe	Bldg-1-FL1-PRN. local
Bldg-1-FL1-PRN.local	A	4500/4486	Vl10	ac18.2651.03fe	10.153.1.1
Bldg-1-FL1-PRN.local	AAAA	4500/4486	Vl10	ac18.2651.03fe	2001:10:153: 1:79: A40C:6BEE: AEEC
Bldg-1-FL1-PRN. _ipp. _tcp.local	TXT	4500/4486	Vl10	ac18.2651.03fe	(451)'txtvers=1''priority=30'' ty=EPSON WF-3620 Series'' usb_MFG=EPSON'' usb_MDL=W~'~

```
Device# show mdns-sd statistics vlan 10

 mDNS Statistics

 Vl10:
  mDNS packets sent              : 612
     IPv4 sent                   : 612
       IPv4 advertisements sent  : 0
       IPv4 queries sent         : 612
     IPv6 sent                   : 0
       IPv6 advertisements sent  : 0
       IPv6 queries sent         : 0
  Unicast sent                   : 0
  mDNS packets rate limited      : 0
  mDNS packets received          : 42
   advertisements received       : 28
   queries received              : 14
     IPv4 received               : 42
       IPv4 advertisements received: 28
       IPv4 queries received       : 14
     IPv6 received                 : 0
       IPv6 advertisements received: 0
       IPv6 queries received       : 0
  mDNS packets dropped           : 0
  ========================================
  Query Type                       : Count
  ========================================
  PTR                              : 12
  SRV                              : 0
  A                                : 0
  AAAA                             : 0
```

```
TXT                                 : 0
ANY                                 : 3
================================================
PTR Name                  Advertisement     Query
================================================
_ipp. _tcp.local            9           4
```

Verifying Wired SDG Agent Configuration and Service-Routing Status

Verify wired SDG agent configuration and service-routing status.

This section provides information on mDNS configuration and service-routing on Wired SDG Agent (SDG-1) with locally attached Layer 2 access switches in Service-Peer (SP-1) mode and with centrally paired Cisco Catalyst Center for Wide Area Bonjour service-routing.

```
Device# show mdns-sd summary vlan 10

VLAN: 10
========================================
mDNS Gateway               : Enabled
mDNS Service Policy        : LOCAL-AREA-POLICY
Active Query               : Disabled
Transport Type             : IPv4
Service Instance Suffix    : Not-Configured
mDNS Query Type            : ALL
SDG Agent IP               : Not-Configured
Source Interface           : Not-Configured
```

```
Device# show mdns-sd cache vlan 10

VLAN: 10
========================================
mDNS Gateway               : Enabled
mDNS Service Policy        : LOCAL-AREA-POLICY
Active Query               : Disabled
Transport Type             : IPv4
Service Instance Suffix    : Not-Configured
mDNS Query Type            : ALL
SDG Agent IP               : Not-Configured
Source Interface           : Not-Configured
```

Name	Type	TTL/ Remaining	Vlan-Id /Interface-name	MAC Address	RR Record Data
_universal._sub._ipp ._tcp.local	PTR	4500/4500	Vl10	ac18.2651.03fe	Bldg-1-FL1-PRN. _ipp. _tcp.local
_ipp. _tcp.local	PTR	4500/4500	Vl10	ac18.2651.03fe	Bldg-1-FL1-PRN. _ipp. _tcp.local
Bldg-1-FL1-PRN. _ipp. _tcp.local	SRV	4500/4500	Vl10	ac18.2651.03fe	0 0 631 Bldg-1-FL1-PRN. local

Name	Type	TTL/ Remaining	Vlan-Id /Interface-name	MAC Address	RR Record Data
Bldg-1-FL1 -PRN.local	A	4500/4500	Vl10	ac18.2651.03fe	10.153.1.1
Bldg-1-FL1-PRN.local	AAAA	4500/4500	Vl10	ac18.2651.03fe	2001:10:153: 1:79: A40C:6BEE: AEEC
Bldg-1-FL1-PRN. _ipp. _tcp.local	TXT	4500/4500	Vl10	ac18.2651.03fe	(451)txtvers=1'prioty=30' ty=EPSON WF-3620 Series''

```
Device# show mdns-sd sp-sdg statistics

                                  One min, 5 mins, 1 hour
Average Input rate (pps)       : 0,       0,       0
Average Output rate (pps)      : 0,       0,       0
 Messages received:
  Query                        : 15796
  ANY query                    : 0
  Advertisements               : 28
  Advertisement Withdraw       : 0
  Interface down               : 0
  Vlan down                    : 0
  Service-peer ID change       : 0
  Service-peer cache clear     : 12
  Resync response              : 6
 Messages sent:
  Query response               : 5975
  ANY Query response           : 0
  Cache-sync                   : 61
  Get service-instance         : 0

Device# show mdns-sd controller detail

Controller: DNAC-Policy
 IP: 100.0.0.1, Dest Port : 9991, Src Port : 42446, State : UP
 Source Interface : Loopback0, MD5 Disabled
 Hello Timer 30 sec, Dead Timer 120 sec, Next Hello 00:00:24
 Uptime 2d05h (17:02:37 UTC Jan 15 2021)
 Service Buffer: Enabled

Service Announcement:
 Filter: DNAC-CONTROLLER-POLICY
 Count 50, Delay Timer 30 sec, Pending Announcement 0, Pending Withdraw 0
 Total Export Count 56, Next Export in 00:00:24

Service Query:
 Query Suppression Enabled
 Query Count 50, Query Delay Timer 15 sec, Pending 0
 Total Query Count 15791, Next Query in 00:00:09
```

Verifying Wireless Service-Peer Configuration and Service Status

Verify wireless service-peer configuration and service status.

The command given below helps determine the operational status after applying configuration and discovering the AirPrint service from the remote network.

```
Device# show mdns-sd summary

mDNS Gateway: Enabled
Mode: Service Peer
Service Announcement Periodicity (in seconds): 30
Service Announcement Count: 50
Service Query Periodicity (in seconds): 15
Service Query Count: 50
Active Response Timer (in seconds): Disabled
ANY Query Forward: Disabled
SDG Agent IP: 10.2.1.254
Source Interface: Vlan4094
Active Query Periodicity (in minutes): 15
Transport Type: IPv4
mDNS AP service policy: default-mdns-service-policy

Device#  show wireless profile policy detailed WLAN-PROFILE | sec mDNS

mDNS Gateway
  mDNS Service Policy name          : LOCAL-AREA-POLICY

Device# show mdns-sd statistics wlan-id 1

mDNS Packet Statistics
------------------------------------------------
mDNS stats last reset time: 01/10/21 21:38:19
mDNS packets sent: 4592
  IPv4 sent: 4592
    IPv4 advertisements sent: 4592
    IPv4 queries sent: 0
  IPv6 sent: 0
    IPv6 advertisements sent: 0
    IPv6 queries sent: 0
  Multicast sent: 0
    IPv4 sent: 0
    IPv6 sent: 0
mDNS packets received: 297
  advertisements received: 80
  queries received: 217
  IPv4 received: 297
    IPv4 advertisements received: 80
    IPv4 queries received: 217
  IPv6 received: 0
    IPv6 advertisements received: 0
    IPv6 queries received: 0
mDNS packets dropped: 297
Query Type Statistics
  PTR queries received: 1720
  SRV queries received: 8
  A query received: 8
  AAAA queries received: 8
  TXT queries received: 97
  ANY queries received: 153
  OTHER queries received: 0
```

```
Device# show mdns-sd sp-sdg statistics

mDNS SP Statistics
last reset time: 01/10/21 21:37:36

  Messages sent:
   Query                      : 12675
   ANY query                  : 0
   Advertisements             : 24
   Advertisement Withdraw             : 0
   Service-peer ID change             : 0
   Service-peer cache clear           : 7
   Resync response                    : 5
  Messages received:
   Query response                     : 4619
   ANY Query response                 : 0
   Cache-sync                         : 48
   Get service-instance               : 0

Device# show mdns-sd query-db

MDNS QUERY DB

Client MAC: 4c32. 7593.e3af
  Vlan ID: 30
  Wlan ID: 1
  Location Group ID: 0
    PTR Name(s):
    _ipp. _tcp.local
```

Verifying Wireless SDG Agent Configuration and Service-Routing Status

Verify wireless SDG agent configuration and service-routing status.

This section providees information on mDNS configuration and service-routing on Wireless SDG Agent (SDG-2) with locally attached controller in service peer (SP-2) mode and with centrally paired Cisco DNA-Center for Wide Area Bonjour service-routing.

```
Device# show mdns-sd summary vlan 30

VLAN: 30
========================================
mDNS Gateway            : Enabled
mDNS Service Policy     : LOCAL-AREA-POLICY
Active Query            : Disabled
Transport Type          : IPv4
Service Instance Suffix : Not Configured
mDNS Query Type         : ALL
SDG Agent IP            : Not Configured
Source Interface        : Not Configured

Device# show mdns-sd sp-sdg statistics

                                One min, 5 mins, 1 hour
Average Input rate (pps)        :0,       0,          0
Average Output rate (pps)       :0,       0,          0
 Messages received:
  Query                         : 12191
  ANY query                     : 0
  Advertisements                : 0
```

```
  Advertisement Withdraw        : 0
  Interface down                : 0
  Vlan down                     : 0
  Service-peer ID change        : 0
  Service-peer cache clear      : 18
  Resync response               : 10
 Messages sent:
  Query response                : 1975
  ANY Query response            : 0
  Cache-sync                    : 19
  Get service-instance          : 0

Device# show mdns-sd controller detail

Controller: DNAC-Policy
 IP: 100.0.0.1, Dest Port : 9991, Src Port : 42931, State : UP
 Source Interface: Loopback0, MD5 Disabled
 Hello Timer 30 sec, Dead Timer 120 sec, Next Hello 00:00:19
 Uptime 2d05h (17:10:18 UTC Jan 15 2021)
 Service Buffer: Enabled

Service Announcement:
 Filter: DNAC-CONTROLLER-POLICY
 Count 50, Delay Timer 30 sec, Pending Announcement 0, Pending Withdraw 0
 Total Export Count 0, Next Export in 00:00:19

Service Query:
 Query Suppression Enabled
 Query Count 50, Query Delay Timer 15 sec, Pending 0
 Total Query Count 17093, Next Query in 00:00:19
```

Verifying Cisco Catalyst Center Configuration and Service-Routing Status

Verify Cisco DNA-Center configuration and service-routing status.

The Cisco Wide Area Bonjour application supports comprehensive assurance capabilities to manage service-routing with network-wide distributed Cisco Catalyst switches in SDG-Agent role and mDNS services discovered over Wide Area Bonjour domain. The assurance capabilities in Cisco Wide Area Bonjour provides ability to determine service-routing state, mDNS service state and many more information at various levels for day-2 operations, analysis and troubleshooting. Each category serves unique function to manage and troubleshoot Wide Area Bonjour service-routing for day-2 operation.

This sub-section provides brief overview for each category of monitor function:

- **Dashboard**: The landing page of Cisco Wide Area Bonjour application provides key statistics in various formats to quickly determine service-routing health across the network. The network administrator can monitor operational status of service-routing with SDG Agent devices, historical chart of service discovery request, processing and drops from network-wide distributed devices and top five talkers across the network.
- **Sub-Domain 360°** : The network administrator can briefly collect statistics and status counts in 360° view. The left-panel monitoring, and configuration bar is automatically open upon clicking selected sub-domain to verify configured policies, discovered service-instances on per sub domain basis of the configuration section.
- **Monitor**: A comprehensive 3-tier monitoring and troubleshooting function of Cisco Wide Area Bonjour application for various day-2 operations. The detail view of SDG Agent, Service-Instance and advanced Troubleshooting capabilities allows network administrator to manage and troubleshoot Wide Area Bonjour domain with single of glass on Cisco Catalyst Center.

For more information, see Cisco Wide Area Bonjour on Cisco Catalyst Center User Guide, Release 2.3.3 guide. The assurance capabilities and operation details are explained in Monitor the Cisco Wide Area Bonjour Application chapter to manage Cisco Wide Area Bonjour application with various supporting service-routing assurance function.

Reference

Reference for Bonjour multilayer wireless and wired.

Table 208:

Related Topic	Document Title
DNA Service for Bonjour Deployment on Cisco Catalyst 9600 Switch	Cisco Catalyst 9600 Series Switch Software Configuration Guide, Release 17.5.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9500 Switch	Cisco Catalyst 9500 Series Switch Software Configuration Guide, Release 17.5.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9400 Switch	Cisco Catalyst 9400 Series Switch Software Configuration Guide, Release 17.5.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9300 Switch	Cisco Catalyst 9300 Series Switch Software Configuration Guide, Release 17.5.X
Cisco Wide Area Bonjour Application on Cisco Catalyst Center User Guide	Cisco Wide Area Bonjour Application on Cisco Catalyst Center User Guide, Release 2.2.2

234 Configuration Example for FlexConnect Mode - Wireless and Wired

Topics:

- Overview

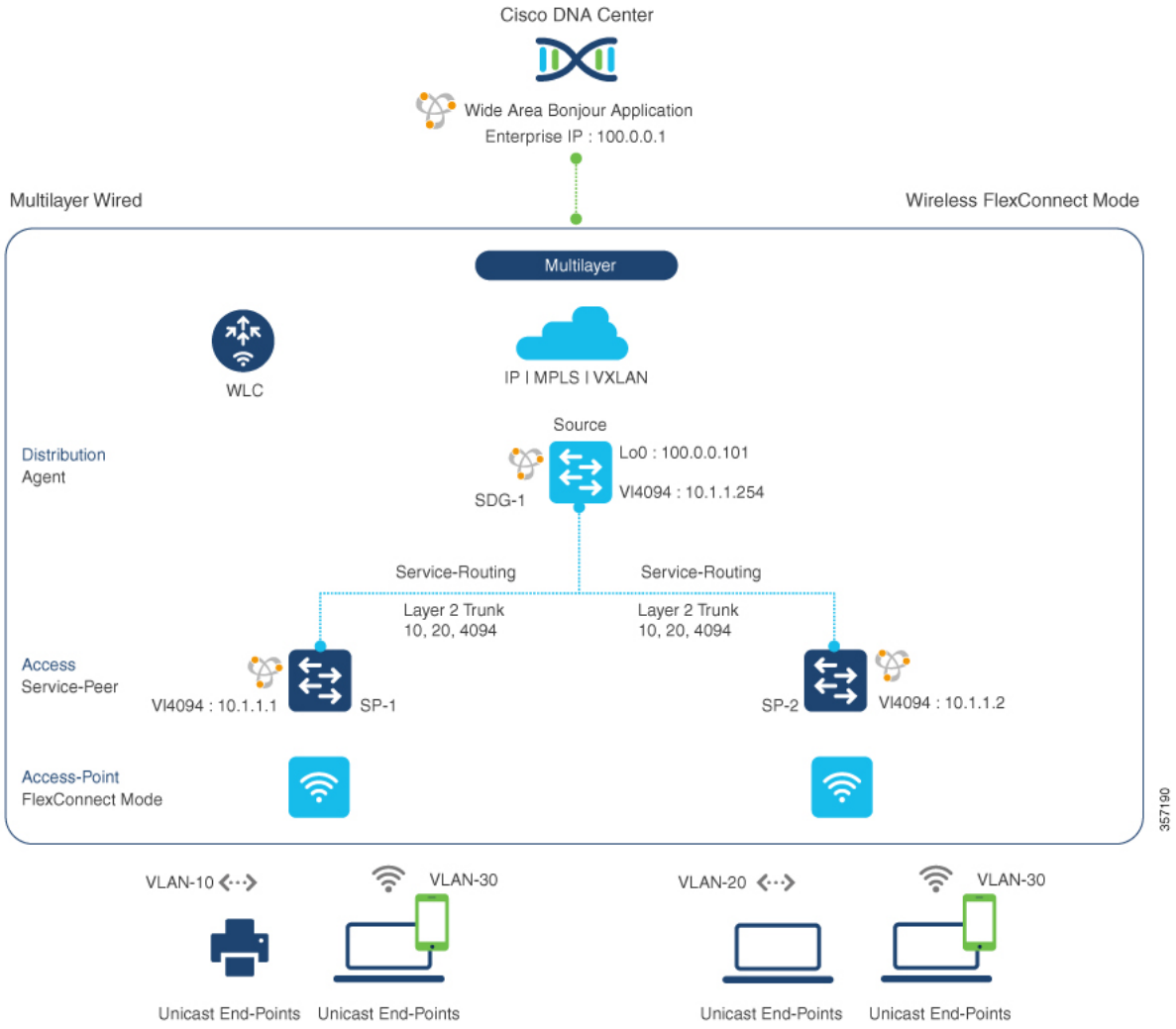
Overview

Introduces FlexConnect mode fundamentals for wireless and wired scenarios, presenting default and customized service list and policy configurations, detailed examples for access and distribution layers, traditional multilayer integration steps, verification commands, and supporting references to streamline deployment and management.

This chapter provides configuration guidelines to implement Local Area Bonjour enabling end-to-end policy-based mDNS service discovery and distribution across multilayer wired and wireless FlexConnect local-switching mode. The first hop mDNS gateway at Layer 2 access switch must be implemented in service peer mode and paired with common distribution-layer switch in SDG agent role IP gateway function to wired and wireless clients. The network-wide distributed SDG agent can be paired alternatively with the Cisco Catalyst Center to enable mDNS service-routing across IP core network providing mDNS service assurance, monitoring and troubleshooting.

The following figure illustrates unicast mode bonjour network environment with AirPrint capable printer and wireless user computer (macOS, Microsoft Windows, and so on.) connected to the same Ethernet switch. The network administrator implements the policy permitting additional endpoints associated to nearby location Ethernet switch to discover and use remote AirPrint capable Printer without flooding mDNS over wired and wireless networks.

Figure 68: Local Area Bonjour Service-Routing Multilayer Wired and Wireless FlexConnect Local-Switching Mode



Configuration Example for Default Service List and Policy in FlexConnect Mode - Wireless and Wired

Configuration example for default service list and policy in FlexConnect mode - wireless and wired.

This section provides guidance on configuring Service-Peer, SDG Agent, and Cisco Catalyst Center, allowing the wired and wireless endpoints to dynamically discover the default service list using Layer 2 unicast and policy.

Example: Wired and Wireless Access Layer Service Peer Configuration

Example for wired and wireless access layer service peer configuration.

The following table provides a sample configuration of wired and wireless controller access layer service peer.

Table 209: Configuring Wired and Wireless Access Layer Service Peer

Configuration Step	Sample Configuration: SP-1 Service-Peer Configuration	Sample Configuration: SP-2 Service-Peer Configuration
Step-1: Enable mDNS gateway and set the gateway mode.	```! mdns-sd gateway mode service-peer !```	```! mdns-sd gateway mode service-peer !```
Step-2: Create unique mDNS inbound policy to permit ingress AirPrint service announcement and query on the Catalyst Switch in service peer mode.	```! mdns-sd service-list LOCAL-AREA-SERVICES-IN in match printer-ipp !```	```! mdns-sd service-list LOCAL-AREA-SERVICES-IN in match printer-ipp !```
Step-3: Create unique mDNS outbound policy to permit egress AirPrint service response on the Catalyst Switch in service peer mode	```! mdns-sd service-list LOCAL-AREA-SERVICES-OUT out match printer-ipp !```	```! mdns-sd service-list LOCAL-AREA-SERVICES-OUT out match printer-ipp !```
Step-4: Associate inbound and outbound service list to a unique service policy.	```! mdns-sd service-policy LOCAL-AREA-POLICY service-list LOCAL-AREA-SERVICES-IN service-list LOCAL-AREA-SERVICES-OUT !```	```mdns-sd service-policy LOCAL-AREA-POLICY service-list LOCAL-AREA-SERVICES-IN service-list LOCAL-AREA-SERVICES-OUT !```
Step-5: Activate unicast mDNS gateway and attach service policy on wired VLAN and wireless FlexConnect user VLAN of SP-1 and SP-2 Layer 2 access switch.	```! vlan configuration 10, 30 mdns-sd gateway service-policy LOCAL-AREA-POLICY active-query timer 3600 !```	```! vlan configuration 20, 30 mdns-sd gateway service-policy LOCAL-AREA-POLICY active-query timer 3600 !```

Configuration Step	Sample Configuration: SP-1 Service-Peer Configuration	Sample Configuration: SP-2 Service-Peer Configuration
Step-6: Enable service routing on wired service peer mDNS between mDNS source and receiver local VLANs. **Note** This step is optional for SP-2 switch as it does not have local mDNS service provider endpoints or VLANs.	``` ! mdns-sd location-filter LOCAL-PROXY match location-group default vlan 10 match location-group default vlan 30 ! mdns-sd service-list LOCAL-AREA-SERVICES-OUT OUT match printer-ipps location-filter LOCAL-PROXY ! ```	

Step-7: Enable unicast service routing between wired and wireless service peer and SDG agent using wired management source VLAN ID and IP address.	```
	vlan configuration 10,
	30
	 mdns-sd gateway
	  source-interface vlan
	 4094
	  sdg-agent 10.1.1.254
	!
	``
		``
		!
		vlan configuration 20,
		30
		 mdns-sd gateway
		  source-interface vlan
		 4094
		  sdg-agent 10.1.1.254
		!
		``

Example: Wired and Wireless Distribution Layer SDG Agent Configuration

Example for wired and wireless distribution layer SDG agent configuration.

The following table provides a sample configuration of distribution layer SDG agent.

Table 210: Configuring Wired and Wireless Distribution Layer SDG Agent

Configuration Step	Sample Configuration: SDG-1 – SDG Agent
Step-1: Enable mDNS gateway and set the gateway mode. The default mode is sdg-agent.	`! mdns-sd gateway !`
Step-2: Create a unique mDNS inbound policy to permit ingress AirPrint service announcement and query the Catalyst Switch in Service-Peer mode.	`! mdns-sd service-list LOCAL-AREA-SERVICES-IN in  match printer-ipp !`
Step-3: Create a unique mDNS outbound policy to permit egress AirPrint service response on Catalyst Switch in Service-Peer mode.	`! mdns-sd service-list LOCAL-AREA-SERVICES-OUT out  match printer-ipp !`
Step-4: Associate the inbound and outbound service-list to a unique service-policy.	`! mdns-sd service-policy LOCAL-AREA-POLICY  service-list LOCAL-AREA-SERVICES-IN  service-list LOCAL-AREA-SERVICES-OUT !`
Step-5: Activate unicast mDNS gateway on wired VLAN and wireless user VLAN on SDG agents.	`! vlan configuration 10, 20, 30  mdns-sd gateway !`
Step-6: Configure the service peer-group and attach service-policy on the SDG agent distribution switch and enable service-routing between the assigned Service Peer switch group.	`! mdns-sd service-peer group  peer-group 1   service-policy LOCAL-AREA-POLICY   service-peer 10.1.1.1 location-group  default   service-peer 10.1.1.2 location-group  default !`
Step-7: Create a unique controller bound mDNS policy to permit egress AirPrint service discovery and distribution from Catalyst Switch in SDG agent mode. Inbound policy towards controller is not required.	`! mdns-sd service-list WIDE-AREA-SERVICES-OUT out  match printer-ipp !`

Configuration Step	Sample Configuration: SDG-1 – SDG Agent
Step-8: Associate outbound service-list to a unique service-policy.	``` ! mdns-sd service-policy WIDE-AREA-POLICY service-list WIDE-AREA-SERVICES-OUT ! ```
Step-9: Enable Wide Area Bonjour service-routing with service export configuration association controller IP Address, source interface for stateful connection, and mandatory egress policy for Wide Area service-routing.	``` ! service-export mdns-sd controller DNAC-CONTROLLER-POLICY controller-address 100.0.0.1 controller-source-interface LOOPBACK 0 controller-service-policy WIDE-AREA-POLICY ! ```

Configuration Example for Customized Service List and Policy in FlexConnect Mode - Wireless and Wired

Configuration example for customized service list and policy in FlexConnect mode - wireless and wired.

This section provides guidance on configuring Service-Peer, SDG Agent, and Cisco Catalyst Center, allowing the wired and wireless endpoints to dynamically discover printer using Layer 2 unicast and policy.

Example: Wired and Wireless Access Layer Service Peer Configuration

Example for wired and wireless access layer service peer configuration.

The following table provides a sample configuration of wired and wireless controller access layer service peer.

Table 211: Configuring Wired and Wireless Access Layer Service Peer

Configuration Step	Sample Configuration: SP-1 Service-Peer Configuration	Sample Configuration: SP-2 Service-Peer Configuration
Step-1: Enable mDNS gateway and set the gateway mode.	! mdns-sd gateway mode service-peer !	! mdns-sd gateway mode service-peer !
Step-2: Create unique mDNS inbound policy to permit ingress AirPrint service announcement and query on the Catalyst Switch in service peer mode.	! mdns-sd service-list LOCAL-AREA-SERVICES-IN in match printer-ipp !	! mdns-sd service-list LOCAL-AREA-SERVICES-IN in match printer-ipp !
Step-3: Create unique mDNS outbound policy to permit egress AirPrint service response on the Catalyst Switch in service peer mode	! mdns-sd service-list LOCAL-AREA-SERVICES-OUT out match printer-ipp !	! mdns-sd service-list LOCAL-AREA-SERVICES-OUT out match printer-ipp !
Step-4: Associate inbound and outbound service list to a unique service policy.	! mdns-sd service-policy LOCAL-AREA-POLICY service-list LOCAL-AREA-SERVICES-IN service-list LOCAL-AREA-SERVICES-OUT !	mdns-sd service-policy LOCAL-AREA-POLICY service-list LOCAL-AREA-SERVICES-IN service-list LOCAL-AREA-SERVICES-OUT !
Step-5: Activate unicast mDNS gateway and attach service policy on wired VLAN and wireless FlexConnect user VLAN of SP-1 and SP-2 Layer 2 access switch.	! vlan configuration 10, 30 mdns-sd gateway service-policy LOCAL-AREA-POLICY active-query timer 3600 !	! vlan configuration 20, 30 mdns-sd gateway service-policy LOCAL-AREA-POLICY active-query timer 3600 !

Configuration Step	Sample Configuration: SP-1 Service-Peer Configuration	Sample Configuration: SP-2 Service-Peer Configuration
Step-6: Enable service routing on wired service peer mDNS between mDNS source and receiver local VLANs. **Note** This step is optional for SP-2 switch as it does not have local mDNS service provider endpoints or VLANs.	``` ! mdns-sd location-filter LOCAL-PROXY match location-group default vlan 10 match location-group default vlan 30 ! mdns-sd service-list LOCAL-AREA-SERVICES-OUT OUT match printer-ipps location-filter LOCAL-PROXY ! ```	

Step-7: Enable unicast service routing between wired and wireless service peer and SDG agent using wired management source VLAN ID and IP address.	```
	vlan configuration 10,
	30
	 mdns-sd gateway
	  source-interface vlan
	 4094
	  sdg-agent 10.1.1.254
	!
	``
		``
		!
		vlan configuration 20,
		30
		 mdns-sd gateway
		  source-interface vlan
		 4094
		  sdg-agent 10.1.1.254
		!
		``

Example: Wired and Wireless Distribution Layer SDG Agent Configuration

Example for wired and wireless distribution layer SDG agent configuration.

The following table provides a sample configuration of distribution layer SDG agent.

Table 212: Configuring Wired and Wireles Distribution Layer SDG Agent

Configuration Step	Sample Configuration: SDG-1 – SDG Agent
Step-1: Enable mDNS gateway and set the gateway mode. The default mode is sdg-agent.	`! mdns-sd gateway !`
Step-2: Create a unique mDNS inbound policy to permit ingress AirPrint service announcement and query the Catalyst Switch in Service-Peer mode.	`! mdns-sd service-list LOCAL-AREA-SERVICES-IN in  match printer-ipp !`
Step-3: Create a unique mDNS outbound policy to permit egress AirPrint service response on Catalyst Switch in Service-Peer mode.	`! mdns-sd service-list LOCAL-AREA-SERVICES-OUT out  match printer-ipp !`
Step-4: Associate the inbound and outbound service-list to a unique service-policy.	`! mdns-sd service-policy LOCAL-AREA-POLICY  service-list LOCAL-AREA-SERVICES-IN  service-list LOCAL-AREA-SERVICES-OUT !`
Step-5: Activate unicast mDNS gateway on wired VLAN and wireless user VLAN on SDG agents.	`! vlan configuration 10, 20, 30  mdns-sd gateway !`
Step-6: Configure the service peer-group and attach service-policy on the SDG agent distribution switch and enable service-routing between the assigned Service Peer switch group.	`! mdns-sd service-peer group  peer-group 1   service-policy LOCAL-AREA-POLICY   service-peer 10.1.1.1 location-group  default   service-peer 10.1.1.2 location-group  default !`
Step-7: Create a unique controller bound mDNS policy to permit egress AirPrint service discovery and distribution from Catalyst Switch in SDG agent mode. Inbound policy towards controller is not required.	`! mdns-sd service-list WIDE-AREA-SERVICES-OUT out  match printer-ipp !`

Configuration Step	Sample Configuration: SDG-1 – SDG Agent
Step-8: Associate outbound service-list to a unique service-policy.	```! mdns-sd service-policy WIDE-AREA-POLICY service-list WIDE-AREA-SERVICES-OUT !```
Step-9: Enable Wide Area Bonjour service-routing with service export configuration association controller IP Address, source interface for stateful connection, and mandatory egress policy for Wide Area service-routing.	```! service-export mdns-sd controller DNAC-CONTROLLER-POLICY controller-address 100.0.0.1 controller-source-interface LOOPBACK 0 controller-service-policy WIDE-AREA-POLICY !```

Cisco Catalyst Center Traditional Multilayer Wired and Wireless Configuration

Cisco DNA Center traditional multilayer wired and wireless configuration.

Configuring Service Filters for Traditional Multilayer Wired and Wireless FlexConnect Local- Switching Mode (GUI)

Configure service filters for traditional multilayer wired and wireless FlexConnect local- switching mode using GUI.

This procedure implements global service filters, which permit the Cisco Wide Area Bonjour application to dynamically discover and distribute service information between trusted Cisco Catalyst SDG agent switches across the IP network.

Procedure

1. Navigate to the **Configuration** tab in the Wide Area Bonjour application.
2. From the sidebar, select the sub-domain for which you want to create the service filter.
3. Check the **Service Filter** box.
4. Click **Service Filter** icon from the topology to view a list of the service filters for the selected domain. You can also manually edit existing service filters from this list.
5. Click **Create Service Filter**.
6. From the **Network Mode** drop-down list, choose **Traditional** (the default mode).
7. Enter a unique name for the service filter.
8. (Optional) Enter a description for the service filter.
9. Select one or more service types to permit announcements and queries.
10. Enable or disable service filters after creating them. By default, service filters are enabled.

Configuring Source SDG Agents in Traditional Multilayer Wired and Wireless FlexConnect Local- Switching Mode (GUI)

Configure source SDG agents in traditional multilayer wired and wireless FlexConnect local- switching mode using GUI.

This procedure configures discovery of wired printer sources from the LAN distribution switches paired with Layer 2 Catalyst Switches in a service peer role. The wireless distribution switches paired with a controller in a service peer role receive query responses for wired printers and distribute the responses to querying devices over the wireless FlexConnect local switching mode network.

Procedure

1. Click **Add** on the upper-right of Cisco Catalyst Center.
2. Click the **Source** radio button to select a source SDG agent. By default, this radio button is selected.
3. From the **SDG Agent/IP** drop-down list, select an SDG agent (100.0.0.101) which announces the services, for example, Printer.
4. Select **Peer** from the **Service Layer** drop-down list.
5. Uncheck the box **Any**. By default, this is unchecked.
6. Select the query **VLAN** (Vlan-10) to distribute services (Printer) from a specific network.
7. Enable or disable services from the selected query IPv4 subnet. By default, this is enabled.
8. Enable or disable services from the selected query IPv6 subnet. By default, this is enabled.
9. Enter the **service peer IPv4 address** (10.1.1.1).

> **Note**
>
> Select **Any** to accept services from any peer on a selected VLAN.

10. (Optional) Click **Add Next** to add more source SDG agents. (Repeat the preceding steps.)
11. Click **DONE**.
12. Click **CREATE**.

Configuring Query SDG Agents in Traditional Multilayer Wired and Wireless FlexConnect Local- Switching Mode (GUI)

Configure query SDG agents in traditional multilayer wired and wireless FlexConnect local- switching mode using GUI.

This procedure configures distributed services to query SDG agents connected to a controller in service peer mode, based on a policy.

f you are using wide area network, see the *Cisco Wide Area Bonjour on Cisco Catalyst Center User Guide* at https://www.cisco.com/c/en/us/support/cloud-systems-management/dna-center/products-user-guide-list.html.

Procedure

1. Click **Add** on the upper-right of Cisco Catalyst Center.
2. Select the **Query SDG agent** radio button. By default, the **Source** radio button is selected.
3. From the **SDG Agent/IP** drop-down list, select an SDG agent (100.0.0.102) that receives queries for the services (Printer).
4. Select **Peer** from the **Service Layer** drop-down list.
5. Uncheck the box **Any**. By default, this is enabled.
6. Select the query **VLAN** (Vlan-30) to distribute services (Printer) to a specific network.
7. Enable or disable services from the selected query IPv4 subnet. By default, this is enabled.
8. Enable or disable services from the selected query IPv6 subnet. By default, this is enabled.
9. Enter the **service peer IPv4 address** (10.2.1.254).
10. Click the **+** icon to add more service-peers, if any. Select **Any** to accept services from any peer on a selected VLAN.
11. (Optional) Click **Add Next** to add more query agents. (Repeat the preceding steps.)
12. Click **DONE**.

13. Click **CREATE**.

Verifying Configuration Example for FlexConnect Mode - Wireless and Wired

Verify configuration example for FlexConnect mode - wireless and wired.

This section provides step-by-step mDNS configuration and service discovery and distribution status based on applied policy on Wired Layer 2 access switch in service peer and SDG agent mode.

Verifying Wired Service-Peer Configuration

Verify wired service-peer configuration.

Use the following commands on the Cisco Catalyst switch in service peer (SP-1 and SP-2) mode to determine the operational status after applying configuration and discovering the AirPrint service from the local network.

```
Device# show mdns-sd summary vlan 10

VLAN: 10
========================================
mDNS Gateway: Enabled
mDNS Service Policy: LOCAL-AREA-POLICY
Active Query: Enabled
            : Periodicity 3600 Seconds
Transport Type: IPv4
Service Instance Suffix: Not Configured
mDNS Query Type: ALL
SDG Agent IP: 10.1.1.254
Source Interface: Vlan4094

Device# show mdns-sd service-policy name LOCAL-AREA-POLICY

Service Policy Name     Service List IN Name     Service List Out Name
=============================================================================
LOCAL-AREA-POLICY       LOCAL-AREA-SERVICES-IN    LOCAL-AREA-SERVICES-OUT

Device# show mdns-sd cache vlan 10
```

Name	Type	TTL/ Remaining	Vlan-Id/ Interface-name	MAC Address	RR Record Data
_universal. _sub. _ipp. _tcp.local	PTR	4500/4486	Vl10	ac18.2651.03fe	Bldg-1-FL1-PRN. _ipp. _tcp.local
_ipp. _tcp.local	PTR	4500/4486	Vl10	ac18.2651.03fe	Bldg-1-FL1-PRN. _ipp. _tcp.local
Bldg-1-FL1-PRN. _ipp. _tcp.local	SRV	4500/4486	Vl10	ac18.2651.03fe	0 0 631 Bldg-1-FL1-PRN.local
Bldg-1-FL1-PRN.local	A	4500/4486	Vl10	ac18.2651.03fe	10.153.1.1
Bldg-1-FL1-PRN.local	AAAA	4500/4486	Vl10	ac18.2651.03fe	2001:10:153: 1:79:A40C:6BEE: AE

Name	Type	TTL/ Remaining	Vlan-Id/ Interface-name	MAC Address	RR Record Data
Bldg-1-FL1-PRN. _ipp. _tcp.local	TXT	4500/4486	Vl10	ac18.2651.03fe	(451)'txtvers=1''priority=30'' ty=EPSON WF-3620 Series'' usb_MFG=EPSON'' usb_MDL=W~'~

```
Device# show mdns-sd statistics vlan 10

 mDNS Statistics

 Vl10:
  mDNS packets sent              : 612
     IPv4 sent                   : 612
       IPv4 advertisements sent  : 0
       IPv4 queries sent         : 612
     IPv6 sent                   : 0
       IPv6 advertisements sent  : 0
       IPv6 queries sent         : 0
  Unicast sent                   : 0
  mDNS packets rate limited      : 0
  mDNS packets received          : 42
   advertisements received       : 28
   queries received              : 14
     IPv4 received               : 42
       IPv4 advertisements received: 28
       IPv4 queries received       : 14
     IPv6 received                 : 0
       IPv6 advertisements received: 0
       IPv6 queries received       : 0
  mDNS packets dropped           : 0
  ========================================
  Query Type                       : Count
  ========================================
  PTR                              : 12
  SRV                              : 0
  A                                : 0
  AAAA                             : 0
  TXT                              : 0
  ANY                              : 3
  ================================================
  PTR Name                 Advertisement     Query
  ================================================
  _ipp. _tcp.local           9          4
```

Verifying Wired SDG Agent Configuration and Service-Routing Status

Verify wired SDG agent configuration and service-routing status.

This section provides information on mDNS configuration and service-routing on Wired and Wireless SDG Agent (SDG-1) with locally attached Layer 2 access switches in Service-Peer (SP-1 and SP-2) mode and with centrally paired Cisco Catalyst Center for Wide Area Bonjour service-routing.

```
Device# show mdns-sd summary vlan 10
```

```
VLAN: 10
========================================
mDNS Gateway                : Enabled
mDNS Service Policy         : LOCAL-AREA-POLICY
Active Query                : Disabled
Transport Type              : IPv4
Service Instance Suffix     : Not Configured
mDNS Query Type             : ALL
SDG Agent IP                : Not-Configured
Source Interface            : Not-Configured
```

```
Device# show mdns-sd cache vlan 10
```

Name	Type	TTL/ Remaining	Vlan-Id /Interface-name	MAC Address	RR Record Data
_universal. _sub. _ipp. _tcp.local	PTR	4500/4500	Vl10	ac18.2651.03fe	Bldg-1-FL1-PRN. _ipp. _tcp.local
_ipp._tcp.local	PTR	4500/4500	Vl10	ac18.2651.03fe	Bldg-1-FL1-PRN. _ipp. _tcp.local
Bldg-1-FL1- PRN. _ipp._tcp.local	SRV	4500/4500	Vl10	ac18.2651.03fe	0 0 631 Bldg-1-FL1-PRN.local
Bldg-1-FL1 -PRN.local	A	4500/4500	Vl10	ac18.2651.03fe	10.153.1.1
Bldg-1-FL1- PRN.local	AAAA	4500/4500	Vl10	ac18.2651.03fe	2001:10:153: 1:79 A40C:6BEE: AEEC
Bldg-1-FL1-PRN. _ipp._tcp.local	TXT	4500/4500	Vl10	ac18.2651.03fe	(451)txtvers=1"priority=30" ty=EPSON WF-3620 Series'' usb_MFG=EPSON usb_MDL=W'~

```
Device# show mdns-sd sp-sdg statistics

                                  One min, 5 mins, 1 hour
Average Input rate (pps)       : 0,       0,       0
Average Output rate (pps)      : 0,       0,       0
 Messages received:
  Query                       : 15796
  ANY query                   : 0
  Advertisements              : 28
  Advertisement Withdraw      : 0
  Interface down              : 0
  Vlan down                   : 0
  Service-peer ID change      : 0
  Service-peer cache clear    : 12
```

```
  Resync response             : 6
 Messages sent:
  Query response              : 5975
  ANY Query response          : 0
  Cache-sync                  : 61
  Get service-instance        : 0

Device# show mdns-sd controller detail

Controller: DNAC-Policy
 IP: 100.0.0.1, Dest Port : 9991, Src Port : 42446, State : UP
 Source Interface: Loopback0, MD5 Disabled
 Hello Timer 30 sec, Dead Timer 120 sec, Next Hello 00:00:24
 Uptime 2d05h (17:02:37 UTC Jan 15 2021)
 Service Buffer: Enabled

Service Announcement:
 Filter: DNAC-CONTROLLER-POLICY
 Count 50, Delay Timer 30 sec, Pending Announcement 0, Pending Withdraw 0
 Total Export Count 56, Next Export in 00:00:24

Service Query:
 Query Suppression Enabled
 Query Count 50, Query Delay Timer 15 sec, Pending 0
 Total Query Count 15791, Next Query in 00:00:09
```

Verifying Cisco Catalyst Center Configuration and Service Routing Status

Verify Cisco DNA Center configuration and service routing status.

The Cisco Wide Area Bonjour application supports comprehensive assurance capabilities to manage service routing with network-wide distributed Cisco Catalyst switches in SDG Agent role and mDNS services discovered over Wide Area Bonjour domain. The assurance capabilities in Cisco Wide Area Bonjour provides the ability to determine service routing state, mDNS service state, and many more information at various levels for day-2 operations, analysis and troubleshooting. Each category serves unique function to manage and troubleshoot Wide Area Bonjour service routing for day-2 operation.

This sub-section provides brief overview for each category of monitor function:

- **Dashboard**: The landing page of Cisco Wide Area Bonjour application provides key statistics in various formats to quickly determine service routing health across the network. The network administrator can monitor operational status of service routing with SDG Agent devices, historical chart of service discovery request, processing and drops from network-wide distributed devices and top five talkers across the network.
- **Sub-Domain 360°** : The network administrator can briefly collect statistics and status counts in 360° view. The left-panel monitoring, and configuration bar is automatically open upon clicking selected sub-domain to verify configured policies, discovered service-instances on per sub-domain basis of the configuration section.
- **Monitor**: A comprehensive 3-tier monitoring and troubleshooting function of Cisco Wide Area Bonjour application for various day-2 operations. The detail view of SDG Agent, Service-Instance, and advanced Troubleshooting capabilities allows network administrator to manage and troubleshoot Wide Area Bonjour domain with a single pane of glass on Cisco Catalyst Center.

For more information, see Cisco Wide Area Bonjour on Cisco Catalyst Center User Guide, Release 2.1.2 guide. The assurance capabilities and operation details are explained in **Monitor the Cisco Wide Area Bonjour Application** chapter to manage Cisco Wide Area Bonjour application with various supporting service routing assurance function.

Reference

Reference for DNA service for Bonjour deployment and application.

Table 213:

Related Topic	Document Title
DNA Service for Bonjour Deployment on Cisco Catalyst 9600 Switch	Cisco Catalyst 9600 Series Switch Software Configuration Guide, Release 17.5.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9500 Switch	Cisco Catalyst 9500 Series Switch Software Configuration Guide, Release 17.5.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9400 Switch	Cisco Catalyst 9400 Series Switch Software Configuration Guide, Release 17.5.X
DNA Service for Bonjour Deployment on Cisco Catalyst 9300 Switch	Cisco Catalyst 9300 Series Switch Software Configuration Guide, Release 17.5.X
Cisco Wide Area Bonjour Application on Cisco Catalyst Center User Guide	Cisco Wide Area Bonjour Application on Cisco Catalyst Center User Guide, Release 2.2.2

Part XXIII

Multicast Domain Name System

Topics:

- Multicast Domain Name System

235 Multicast Domain Name System

Topics:

- mDNS Gateway
- Location based service filtering
- Nearest mDNS based wired service filtering

This chapter explains how to configure and manage mDNS gateway and AP features on Cisco controllers to enable cross-subnet Bonjour service discovery, apply location- and policy-based filtering, and verify/debug wired and wireless mDNS behavior.

mDNS Gateway

Explains how mDNS gateways enable service discovery across different network segments by bridging multicast DNS traffic, facilitating interoperability in large or segmented networks.

A mDNS gateway is an Apple service discovery protocol that

- listens for multicast DNS (mDNS) service announcements and queries from local domains with the use of mDNS service records
- caches and forwards mDNS advertisements between segmented Layer 2 networks or subnets, and
- enables devices and services to be discovered even when clients and sources reside in different network segments.

The Bonjour protocol operates using service announcements and queries. Each query or advertisement is sent to the Bonjour multicast address IPv4 224.0.0.251 or IPv6 FF02::FB. The protocol uses mDNS on UDP port 5353.

The address used by the Bonjour protocol is a link-local multicast address and is therefore only forwarded to the local Layer 2 network. Because multicast DNS is limited to a Layer 2 domain, a client must be part of the same Layer 2 domain to discover a service. However, this requirement is not always practical in large-scale deployments or enterprises.

To address this issue, the Cisco Catalyst 9800 Series Wireless Controller acts as a Bonjour Gateway. The controller then listens for Bonjour services and caches these Bonjour advertisements (AirPlay, AirPrint, and so on) from the source or host. For example, Apple TV responds to Bonjour clients when requested for a service. This allows sources and clients to exist in different subnets.

By default, the mDNS gateway is disabled on the controller. To enable mDNS gateway functionality, you must explicitly configure the mDNS gateway using the CLI or Web UI.

The source IP address of all outgoing mDNS packets uses the mDNS source interface VLAN SVI IP address. By default, wireless management interface will be the source interface.

Guidelines and restrictions for configuring mDNS AP

This topic describes how configuring mDNS APs involves understanding both operational guidelines and inherent restrictions to ensure scalability, reliability, and optimal network performance.

Guidelines for configuring mDNS AP

- Cisco recommends deploying scalable Wide Area Bonjour to route mDNS service between wired and wireless networks. Cisco Catalyst 9800 Series Wireless LAN Controller introduces a new mDNS gateway, called Service-Peer mode, which replaces the classic mDNS flood-n-learn and supports enterprise-grade, scalable, stateful, and reliable complete unicast-based mDNS service-routing with upstream gateway Cisco Catalyst 9000 Series Switches. For more information, see Part: Cisco DNA Service for Bonjour .
- The mDNS AP (classic flood-and-learn based feature) is enhanced with complete unicast-based service-routing using Cisco Wide Area Bonjour. This enhancement supports flood-free wired and wireless networks and helps overcome several operational, scalability, and service resiliency challenges. The mDNS AP extends the mDNS flood from Wired VLANs to the AP. It further extends this traffic over the CAPWAP tunnel to the controller for central processing across the core network. Cisco recommends using the mDNS AP only for small network environments.
- The wired mDNS service-provider VLANs must be extended to flood mDNS traffic up to the mDNS AP Ethernet port in trunk mode. The wired VLAN extension to the mDNS AP may include other wired flood traffic, such as broadcast, unknown unicast, and Layer 2 multicast. This additional traffic can affect the mDNS AP's scale and performance.
- It is recommended to deploy at least one mDNS AP for each Layer 3 Access switch. All wired mDNS traffic is flooded using alternate L2 methods if a single mDNS AP is shared between multiple Layer 3 Access switches.
- The old wired mDNS service entry continues to be advertised to all wireless users for up to 4500 seconds, based on the mDNS cache timers on the controller. Stale entries require manual clearing from the local cache in the controller.

-
- Wireless multicast link-local is enabled by default. When wireless link-local is enabled, only mDNS Bridging mode is supported. If you require mDNS Gateway for wired services, disable wireless link-local.
- The controller sends mDNS queries with the "QU" (Query Unicast) flag at regular intervals. This flag instructs mDNS service providers (mDNS-SP) to respond with an mDNS advertisement via unicast reply. Beginning with the Cisco Catalyst IOS XE 17.9.x release, these queries are generated from the WMI interface. This change eliminates the need for interface VLANs where Service Providers are connected.
- If your Service Provider honors the "QU" flag and sends a unicast reply to the controller, you may see the mDNS-AP entries on your controller change to wired entries. You see this change because the controller receives the advertisement from the wired infrastructure. If the Service Provider does not honor the "QU" flag, you will not encounter this issue, such as on Apple devices.

> **Note**
>
> RFC 6762 compliance: RFC 6762 states that the service provider may choose not to respond to queries from the WMI, because these queries are outside the service provider's link-local scope. The controller is unable to learn about such service providers due to this limitation.

Restrictions for configuring mDNS AP

- The mDNS AP is supported only in Local and Monitor modes. If the Cisco Wireless AP operates in FlexConnect mode, the Fabric mode AP does not support the mDNS AP feature. For information about enabling mDNS service routing for various distributed wireless modes, refer to Part: Cisco DNA Service for Bonjour. If you use a FlexConnect AP as an mDNS gateway, do not use a period (.) in the service provider name, as this is not supported.
- Wireless users connected to mDNS AP may not be able to browse the Wired mDNS services across flooded Wired VLAN to mDNS AP.
- The mDNS AP scale limit for each Cisco Catalyst 9800 Series Wireless LAN Controller is restricted. The maximum mDNS Wired VLAN count for each controller is limited. The mDNS AP does not support mDNS query packet suppression or rate-limiter in AP. The wired mDNS flood from all wired VLANs is forwarded to the controller for central policy enforcement.
- The maximum number of flooded packets processed per second from wired VLANs to the mDNS AP is limited. The performance and reliability of the mDNS AP may decrease in large network environments. A maximum of 10 Wired VLANs' mDNS flood can be extended to mDNS AP. A combination of large wired VLAN scale and mDNS AP scale may impact performance in both the AP and controller.
- Only one mDNS AP is supported for each wired VLAN. Configuring multiple mDNS APs to map the same wired VLAN ID causes service instability and duplicate processing. Only one wired mDNS service policy is supported for all network-wide mDNS APs.
- High availability is not supported for multiple mDNS APs. mDNS services across wired and wireless networks are disrupted when connectivity to an mDNS AP is lost due to any type of failure.
- These limitations apply when an mDNS AP introduces LSS-based mDNS service filtering between flooded wired VLANs and wireless networks:
  - A single mDNS AP with LSS enabled can distribute wired mDNS services only to nearby APs in its neighbor list. Wireless users connected to APs not in the neighbor list may not be able to discover any wired mDNS services.
  - Only one mDNS AP can be deployed in each Wired VLAN. Wired VLANs must be reconfigured across the LAN to enable unique LSS-based mDNS APs in specific locations. For example, to achieve mDNS service discovery on each floor, each floor must have a dedicated wired VLAN or subnet, with one mDNS AP per floor to discover all other APs as neighbors on the same floor.

- The mDNS AP does not support IPv6 for wired mDNS service providers or service receivers. Only IPv4 is supported. The mDNS AP does not support role-based mDNS service filtering between wired and wireless networks. The mDNS AP does not detect or automatically resolve duplicate mDNS service instance names across wired VLANs. The Cisco Catalyst 9800 Series Wireless LAN Controller discovers and records the first service instance with a unique name in its local cache database. If a duplicate service instance name is discovered, the controller rejects the duplicate name and does not distribute it to wireless clients.
- In mDNS gateway mode, the controller does not support service discovery from mDNS messages that use multiple IP fragments.

Enable mDNS Gateway (GUI)

Learn how to enable the mDNS gateway using the GUI interface.

Enable or disable the mDNS gateway for service discovery using the GUI.

Procedure

1. Choose **Configuration** > **Services** > **mDNS**.
2. In the **Global** section, toggle the slider to enable or disable the **mDNS Gateway**.
3. From the **Transport** drop-down list, select one of these types:
   - **ipv4**
   - **ipv6**
   - **both**
4. Enter an appropriate timer value in **Active-Query Timer**. The valid range is between 1 to 120 minutes. The default is 30 minutes.
5. From the **mDNS-AP Service Policy** drop-down list, select an mDNS service policy.

   Note

   Service policy is optional only if mDNS-AP is configured. If mDNS-AP is not configured, the system uses default-service-policy.

6. Click **Apply**.

Enable or disable mDNS Gateway (GUI)

Follow these steps to enable or disable the mDNS Gateway using the GUI.

Use this procedure to configure or change mDNS service discovery for your device using the GUI.

Procedure

1. Choose **Configuration** > **Services** > **mDNS** > **Global**.
2. Enable or disable the **mDNS Gateway** toggle button.
3. Select **ipv4** or **ipv6** or **both** from the **Transport** drop-down list.

4. Enter the **Active-Query Timer**.
5. Click **Apply**.

Enable or disable mDNS Gateway (CLI)

Configure the mDNS gateway feature using commands to enable or disable service discovery on your controller

Enable or disable mDNS gateway to control multicast DNS service discovery for wireless devices using commands.

Note

- mDNS gateway is disabled by default globally on the controller.
- You need both global and WLAN configurations to enable mDNS gateway.

Procedure

1. Enable the privileged exec mode.

   Example

   ```
   Device> enable
   ```

   Enter your password, if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Enable mDNS gateway.

   Example

   ```
   Device(config)# mdns-sd gateway
   ```

4. Filter the mDNS gateway based on location.

   Example

   ```
   Device(config-mdns-sd)# location {ap-location | ap-name | location-group |
   lss | regex | site-tag | ssid}
   ```

   Here,

   - **ap-location** signifies location-based filtering using AP location.
   - **ap-name** signifies location-based filtering using AP name.
   - **location-group** signifies location-based filtering using location group.

- **lss** signifies location-based filtering using Location Specific Services (LSS).
- **regex** signifies location-based filtering using Regular Expression.
- **site-tag** signifies location-based filtering using site tag.
- **ssid** signifies location-based filtering using SSID.

> **Note**
>
> The **lss** is the default location filter, if mDNS gateway is configured globally.

5. Process mDNS message on a specific transport.

   Example

   ```
   Device(config-mdns-sd)# transport {ipv4 | ipv6 | both}
   ```

   Here,

   ipv4 signifies that the IPv4 mDNS message processing is enabled. This is the default value.

   ipv6 signifies that the IPv6 mDNS message processing is enabled.

   both signifies that the IPv4 and IPv6 mDNS message is enabled for each network.

6. Change the periodicity of mDNS multicast active query.

   Example

   ```
   Device(config-mdns-sd)# active-query timer active-query-periodicity
   ```

   > **Note**
   >
   > An active query is a periodic mDNS query to refresh dynamic cache.

   Here, *active-query-periodicity* refers to the active query periodicity in minutes. The valid range is from 1 to 120 minutes. Active query runs with a default periodicity of 30 minutes.

7. Configure the source interface to communicate between SDG agent and service peer.

   Example

   ```
   Device(config-mdns-sd)# source-interface vlan vlan-id
   ```

   By default, wireless management interface is used. The interface that you configure is used for all mDNS transactions.

8. Return to the global configuration mode.

Example

```
Device(config-mdns-sd)# exit
```

Create default service policy

This topic outlines the default mDNS service policy behavior and guides you in overriding it with a custom service policy if needed.

Outlines the default mDNS service policy behavior and guides you in overriding it with a custom service policy if needed.

When the mDNS gateway is enabled on any of the WLANs by default, mdns-default-service-policy is associated with it. Default service policy consists of default-service-list and their details are explained in this section. You can override the default service policy with a custom service policy.

Procedure

1. Create a service-definition if the service is not listed in the preconfigured services.
2. Create a service list for IN and OUT by using the service-definitions.
3. Use the existing service list to create a new service. For more information, refer to *Creating Service Policy* section.
4. Attach the *mdns-service-policy* to the profile or VLAN that needs to be enforced.
5. To check the default-mdns-service list, use this command:
   show mdns-sd default-service-list

Create custom service definition (GUI)

Configure a custom service definition using the GUI.

Create a service definition that specifies custom settings for mDNS through the GUI for device configuration.

Procedure

1. Choose **Configuration** > **Services** > **mDNS**.
2. In the **Service Definition** section, click **Add**.
3. In the **Quick Setup: Service Definition** page that is displayed, enter a name and description for the service definition.
4. Enter a service type and click **+** to add the service type.
5. Click **Apply to Device**.

Create custom service definition (CLI)

Configure a custom mDNS service definition using commands.

Add a custom service definition for mDNS, allowing you to associate user-friendly names with service types or PTR resource records using commands.

A service definition is a construct that provides an admin-friendly name to one or more mDNS service types or to a pointer (PTR) resource record name.

By default, a few built-in service definitions are already available for the admin to use.

In addition to built-in service definitions, the admin can define custom service definitions.

You can execute this command to view all service definitions (built-in and custom):

```
Device# show mdns-sd master-service-list
```

Procedure

1. Enable the privileged exec mode.

   Example

   ```
   Device> enable
   ```

   Enter your password, if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Configure mDNS service definition.

   Example

   ```
   Device(config)# mdns-sd service-definition service-definition-name
   ```

   Note

   - All the created custom service definitions are added to the primary service list.
   - Primary service list comprises of a list of custom and built-in service definitions.

4. Configure mDNS service type.

   Example

   ```
   Device(config-mdns-ser-def)# service-type string
   ```

5. Return to the global configuration mode.

   Example

   ```
   Device(config-mdns-ser-def)# exit
   ```

Create service list (GUI)

Use this task to configure a service list for mDNS filtering in the GUI, specifying direction, services, and message types, and applying the configuration to the device.

Define and manage which mDNS services and message types are allowed or filtered for inbound or outbound traffic using the GUI.

Procedure

1. Choose **Configuration** > **Services** > **mDNS**.
2. In the **Service List** section, click **Add**.
3. In the **Quick Setup: Service List** page that is displayed, enter a name for the service list.
4. From the **Direction** drop-down list, select **IN** for inbound filtering or **OUT** for outbound filtering.
5. From the **Available Services** drop-down list, select a service type to match the service list.

   Note

   To allow all services, select the **all** option.

6. Click **Add Services**.
7. From the **Message Type** drop-down list, select the message type to match from the following options:
   - **any**: To allow all messages.
   - **announcement**: To allow only service advertisements or announcements for the device.
   - **query**: To allow only a query from the client for a service in the network.
8. Click **Save** to add services.
9. Click **Apply to Device**.

Create service list (CLI)

Configure a custom mDNS service list to filter service announcements or queries for use with mDNS service policies using commands.

Create an mDNS service list in Cisco IOS XE to filter mDNS service announcements or queries for inbound or outbound traffic using commands.

mDNS service list is a collection of service definitions.

Procedure

1. Enable the privileged exec mode.

   Example

```
Device> enable
```

Enter your password, if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Configure mDNS service list.

   Example

   ```
   Device(config)# mdns-sd service-list service-list-name {IN | OUT}
   ```

   - **IN**: provides inbound filtering.
   - **Out**: provides outbound filtering.

4. Match the service to the message type.

   Example

   ```
   Device(config-mdns-sl-in)# match service-definition-name message-type
   {announcement | query}
   ```

   Here, *service-definition-name* refers to the names of services, such as, airplay, airserver, airtunes, and so on.

   > **Note**
   >
   > To add a service, the service name must be part of the primary service list.
   >
   > If the mDNS service list is set to IN, you get to view this command: **match service-definition-name message-type {announcement | any | query}**.
   >
   > If the mDNS service list is set to OUT, you get to view this command: **match service-definition-name**.

   (OR)

5. Match all services to the message type.

   Example

   ```
   Device(config-mdns-sl-in)# match all message-type {announcement | any | query}
   ```

> **Note**
>
> To add a service, the service name must be part of the primary service list.
>
> If the mDNS service list is set to IN, you get to view this command: **match all message-type {announcement | any | query}**.
>
> If the mDNS service list is set to OUT, you get to view this command: **match all**.

In case of IN or OUT filter, if any of the service contains the same or subset of the message type (query or announcement), the **match all** is not allowed unless the existing services are removed.

6. Display inbound or outbound direction list of the configured service-list to classify matching service-types for service-policy.

   Example

   ```
   Device(config-mdns-sl-in)# show mdns-sd service-list {direction | name }
   ```

   The list can be filtered by name or specific direction.

7. Return to the global configuration mode.

   Example

   ```
   Device(config-mdns-sl-in)# exit
   ```

Create service policy (GUI)

Learn how to create a service policy using the GUI to define input and output service lists for a specified location.

Create a service policy and associate service lists with a specific location through the GUI.

Procedure

1. Choose **Configuration** > **Services** > **mDNS**.
2. In the **Service Policy** section, click **Add**.
3. In the **Quick Setup: Service Policy** page that is displayed, enter a name for the service policy.
4. From the **Service List Input** drop-down list, select one of the types.
5. From the **Service List Output** drop-down list, select one of the types.
6. From the **Location** drop-down list, select the location you want to associate with the service list.
7. Click **Apply to Device**.

Create service policy (CLI)

Configure an mDNS service policy using commands to control mDNS service filtering, define location-based rules, and manage service-list directions.

Create and apply an mDNS service policy for service filtering and control of mDNS learning and responses using commands. mDNS service policy is used for service filtering while learning services or responding to queries.

Procedure

1. Enable the privileged exec mode.

   Example

   ```
   Device> enable
   ```

   Enter your password, if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Enable mDNS service policy.

   Example

   ```
   Device(config)# mdns-sd service-policy service-policy-name
   ```

4. Filter mDNS service types based on location filter.

   Example

   ```
   Device(config-mdns-ser-pol)# location {ap-location | ap-name | location-group
    | lss | regex | site-tag | ssid}
   ```

   Note

   - If a location filter is not applied during the service policy, the system uses the global location filter, which is set to **lss** by default.
   - The location filter specified in the service policy takes precedence, even when a global location filter is configured.
   - In Location Specific Services (LSS) based filtering, the mDNS gateway responds with the service instances learned from the neighboring APs of the querying client AP. The mDNS gateway filters service instances from all other APs.
   - In Site tag based filtering, the mDNS gateway responds with the service instances that belong to the same site-tag as the querying client.
   - The mDNS gateway provides wired services, even when location based filtering is enabled.

5. Configure various service-list names for IN and OUT directions.

Example

```
Device(config-mdns-ser-pol)# service-list service-list-name {IN | OUT}
```

Note

If an administrator creates or uses a custom service policy, the policy must have service-lists for both IN and OUT directions. Without an IN service-list, the mDNS Gateway cannot learn services. Without an OUT service-list, the mDNS Gateway cannot reply to or announce services it has learned.

6. Return to the global configuration mode.

Example

```
Device(config-mdns-ser-pol)# exit
```

Configure a local or native profile for an mDNS policy (CLI)

Configure a native profile on a device to apply an mDNS policy when AAA servers do not supply one using commands.

Ensure mDNS packets are processed based on locally defined profiles when external AAA servers do not provide a policy using commands.

When an administrator configures local authentication and authorization and does not expect to get any mDNS policy from the AAA server, the administrator can configure a local or native profile. This profile selects a mDNS policy based on user, role, or device type. When this local or native profile is mapped to the wireless profile policy, the mDNS service policy is applied to the mDNS packets that are processed on that WLAN.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure the service-template or identity policy.

Example

```
Device(config)# service-template template-name
```

3. Configure the mDNS policy.

Example

```
Device(config-service-template)# mdns-service-policy mdns-policy-name
```

4. Return to the global configuration mode.

Example

```
Device(config-service-template)# exit
```

Configure an mDNS FlexConnect profile (GUI)

Configure an mDNS FlexConnect profile through the GUI to manage multicast DNS services on FlexConnect-enabled devices.

Set up and apply an mDNS FlexConnect profile to define multicast DNS settings for FlexConnect devices using the GUI.

Procedure

1. Choose **Configuration** > **Services** > **mDNS**.
2. In the **mDNS Flex Profile** section, click **Add**.
   The **Add mDNS Flex Profile** window is displayed.
3. In the **Profile Name** field, enter the FlexConnect mDNS profile name.
4. In the **Service Cache Update Timer** field, specify the service cache update time. The default value is 1 minute. The valid range is from 1 to 100 minutes.
5. In the **Statistics Update Timer** field, specify the statistics update timer. The default value is 1 minute. The valid range is from 1 to 100 minutes.
6. In the **VLANs** field, specify the VLAN ID. You can enter multiple VLAN IDs separated by commas, or enter a range of VLAN IDs. Maximum number of VLANs allowed is 16.
7. Click **Apply to Device**.

Configure an mDNS FlexConnect Profile (CLI)

Configure an mDNS FlexConnect profile using commands to manage multicast DNS settings for FlexConnect-enabled devices.

Configure an mDNS FlexConnect profile using commands to manage multicast DNS settings for FlexConnect-enabled devices. This procedure enables customization of service cache timers, statistics timers, and wired VLAN range for optimal mDNS performance.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Enter the mDNS FlexConnect profile mode.

   Example

   ```
   Device(config)# mdns-sd flex-profile mdns-flex-profile-name
   ```

3. Configure the mDNS update service cache timer for the FlexConnect profile.

Example

```
Device(config-mdns-flex-profile)# update-timer service-cache service-cache
timer-value <1-100>
```

The default value is 1 minute. Value range is between 1 minute and 100 minutes.

4. Configure the mDNS update statistics timer for the FlexConnect profile.

Example

```
Device(config-mdns-flex-profile)# update-timer statistics statistics
timer-value <1-100>
```

The default value is 1 minute. The valid range is from 1 to 100 minutes.

5. Configure the mDNS wired VLAN range for the FlexConnect profile between 10 - 20.

Example

```
Device(config-mdns-flex-profile)# wired-vlan-range wired-vlan-range value
```

Apply an mDNS FlexConnect profile to a wireless FlexConnect profile (GUI)

Apply an mDNS FlexConnect profile to a wireless FlexConnect profile using the GUI.

Apply an mDNS FlexConnect profile to a wireless FlexConnect profile using the GUI. This procedure enables mDNS service discovery for wireless devices connected via FlexConnect.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Flex**.
2. Click **Add**.
   The **Add Flex Profile** window is displayed.
3. Under the **General** tab, from the **mDNS Flex Profile** drop-down list, select a FlexConnect profile name from the list.
4. Click **Apply to Device**.

Apply an mDNS FlexConnect profile to a Wireless FlexConnect profile (CLI)

Use this task to apply an mDNS FlexConnect profile to a Wireless FlexConnect profile using commands

Enable mDNS features for all APs in the Wireless FlexConnect profile using commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Enter the wireless FlexConnect profile configuration mode.

Example

```
Device# wireless profile flex wireless-flex-profile-name
```

3. Enable the mDNS features for all the APs in the profile.

Example

```
Device(config-wireless-flex-profile)# mdns-sd mdns-flex-profile
```

Enable the mDNS gateway on the VLAN interface (CLI)

Configure the mDNS service policy for a specific VLAN interface to control mDNS traffic at the VLAN level using commands.

Enable multicast DNS (mDNS) gateway functions on a chosen VLAN interface, allowing granular service policy control for mDNS packets using commands.

This procedure configures the mDNS service policy for a specific VLAN. This allows the administrator to configure different settings to the mDNS packets on per VLAN interface basis and not on per WLAN basis.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure a VLAN ID and enter the interface configuration mode.

Example

```
Device(config)# interface vlan vlan-interface-number
```

3. Configure the IP address for the interface.

Example

```
Device(config-if)# ip address ip-address subnet-mask
```

4. Enable mDNS configuration on a VLAN interface.

Example

```
Device(config-if)# mdns-sd gateway
```

5. Configure the service policy.

   Example

   ```
   Device(config-if-mdns-sd)# service-policy service-policy-name
   ```

   > **Note**
   >
   > If you do not define a specific **service-policy name**, the VLAN uses the **default-mDNS-service-policy** by default.
   >
   > The system automatically creates the **default-mDNS-service-policy**, which uses the **default-mDNS-service-list** configuration to filter mDNS service announcements and queries.

6. Returns to the privileged EXEC mode.

   Example

   ```
   Device(config-if-mdns-sd)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure mDNS AP (CLI)

Configure the mDNS gateway and enable mDNS on an AP using commands.

Enable Multicast DNS (mDNS) services on a specific AP and configure permitted service VLANs using commands.

In most of the deployments, the services may be available in VLANs that the APs can hear in the wired side (allowed in the switchport where the AP is directly connected: its own VLAN, or even more VLANs if switchport is a trunk).

This procedure shows how to configure mDNS AP:

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the mDNS gateway.

   Example

   ```
   Device(config)# mdns-sd gateway
   ```

3. Enable mDNS on the AP, and configure a VLAN for the mDNS AP.

Example

```
Device# ap name ap-name mdns-ap enable vlan vlan-id
```

4. Add a VLAN to the mDNS AP.

   Example

   ```
   Device# ap name ap-name mdns-ap vlan add vlan-id
   ```

   The *vlan-id* ranges from 1 to 4096.

5. Delete a VLAN from the mDNS AP.

   Example

   ```
   Device# ap name ap-name mdns-ap vlan del vlan-id
   ```

6. Disable the mDNS AP.

   Example

   ```
   Device# ap name ap-name mdns-ap disable
   ```

7. Return to the privileged EXEC mode.

   Example

   ```
   Device# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

   Note

   You can configure a maximum of 10 VLANs per AP.

Enable mDNS Gateway on the RLAN interface (CLI)

Configure the mDNS gateway service on a remote LAN interface using commands.

Enable mDNS gateway to allow devices connected to a Remote LAN (RLAN) interface to discover services across VLANs using commands.

By configuring the mDNS gateway mode on the RLAN interface, you can configure the mDNS service policy for a specific RLAN.

Procedure

1. Enter the global configuration mode and configure a remote LAN profile.

Example

```
Device# configure terminal
```

```
Device(config)# ap remote-lan profile-name remote-lan-profile-name rlan-id
```

Example

- *remote-lan-profile*: Remote LAN profile name. The range is from 1 to 32 alphanumeric characters.
- *rlan-id*: Remote LAN identifier. The range is from 1 to 128.

> **Note**
>
> You can create a maximum of 128 RLANs. Also, you cannot use the *rlan-id* of an existing RLAN while creating another RLAN.

2. Enable mDNS configuration on an RLAN interface and restart the RLAN profile.

Example

mdns-sd-interface

```
Device(config-remote-lan)# mdns-sd-interface {gateway | drop}
```

```
Device(config-remote-lan)# no shutdown
```

3. Exit the remote LAN configuration mode and configure the RLAN policy profile and enter the wireless policy configuration mode.

Example

```
Device(config-remote-lan)# exit
```

```
Device(config)# ap remote-lan-policy policy-name profile name
```

4. Enable an mDNS service policy and configure the RLAN for central switching.

Example

```
Device(config-remote-lan-policy)# mdns-sd service-policy service-policy-name
```

```
Device(config-remote-lan-policy)# central switching
```

5. Configure the central DHCP for centrally switched clients and assign the profile policy to a VLAN.

Example

```
Device(config-remote-lan-policy)# central dhcp
```

```
Device(config-remote-lan-policy)# vlan vlan-name
```

6. Restart the RLAN profile and configure a policy tag.

Example

```
Device(config-remote-lan-policy)# no shutdown
```

```
Device(config)# wireless tag policy policy-tag-name
```

7. Map the RLAN policy profile to the RLAN profile.

Example

```
Device(config-policy-tag)# remote-lan remote-lan-profile-name policy
rlan-policy-profile-name port-id port-id
```

- *remote-lan-profile-name* : Name of the RLAN profile.
- *rlan-policy-profile-name*: Name of the policy profile.
- *port-id*: LAN port number on the AP. The range is from 1 to 4.

8. Return to the global configuration mode and configure the AP and enter the AP tag configuration mode.

Example

```
Device(config-policy-tag)# exit
```

```
Device (config)# ap mac-address
```

> **Note**
>
> Use the Ethernet MAC address.

9. Map a policy tag to the AP and return to the privileged EXEC mode.

Example

```
Device (config-ap-tag)# policy-tag policy-tag-name
```

```
Device(config-guest-lan)# end
```

Enable mDNS Gateway on guest LAN interface (CLI)

Enable mDNS gateway on the guest LAN interface.

Enable the mDNS gateway on a guest LAN interface to support discovery and service policies for wired and anchor controllers using commands.

By configuring the mDNS gateway mode on a Guest LAN interface, you can configure the mDNS service policy for a specific Guest LAN interface.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure guest LAN profile with a wired VLAN.

   Example

   ```
   Device(config)# guest-lan profile-name guest_lan_profile_name num wired-vlan
    wired_vlan_num
   ```

   Note

   Configures the wired VLAN only for the Guest Foreign controller.

   - *num*: Guest LAN identifier. The valid range is from 1 to 5.
   - *wired_vlan_num*: Wired VLAN number. The valid range is from 1 to 4094.

3. Configure the guest LAN profile without a VLAN for the Guest Anchor controller.

   Example

   ```
   Device(config)# guest-lan profile-name guest_lan_profile_name
   ```

4. Configure the mDNS gateway for a Guest LAN.

   Example

   ```
   Device(config-guest-lan)# mdns-sd interface {gateway | drop}
   ```

> **Note**
>
> You need to enable mDNS gateway globally for the Guest LAN to work.

5. Return to the privileged EXEC mode.

Example

```
Device(config-guest-lan)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Associate mDNS service policy with wireless profile policy (GUI)

Configure the association between an mDNS service policy and a wireless profile policy using the GUI.

Apply an mDNS service policy to a wireless profile policy so that multicast DNS behavior aligns with desired network access and discovery parameters using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **Policy**.
2. Click the **policy profile** name.
3. In the **Advanced** tab, select the mDNS service policy from the **mDNS Service Policy** drop-down list.
4. Click **Update & Apply to Device**.

Associate mDNS service policy with wireless profile policy (CLI)

Configure the association of an mDNS service policy with a wireless profile policy using commands.

Enable customized mDNS service filtering and announcements within a wireless profile policy using commands.

> **Note**
>
> You must globally configure the mDNS service policy before associating it with the wireless profile policy.

A default mDNS service policy is already attached when the wireless profile policy is created. You can use the following commands to override the default mDNS service policy with any of your service policy:

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configures the wireless profile policy.

   Example

   ```
   Device(config)# wireless profile policy profile-policy
   ```

   Here, *profile-policy* refers to the name of the WLAN policy profile.

3. Associate an mDNS service policy with the wireless profile policy.

   Example

   ```
   Device(config-wireless-policy)# mdns-sd service-policy
   custom-mdns-service-policy
   ```

   The default mDNS service policy name is **default-mdns-service-policy**.

Note

The **default-mdns-profile-policy** uses **default-mdns-service-list** configuration to filter mDNS service announcements and queries.

In a wireless network, the mDNS packets are consumed by the mDNS gateway, so clients or devices are unable to learn about these services. By sharing the service with the device and simplifying configuration for the administrator, a list of standard service types is shared by default on the wireless network. This list of standard service types is referred to as the default service policy, which comprises a set of service types.

The table covers a sample service list in the default service policy.

Table 214: Default Name and mDNS Service Type

Default Name	mDNS Service Type
Apple TV	_airplay._tcp.local _raop._tcp.local
Apple HomeSharing	_home-sharing._tcp.local
Printer-IPPS	_ipps._tcp.local
Apple-airprint	_ipp._tcp.local _universal._sub._ipp._tcp.local
Google-chromecast	_googlecast._tcp.local _googlerpc._tcp.local _googlezone._tcp.local
Apple-remote-login	_sftp-ssh._tcp.local _ssh._tcp.local
Apple-screen-share	_rfb._tcp.local
Google-expeditions	_googexpeditions._tcp.local
Multifunction-printer	_fax-ipp._tcp.local _ipp._tcp.local _scanner._tcp.local
Apple-windows-fileshare	_smb._tcp.local

Note

- Location would be disabled on mDNS default service policy.
- You cannot change the contents of the mDNS default service policy. However, you can create separate mDNS service policies and associate them under the wireless policy profile.

4. Return to the global configuration mode.

Example

```
Device(config-wireless-policy)# exit
```

Enable or disable mDNS Gateway for WLAN (GUI)

Learn how to enable or disable the mDNS Gateway for a WLAN using the GUI.

Enable or disable the mDNS Gateway feature for a specific WLAN to optimize device discovery and network performance using the GUI.

Procedure

1. Choose **Configuration** > **Tags & Profiles** > **WLANs**.
2. Click on the WLAN.
3. In the **Advanced** tab, select the mode in **mDNS Mode** drop-down list.
4. Click **Update & Apply to Device**.

Enable or disable mDNS Gateway for WLAN (CLI)

Use this task to configure the mDNS Gateway for a specified WLAN using commands.

Control mDNS Gateway functionality for a wireless LAN using commands.

> **Note**
>
> Bridging is the default behaviour. This means that the mDNS packets are always bridged.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Specify the WLAN name and ID.

Example

```
Device(config)# wlan profile-name wlan-id ssid-name
```

- *profile-name* is the WLAN name, and it can contain up to 32 alphanumeric characters.
- *wlan-id* is the wireless LAN identifier. The valid range is from 1 to 4096.

- *ssid-name* is the SSID which can contain 32 alphanumeric characters.

> **Note**
>
> Global configuration must be in place for mDNS Gateway to work.

3. Enable or disable mDNS Gateway and bridge functions on WLAN.

   Example

   ```
   Device(config-wlan)# mdns-sd interface {gateway | drop}
   ```

4. Return to the global configuration mode.

   Example

   ```
   Device(config-wlan)# exit
   ```

5. Verify the status of mDNS on WLAN.

   Example

   ```
   Device# show wlan name wlan-name show wlan all
   ```

6. Verify the service policy configured in WLAN.

   Example

   ```
   Device# show wireless profile policy
   ```

---

mDNS Gateway with guest anchor support and mDNS bridging

This topic explains how mDNS gateway functions in guest anchor deployments, depending on whether the gateway is enabled on Anchor, Foreign, or both controllers.

- When mDNS Gateway is enabled on both the Anchor and Foreign controllers, the mDNS gateway supports guest anchor deployments. In this mode, clients on a guest LAN or WLAN with guest anchor enabled receive responses with any services or cache from the export Foreign controller. All advertisements received on the guest LAN or WLAN by the export Foreign are learned on the export Foreign itself. All queries received on the guest LAN or WLAN are answered by the export Foreign itself.
- When mDNS Gateway is enabled on the Anchor and disabled on the Foreign controller (Bridging Mode), the mDNS gateway supports guest anchor deployments. In this scenario, clients on the guest LAN or WLAN with guest anchor enabled receive any services or cache from the export Anchor, even if the clients are connected to the Foreign. All advertisements received on the guest LAN or WLAN by the export Foreign are forwarded to the Anchor, and the cache is stored on the Anchor. All queries received on the guest LAN or WLAN are answered by the export Anchor itself.

> **Note**
>
> Comment by CISCO\ashijadh: based on PRRQ comments
>
> - You must configure the guest LAN to a wireless profile policy that is configured with the required mDNS service policy.
> - To configure a non-guest LAN mDNS gateway, see the mDNS Gateway chapter.

Configure mDNS Gateway on guest anchor (CLI)

Configure the mDNS Gateway on a guest anchor using commands.

Configures the mDNS Gateway functionality on a guest anchor device using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the guest LAN profile with a wired VLAN.

   Example

   ```
   Device(config)# guest-lan profile-name guest-lan-profile-name guest-lan-id
   ```

3. Enable mDNS Gateway on the guest LAN.

   Example

   ```
   Device(config-guest-lan)# mdns-sd gateway
   ```

Configure mDNS Gateway on guest foreign (Guest LAN) (CLI)

Configure the mDNS Gateway on a Guest Foreign controller for Guest LAN wired VLANs using commands.

Enable mDNS Gateway functionality for users and devices connected to the Guest LAN via a wired VLAN using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure guest LAN profile with a wired VLAN.

   Example

   ```
   Device(config)# guest-lan profile-name guest-lan-profile-name guest-lan-id
   wired-vlan vlan-id
   ```

   > **Note**
   >
   > Configure the wired VLAN only for the Guest Foreign controller.

3. Enable mDNS Gateway on the guest LAN.

   Example

   ```
   Device(config-guest-lan)# mdns-sd gateway
   ```

4. Return to the global configuration mode.

   Example

   ```
   Device(config-wireless-policy)# exit
   ```

Configure mDNS Gateway on guest anchor (CLI)

Configure an mDNS Gateway for a guest WLAN anchored on the controller using commands.

Enable mDNS Gateway functionality on a guest WLAN profile anchored to a VLAN, allowing multicast DNS service discovery for guest users using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the guest WLAN profile with a wired VLAN.

   Example

   ```
   Device(config)# guest-wlan profile-name guest-lan-profile-name guest-wlan-id
   ```

3. Enable mDNS Gateway on the guest WLAN.

Example

```
Device(config-guest-wlan)# mdns-sd gateway
```

Configure mDNS Gateway on guest foreign (Guest WLAN) (CLI)

Configure the mDNS Gateway on a guest foreign controller for a guest WLAN that includes a wired VLAN.

Enable service discovery for devices on a guest wireless LAN (WLAN) and ensure multicast DNS (mDNS) works for clients connected to a guest wired VLAN on the foreign controller using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure guest WLAN profile with a wired VLAN.

   Example

   ```
   Device(config)# guest-wlan profile-name guest-lan-profile-name guest-wlan-id
    wired-vlan vlan-id
   ```

   Note

   Configure the wired VLAN only for the Guest Foreign controller.

3. Enable mDNS Gateway on the guest WLAN.

   Example

   ```
   Device(config-guest-wlan)# mdns-sd gateway
   ```

4. Return to the global configuration mode.

   Example

   ```
   Device(config-wireless-policy)# exit
   ```

Verify mDNS Gateway configurations

This section lists the key commands and their outputs for verifying mDNS Gateway configurations, cache details, service statistics, and AP associations in your network.

To verify the mDNS summary, use this command:

```
Device# show mdns-sd summary

mDNS Gateway: Enabled
Active Query: Enabled
  Periodicity (in minutes): 30
Transport Type: IPv4
```

To verify the mDNS cache, use this command:

```
Device# show mdns-sd cache

------------------------------------------------------------ PTR Records
--------------------------------------
RECORD-NAME                    TTL      WLAN   CLIENT-MAC       RR-RECORD-DATA

_____________________________________________________________________________
_airplay._tcp.local             4500     30     07c5.a4f2.dc01
CUST1._airplay._tcp.local
_ipp._tcp.local                 4500     30     04c5.a4f2.dc01
CUST3._ipp._tcp.local2
_ipp._tcp.local                 4500     15     04c5.a4f2.dc01
CUST3._ipp._tcp.local4
_ipp._tcp.local                 4500     10     04c5.a4f2.dc01
CUST3._ipp._tcp.local6
_veer_custom._tcp.local         4500     10     05c5.a4f2.dc01
CUST2._veer_custom._tcp.local8
```

To verify the mDNS cache from wired service provider, use this command:

```
Device# show mdns-sd cache wired

------------------------------------------------------------ PTR Records
--------------------------------------
RECORD-NAME                                TTL      VLAN      CLIENT-MAC
  RR-RECORD-DATA
_____________________________________________________________________________
_airplay._tcp.local                        4500     16        0866.98ec.97af
  wiredapple._airplay._tcp.local
_raop._tcp.local                           4500     16        0866.98ec.97af
  086698EC97AF@wiredapple._raop._tcp.local
------------------------------------------------------------ SRV Records
--------------------------------------
RECORD-NAME                                TTL      VLAN      CLIENT-MAC
  RR-RECORD-DATA
_____________________________________________________________________________
wiredapple._airplay._tcp.local             4500     16        0866.98ec.97af
  0 0 7000 wiredapple.local
086698EC97AF@wiredapple._raop._tcp.local   4500     16        0866.98ec.97af
  0 0 7000 wiredapple.local
------------------------------------------------------------ A/AAAA Records
--------------------------------------
RECORD-NAME                                TTL      VLAN      CLIENT-MAC
  RR-RECORD-DATA
_____________________________________________________________________________
wiredapple.local                           4500     16        0866.98ec.97af
  2001:8:16:16:e5:c446:3218:7437
------------------------------------------------------------ TXT Records
--------------------------------------
RECORD-NAME                                TTL      VLAN      CLIENT-MAC
  RR-RECORD-DATA
```

```
_____________________________________________________________________________________________
wiredapple._airplay._tcp.local              4500     16        0866.98ec.97af
  [343]'acl=0''deviceid=08:66:98:EC:97:AF''features=
086698EC97AF@wiredapple._raop._tcp.local    4500     16        0866.98ec.97af
  [193]'cn=0,1,2,3''da=true''et=0,3,5''ft=0x5A7FFFF7
```

To verify the mdns-sd type PTR, use this command:

```
Device# show mdns-sd cache type {PTR | SRV | A-AAA | TXT}

RECORD-NAME                                    TTL      WLAN      CLIENT-MAC
      RR-Record-Data
_____________________________________________________________________________________________
_custom1._tcp.local                           4500     2         c869.cda8.77d6
   service_t1._custom1._tcp.local
_custom1._tcp.local                           4500     2         c869.cda8.77d6
   vk11._custom1._tcp.local
_ipp._tcp.local                               4500     2         c869.cda8.77d6
   service-4._ipp._tcp.local
```

To verify the mdns-sd cache for a client MAC, use this command:

```
Device# show mdns-sd cache {ap-mac <ap-mac> | client-mac <client-mac>


                    | glan-id <glan-id> | mdns-ap <mac-address> | rlan-id
<rlan-id>

                | wlan-id <wlan-id> | wired}
RECORD-NAME                                    TTL      WLAN      CLIENT-MAC
      RR-Record-Data
_____________________________________________________________________________________________
_custom1._tcp.local                           4500     2         c869.cda8.77d6
   service_t1._custom1._tcp.local
_custom1._tcp.local                           4500     2         c869.cda8.77d6
   vk11._custom1._tcp.local
_ipp._tcp.local                               4500     2         c869.cda8.77d6
   service-4._ipp._tcp.local
------------------------------------------------------------ SRV Records
------------------------------------------------------------
RECORD-NAME                                    TTL      WLAN      CLIENT-MAC
      RR-Record-Data
_____________________________________________________________________________________________
service-4._ipp._tcp.local                     4500     2         c869.cda8.77d6
   0 0 1212 mDNS-Client1s-275.local
vk11._custom1._tcp.local                      4500     2         c869.cda8.77d6
   0 0 987 mDNS-Client1s-275.local
service_t1._custom1._tcp.local                4500     2         c869.cda8.77d6
   0 0 197 mDNS-Client1s-275.local
------------------------------------------------------------ A/AAAA Records
------------------------------------------------------------
RECORD-NAME                                    TTL      WLAN      CLIENT-MAC
      RR-Record-Data
_____________________________________________________________________________________________
mDNS-Client1s-275.local                       4500     2         c869.cda8.77d6
   120.1.1.33
------------------------------------------------------------ TXT Records
------------------------------------------------------------
RECORD-NAME                                    TTL      WLAN      CLIENT-MAC
      RR-Record-Data
_____________________________________________________________________________________________
```

```
service-4._ipp._tcp.local                    4500     2         c869.cda8.77d6
   'CLient1'
vk11._custom1._tcp.local                     4500     2         c869.cda8.77d6
   'txtvers=11'
service_t1._custom1._tcp.local               4500     2         c869.cda8.77d6
   'txtvers=12'
```

To verify the mdns-sd cache with respect to the RLAN ID, use this command:

```
Device# show mdns-sd cache rlan-id 1 detail

Name: _printer._tcp.local
  Type: PTR
  TTL: 4500
  RLAN: 1
  RLAN Name: rlan_test_1
  VLAN: 141
  Client MAC: 000e.c688.3942
  AP Ethernet MAC: 0042.5ab6.0ef0
  Remaining-Time: 4485
  Site-Tag: default-site-tag
  mDNS Service Policy: mdnsTV6
  Overriding mDNS Service Policy: NO
  UPN-Status: Disabled
  Rdata: printer._printer._tcp.local
Name: lab-47-187.local
  Type: A/AAAA
  TTL: 4500
  RLAN: 1
  RLAN Name: rlan_test_1
  VLAN: 141
  Client MAC: 000e.c688.3942
  AP Ethernet MAC: 0042.5ab6.0ef0
  Remaining-Time: 4485
  Site-Tag: default-site-tag
  mDNS Service Policy: mdnsTV6
  Overriding mDNS Service Policy: NO
  UPN-Status: Disabled
  Rdata: 10.15.141.124
```

To verify the mdns-sd cache with respect to mDNS-AP, use this command:

```
Device# show mdns-sd cache mdns-ap 706b.b97d.b060 detail

Name: _printer._tcp.local
  Type: PTR
  TTL: 4500
  VLAN: 145
  Client MAC: 0050.b626.5bfa
  mDNS AP Radio MAC: 706b.b97d.b060
  mDNS AP Ethernet MAC: 706b.b97c.5208
  Remaining-Time: 4480
  mDNS Service Policy: mdnsTV
  Rdata: printer._printer._tcp.local
Name: Client-46-153.local
  Type: A/AAAA
  TTL: 4500
  VLAN: 145
  Client MAC: 0050.b626.5bfa
  mDNS AP Radio MAC: 706b.b97d.b060
  mDNS AP Ethernet MAC: 706b.b97c.5208
```

```
  Remaining-Time: 4480
  mDNS Service Policy: mdnsTV
  Rdata: 10.15.145.103
```

To verify the mdns-sd cache in detail, use this command:

```
Device# show mdns-sd cache detail

Name: _custom1._tcp.local
  Type: PTR
  TTL: 4500
  WLAN: 2
  WLAN Name: mdns120
  VLAN: 120
  Client MAC: c869.cda8.77d6
  AP Ethernet MAC: 7069.5ab8.33d0
  Expiry-Time: 09/09/18 21:50:47
  Site-Tag: default-site-tag
  Rdata: service_t1._custom1._tcp.local
```

To verify the mdns-sd cache statistics, use this command:

```
Device# show mdns-sd cache statistics

mDNS Cache Stats
Total number of Services: 4191
```

To verify the mdns-sd statistics, use this command:

```
Device# show mdns-sd statistics

-------------------------------------------------------
Consolidated mDNS Packet Statistics
-------------------------------------------------------
mDNS stats last reset time: 03/11/19 04:17:35
mDNS packets sent: 61045
  IPv4 sent: 30790
    IPv4 advertisements sent: 234
    IPv4 queries sent: 30556
  IPv6 sent: 30255
    IPv6 advertisements sent: 17
    IPv6 queries sent: 30238
  Multicast sent: 57558
    IPv4 sent: 28938
    IPv6 sent: 28620
mDNS packets received: 72796
  advertisements received: 13604
  queries received: 59192
  IPv4 received: 40600
    IPv4 advertisements received: 6542
    IPv4 queries received: 34058
  IPv6 received: 32196
    IPv6 advertisements received: 7062
    IPv6 queries received: 25134
mDNS packets dropped: 87
-------------------------------------------------------
Wired mDNS Packet Statistics
-------------------------------------------------------
mDNS stats last reset time: 03/11/19 04:17:35
```

```
mDNS packets sent: 61033
  IPv4 sent: 30778
    IPv4 advertisements sent: 222
    IPv4 queries sent: 30556
  IPv6 sent: 30255
    IPv6 advertisements sent: 17
    IPv6 queries sent: 30238
  Multicast sent: 57558
    IPv4 sent: 28938
    IPv6 sent: 28620
mDNS packets received: 52623
  advertisements received: 1247
  queries received: 51376
  IPv4 received: 32276
    IPv4 advertisements received: 727
    IPv4 queries received: 31549
  IPv6 received: 20347
    IPv6 advertisements received: 520
    IPv6 queries received: 19827
mDNS packets dropped: 63
-------------------------------------------------------
mDNS Packet Statistics, for WLAN: 2
-------------------------------------------------------
mDNS stats last reset time: 03/11/19 04:17:35
mDNS packets sent: 12
  IPv4 sent: 12
    IPv4 advertisements sent: 12
    IPv4 queries sent: 0
  IPv6 sent: 0
    IPv6 advertisements sent: 0
    IPv6 queries sent: 0
  Multicast sent: 0
    IPv4 sent: 0
    IPv6 sent: 0
mDNS packets received: 20173
  advertisements received: 12357
  queries received: 7816
  IPv4 received: 8324
    IPv4 advertisements received: 5815
    IPv4 queries received: 2509
  IPv6 received: 11849
    IPv6 advertisements received: 6542
    IPv6 queries received: 5307
mDNS packets dropped: 24
```

To verify the default service list details, use this command:

```
Device# show mdns-sd default-service-list

-----------------------------------------
mDNS Default Service List
-----------------------------------------
Service Definition: apple-tv
Service Names: _airplay._tcp.local
_raop._tcp.local
Service Definition: homesharing
Service Names: _home-sharing._tcp.local
Service Definition: printer-ipps
Service Names: _ipps._tcp.local
Service Definition: apple-airprint
Service Names: _ipp._tcp.local
_universal._sub._ipp._tcp.local
```

```
Service Definition: google-chromecast
Service Names: _googlecast._tcp.local
_googlerpc._tcp.local
_googlezone._tcp.local
Service Definition: apple-remote-login
Service Names: _sftp-ssh._tcp.local
_ssh._tcp.local
Service Definition: apple-screen-share
Service Names: _rfb._tcp.local
Service Definition: google-expeditions
Service Names: _googexpeditions._tcp.local
Service Definition: multifunction-printer
Service Names: _fax-ipp._tcp.local
_ipp._tcp.local
_scanner._tcp.local
Service Definition: apple-windows-fileshare
Service Names: _smb._tcp.local
```

To verify the primary service list details, use this command:

```
Device# show mdns-sd master-service-list

-----------------------------------------
        mDNS Master Service List
-----------------------------------------
Service Definition: fax
Service Names: _fax-ipp._tcp.local
Service Definition: roku
Service Names: _rsp._tcp.local
Service Definition: airplay
Service Names: _airplay._tcp.local
Service Definition: scanner
Service Names: _scanner._tcp.local
Service Definition: spotify
Service Names: _spotify-connect._tcp.local
Service Definition: airtunes
Service Names: _raop._tcp.local
Service Definition: airserver
Service Names: _airplay._tcp.local
               _airserver._tcp.local
.
.
.
Service Definition: itune-wireless-devicesharing2
Service Names: _apple-mobdev2._tcp.local
```

To verify the mdns-sd service statistics on the controller, use this command:

```
Device# show mdns-sd service statistics

Service Name                                                    Service Count

-------------------------------------------------------------------------------
_atc._tcp.local                                                   137

_hap._tcp.local                                                   149

_ipp._tcp.local                                                   149

_rfb._tcp.local                                                   141
```

```
_smb._tcp.local                                                    133

_ssh._tcp.local                                                    142

_daap._tcp.local                                                   149

_dpap._tcp.local                                                   149

_eppc._tcp.local                                                   138

_adisk._tcp.local                                                  149
```

To verify the mDNS-AP configured on the controller and VLAN(s) associated with it, use this command:

```
Device# show mdns-sd ap

Number of mDNS APs.................................. 1
AP Name   Ethernet MAC   Number of Vlans   Vlanidentifiers
-----------------------------------------------------------------------------------------------
AP3600-1   7069.5ab8.33d0          1      300
```

Further Debug

To debug mDNS further, use this procedure:

1. Run this command at the controller:

```
set platform software trace wncd <0-7> chassis active R0 mdns debug
```

2. Reproduce the issue.
3. Run this command to gather the traces enabled:

```
show wireless loadbalance ap affinity wncd 0
AP MAC   Discovery Timestamp  Join Timestamp               Tag
Vlanidentifiers
-----------------------------------------------------------------------------------------
0cd0.f894.0600      06/30/21 12:39:48    06/30/21 12:40:021 default-site-tag
    300
```

Location based service filtering

Explains how location-based service filters control access to services based on a device's physical or logical location within a network.

Prerequisite for location-based service filtering

This topic outlines the prerequisites you must meet before configuring location-based service filtering.

- You need to create the Service Definition and Service Policy.
- For more information, see Creating Custom Service Definition section and Creating Service Policy section.

Configure mDNS location-based filtering using SSID (CLI)

Configure a service policy to enable mDNS location-based filtering using the SSID as the location criteria.

Limit mDNS service announcements to only those learned on the configured SSID, enhancing security and service relevance using commands.

When a service policy is configured with the SSID as the location name, the response to the query will be the services that were learnt on that SSID.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the service policy.

   Example

   ```
   Device(config)# mdns-sd service-policy service-policy-name
   ```

3. Configure location-based filtering using SSID.

   Example

   ```
   Device(config-mdns-ser-pol)# location ssid
   ```

4. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-mdns-ser-pol)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure mDNS location-based filtering using AP name (CLI)

Configure an mDNS service policy to provide location-based filtering using the AP name as the location reference.

Limit mDNS query responses to only services discovered on a specified AP, improving service localization and security using commands.

When a service policy is configured with the AP name as the location, the response to the query will be the services that were learnt on that AP.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the service policy.

   Example

   ```
   Device(config)# mdns-sd service-policy service-policy-name
   ```

3. Configure location-based filtering using an AP name.

   Example

   ```
   Device(config-mdns-ser-pol)# location ap-name
   ```

4. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-mdns-ser-pol)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure mDNS location-based filtering using AP location (CLI)

Configure Multicast DNS (mDNS) location-based filtering using the AP location in the CLI.

Configure mDNS location-based filtering using the AP location and these commands.

When a service policy is configured with location as the AP-location, the response to the query will be the services that were learnt on all the APs using the same AP "location" name (not to be confused with "site-tag").

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the service policy.

   Example

   ```
   Device(config)# mdns-sd service-policy service-policy-name
   ```

3. Configure location-based filtering using the AP location.

   Example

   ```
   Device(config-mdns-ser-pol)# location ap-location
   ```

4. Return to the privileged EXEC mode.

Example

```
Device(config-mdns-ser-pol)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure mDNS location-based filtering using regular expression (CLI)

Learn how to configure mDNS location-based filtering using regular expressions in CLI to match services based on AP name or AP location.

Enable filtering of mDNS services using location-based regular expressions for targeted service policy assignment using commands.

- If a service policy uses a location specified as a regular expression matching the AP name, the query returns services learned on all APs whose names match.
- If a service policy uses a location specified as a regular expression matching the AP location, the query returns services learned on all APs whose locations match.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure the service policy.

   Example

   ```
   Device(config)# mdns-sd service-policy service-policy-name
   ```

3. Configure location-based filtering using regular expression.

   Example

   ```
   Device(config-mdns-ser-pol)# location regex {ap-location regular-expression
   | ap-name regular-expression}
   ```

4. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-mdns-ser-pol)# end
   ```

> **Note**
>
> To filter the services for which AP names have the specific keyword such as *AP-2FLR-SJC-123*, you can use the regex AP name as *AP-2FLR-* to match the services that are learnt from the set of APs.

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure mDNS location based filtering using location group

-

mDNS location-Based filtering using location group (Microlocation)

Information about mDNS location-based filtering using location group (microlocation).

mDNS location-based filtering using location groups is a method that

- manages and restricts mDNS service advertisements and discovery, based on the deployment area of wireless APs, and
- a location group, also known as a microlocation or mDNS AP Group, is formed by assigning APs to groups, either by AP name or by their configured physical location.

To create an mDNS AP location group, complete these steps

1. Define multiple rules with priority in the wireless rule-based mDNS application.

> **Note**
>
> Assign AP microlocation grouping by AP name or AP location.

2. Match the highest priority rule-based on the configured regular expression with AP name and AP location-based grouping.
3. Map an AP to a location group (mDNS Group ID).

> **Note**
>
> When you delete or modify a rule, the corresponding APs are revalidated using the **capwap restart** command to apply the updated configuration.

In the context of Apple Bonjour, microlocation refers to a subset of a wireless location. This subset is also known as an mDNS AP group or location group.

AP Microlocation Support Based on AP Location

Starting from Cisco IOS-XE 17.3, configure AP location using the **ap name** *name* **location** *location* command.

Starting from Cisco IOS-XE 17.9, leverage AP location to group APs within a location and form a location group.

By default, AP microlocation based on either AP name or AP location is disabled.

Feature history for mDNS location-based filtering using location group (microlocation)

This table provides release and related information for the feature explained in this module.

This feature is also available in all the releases subsequent to the one in which they are introduced in, unless noted otherwise.

Table 215: Feature history for mDNS location-based filtering using location group (microlocation)

Release	Feature	Feature Information
Cisco IOS XE Cupertino 17.9.1	mDNS location-based filtering using location group (microlocation)	The controller is enhanced to support microlocation from wireless clients tagged with the location group (mDNS Group ID) tag. From Cisco IOS-XE 17.3 onwards, location grouping is done based on AP names. From Cisco IOS-XE 17.9 onwards, location grouping is extended to AP location.

Use cases for mDNS location-based filtering using location group (microlocation)

This topic lists common scenarios where mDNS location-based filtering using location groups (microlocation) enhances service access and containment across different environments.

- Restricts services across departments to ensure only authorized users have visibility.
- Shares files across buildings or sites without exposing resources network-wide.
- Provides service visibility to teachers or students, doctors or patients, employees or specific groups within contained environments, avoiding the need for IT intervention to change Layer 2 or Layer 3 networks.

Prerequisites for mDNS location-based filtering using location group (microlocation)

This topic explains the essential requirements and system defaults governing mDNS location-based filtering using location groups (microlocation).

- You must configure the mDNS rule.
- By default, the system uses AP name-based microlocation grouping.

Enable location group (CLI)

This topic describes how to enable a location group on a device by using command-line interface commands.

To enable a location group, follow these steps to configure the required mDNS service policy and apply location-based filtering using commands.

Procedure

1. Enter the global configuration mode.

Example

```
Device# configure terminal
```

2. Configure mDNS service policy.

   Example

   ```
   Device(config)# mdns-sd service-policy service-policy-name
   ```

3. Configure service lists for IN and OUT directions.

   Example

   ```
   Device(config-mdns-ser-pol)# service-list service-list-name {in | out}
   ```

4. Configure location-based filtering using location group.

   Example

   ```
   Device(config-mdns-ser-pol)# location location-group
   ```

5. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-mdns-ser-pol)# end
   ```

Add APs to a location group (CLI)

Configure APs to join a location group for mDNS applications using commands.

Group APs based on location name or AP name for mDNS rule-based applications using commands.

Procedure

1. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

2. Configure wireless rule-based MDNS application.

   Example

   ```
   Device(config)# wireless rule application mdns
   ```

3. Configure rule priority.

Example

```
Device(config-app-rule)# rule-priority rule_priority rule-name rule_name
```

Here,

- *rule_priority*: The valid range is from 0 to 4096.

  Note

  0 is the lower priority number and 4096 is the higher priority number.

- *rule_name*: The rule name can be between 1 to 32 characters.

Note

When you configure the rule priority, you will see the prompt: `Changing regex string or other rule configuration may cause associated APs to rejoin.`

When you see this prompt, enter **Y** to continue with the configuration.

4. Configure rule-based on AP name or AP location to match the regular expression.

   Example

   ```
   Device(config-rule-params)# regex regular_expression_string
   ```

5. Group APs based on the filter string.

   Example

   ```
   Device(config-rule-params)# action-type grouping
   ```

6. Configure the mDNS location group identifier.

   Example

   ```
   Device(config-rule-action-mdns)# group-id location_group_identifier
   ```

   The valid range for *location_group_identifier* is 1 to 4096.

7. Configure AP location-based grouping.

   Example

   ```
   Device(config-rule-action-mdns)# group-method ap location
   ```

> **Note**
>
> If you treat the **group-method** as **ap location** , the regular expression captures the *AP_LOC_NAME*. By default, the **group-method** uses *AP_NAME*.

8. Configure mDNS location group name.

   Example

```
Device(config-rule-action-mdns)# group-name location_group_name
```

9. Return to the privileged EXEC mode.

   Example

```
Device(config-rule-action-mdns)# end
```

Verify AP location

This topic provides commands to verify AP mDNS group associations, rule names, group IDs, and methods.

To verify the mDNS location Group ID associated with an AP, use this command:

```
Device# show ap config general | sec MDNS | AP Name
Cisco AP Name   : AP2800

MDNS Group Id                                   : 101
MDNS Rule Name                                  : R101
MDNS Group Method                               : AP Location
```

To verify all the APs associated with the configured mDNS rule name, use this command:

```
Device# show wireless associated-ap mdns-rule-name R1
AP MAC              AP Name
-----------------------------------------------------------------------
0cd0.f894.a840      AP0CD0.F894.083C
4001.7a03.8560      APA023.9F66.4F96
--------------------
```

To verify all the APs associated with the configured mDNS location group ID, use this command:

```
Device# show wireless associated-ap mdns-group-id 1
AP MAC              AP Name
-----------------------------------------------------------------------
0cd0.f894.a840      AP0CD0.F894.083C
4001.7a03.8560      APA023.9F66.4F96
--------------------
```

To verify the mDNS group method detail for each AP, use this command:

```
Device# show ap config general | inc MDNS|AP Name|Location
Cisco AP Name   : AP-1
MDNS Group Id                                   : 100
MDNS Rule Name                                  : R100

MDNS Group Method                               : AP Location
```

To verify the mDNS group method detail for each rule, use this command:

```
                Device# show wireless rule application mdns
Rule Name          : R100
Rule Priority      : 100
Regular Expression : AP0
Action Type        : MDNS Grouping
  MDNS Group ID    : 100
MDNS Group Name  : G100

                MDNS Group Method: AP Location
```

Nearest mDNS based wired service filtering

Explains nearest mDNS-based wired service filters, what category they belong to, and their distinguishing attributes for controlling multicast DNS service advertisements on local area networks.

Nearest mDNS-based wired service filtering

Information about nearest mDNS-based wired service filtering.

A nearest mDNS-based wired service filter is a service discovery mechanism that

- classifies wired services based on proximity to the wireless clients
- advertises only those wired services available from nearest mDNS-enabled APs, and
- supports cache population and filtering on the controller for relevant service visibility.

Prior to the Cisco IOS XE 17.8.1 release, wireless clients discovered:

- All wired services from mDNS-AP.
- Service providers on VLANs visible to the controller.

Note

The current filtering is supported only for wireless services.

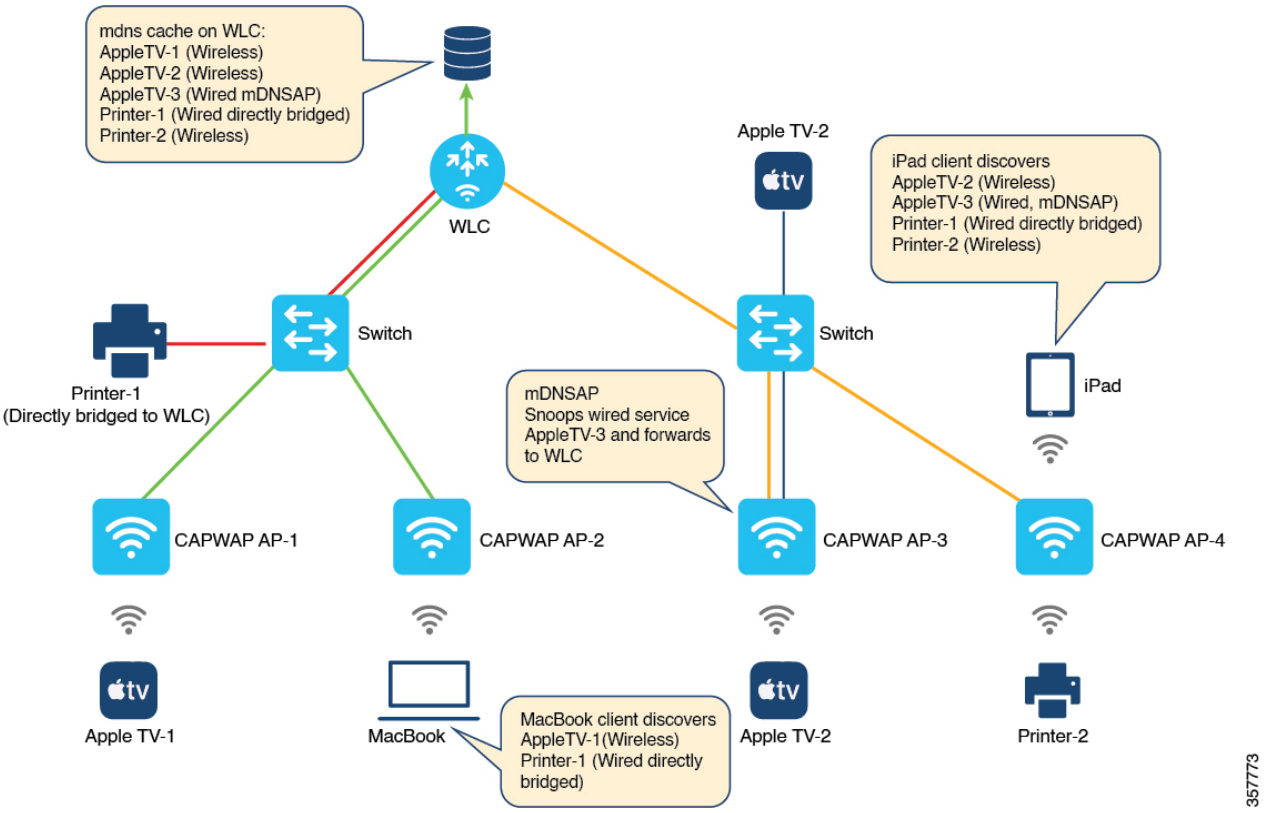
Note

The controller classifies wired services as the nearest wired services once LSS is enabled. The mDNS-AP then forwards or advertises the nearest wired services.

Starting from Cisco IOS XE 17.8.1, wireless clients are enhanced to support filtering based on the nearest wired service provider.

This figure illustrates the nearest wired service provider and discovery:

Figure 69: Nearest Wired Service Provider and Discovery



As per the figure, the controller is associated with these four APs:

- CAPWAP AP-1
- CAPWAP AP-2
- CAPWAP AP-3
- CAPWAP AP-4

The client connected to CAPWAP AP-1 is wireless and advertises the service Apple TV-1.

Similarly, the client connected to CAPWAP AP-2 is wireless and advertises the service MacBook query client.

The CAPWAP AP-3 is enabled as an mDNS-AP. This AP discovers wired services on VLANs and forwards them to the controller. In this case, the client advertising the service AppleTV-3 is a wired service, which is discovered by CAPWAP AP-3 and forwarded to the controller. You may also see another client connected to CAPWAP AP-3 that is wireless and advertises the service AppleTV-2.

The client connected to CAPWAP AP-4 is wireless and advertises the service Printer-2 and iPad query client.

Also, a client is connected directly to the controller, which advertises the Printer-1.

The controller covers cache populated from both wireless and wired service providers.

The controller creates a cache with these entries:

- AppleTV-1 (Wireless service from CAPWAP AP-1)
- AppleTV-2 (Wireless service from CAPWAP AP-3)
- AppleTV-3 (Wired service from mDNS-AP enabled AP-3)
- Printer-1 (Wired service from directly bridged service provider)
- Printer-2 (Wireless service from AP-4)

When LSS is enabled, AP-1 and AP-2 discover each other as LSS neighbors. Similarly, AP-3 and AP-4 discover each other as LSS neighbors.

MacBook discovers these services:

- AppleTV-1 (wireless service from AP-1)
- Printer-1 (wired service from the directly bridged service provider)

> **Note**
>
> MacBook does not discover the wired service AppleTV-3 (forwarded by mDNS-AP AP-3). The AP-2 does not see AP-3 as the LSS neighbor. Thus, the controller does not classify the wired service AppleTV-3 as nearby.

iPad discovers these services:

- AppleTV-2 (wireless service from AP-3)
- AppleTV-3 (wired service from mDNS-AP enabled AP-3)
- Printer-1 (wired service from directly bridged service provider)
- Printer-2 (wireless service from AP-4)

> **Note**
>
> iPad discovers the wired service AppleTV-3 (forwarded by mDNS-AP AP-3). The AP-4 sees AP-3 as the LSS neighbor. Thus, the controller classifies the wired service AppleTV-3 as nearby.

> **Note**
>
> This feature supports only the wired services advertised by mDNS-AP in centrally switched local mode.

Feature history for nearest mDNS-based wired service filtering

This table provides release and related information for features explained in this module.

These features are available on all releases subsequent to the one they were introduced in, unless noted otherwise.

Table 216: Feature history for nearest mDNS-based wired service filtering

Release	Feature	Feature Information
Cisco IOS XE Cupertino 17.8.1	Nearest mDNS-based wired service filtering	This feature supports these functionalities: • Nearest mDNS based wired service filtering. (Supported in Central switched Local mode.) • Custom wired service policy support for FlexConnect mode. • VLAN and MAC based wired service filtering. (Supported in Central switched Local mode.)

Custom wired service policy support for FlexConnect mode

Information about custom wired service policy support for FlexConnect mode.

A custom wired service policy is a configuration feature that

- allows users to apply mDNS service policies to wired networks in FlexConnect profiles, and
- is available starting from Cisco IOS XE 17.8.1, providing extended policy support.

VLAN and MAC based wired service filtering

Information about VLAN and MAC based wired service filtering.

VLAN and MAC based wired service filterings are network control mechanisms that

- allow filtering of wired services by VLAN and MAC address

- enable OUT direction filter application for mDNS-AP and bridged wired services, and
- support centrally switched local mode from Cisco IOS XE 17.8.1 onward.

Before Cisco IOS XE 17.8.1, service filtering was based on service type, location type, and location filter. These filters apply to wireless services but are not supported for wired services.

Prerequisite for nearest mDNS-based wired service filtering

- Enable the mDNS gateway on the controller.

Use cases

This topic ists use cases and limitations of the nearest mDNS-based wired service filtering for Cisco wireless controllers.

These are the use cases:

- Nearest mDNS-based wired service filtering.
- Custom wired service policy support for FlexConnect mode.
- VLAN and MAC based wired service filtering.

While migrating from AireOS wireless controllers to the Cisco Catalyst 9800 Series wireless controllers, these limitations occur:

- Wireless clients discover all wired services, not just the nearby service from the wired service provider, when centrally switched local mode and LSS are enabled. The wired services belong to both the forwarded mDNS-AP and the directly bridged services.
- TYou cannot apply the custom service policy for wired services when locally switched FlexConnect mode is enabled. The mDNS FlexConnect profile must include the custom wired service policy.
- You cannot filter wired services by VLAN and MAC address in centrally switched local mode.

Configure wired service policy support in FlexConnect profile

Create service list (CLI)

Configure mDNS service lists for inbound and outbound filtering using commands.

Define service lists to filter mDNS traffic for both inbound and outbound directions using commands.

Procedure

1. Enable the privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

   Enter your password, if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Configure mDNS service list for inbound filtering.

   Example

   ```
   Device(config)# mdns-sd service-list service-list-name IN
   ```

4. Match the service to the service definition name.

   Example

   ```
   Device(config)# match service-definition-name
   ```

   Here, *service-definition-name* refers to the names of services, such as, airplay, airserver, airtunes, and so on.

   > **Note**
   >
   > To add a service, the service name must be part of the primary service list.
   >
   > The same set of service list is used for both IN and OUT filters.

5. Configure mDNS service list for outbound filtering.

   Example

   ```
   Device(config)# mdns-sd service-list service-list-name OUT
   ```

6. Match the service to the service definition name.

   Example

   ```
   Device(config-mdns-sl-out)# match service-definition-name
   ```

   Here, *service-definition-name* refers to the names of services, such as, airplay, airserver, airtunes, and so on.

   > **Note**
   >
   > To add a service, the service name must be part of the primary service list.
   >
   > The same set of service list is used for both IN and OUT filters.

7. Exit mDNS service list configuration mode.

   Example

   ```
   Device(config-mdns-sl-out)# exit
   ```

Create service policy (CLI)

Use this task to create an mDNS service policy configuration using commands for device management.

Configure an mDNS service policy for traffic management and device discovery in your network using commands.

Procedure

1. Enable the privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

   Enter your password, if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Configure mDNS service policy.

   Example

   ```
   Device(config)# mdns-sd service-policy service-policy-name
   ```

4. Configure service lists for IN and OUT directions.

   Example

   ```
   Device(config-mdns-ser-pol)# service-list service-list-name {in | out}
   ```

5. Enable Location Specific Services (LSS) for the mDNS service.

   Example

   ```
   Device(config-mdns-ser-pol)# location lss
   ```

6. Exit the mDNS service policy configuration mode.

   Example

   ```
   Device(config-mdns-ser-pol)# exit
   ```

Configure an mDNS FlexConnect profile (GUI)

Configure an mDNS FlexConnect profile using the GUI to control service policy and VLAN details.

Set up and apply an mDNS FlexConnect profile to manage multicast DNS (mDNS) service discovery and VLAN filtering on your APs.

Procedure

1. Choose **Configuration** > **Services** > **mDNS**.
2. In the **mDNS Flex Profile** section, click **Add**.
3. In the **Add mDNS Flex Profile** window that is displayed, enter the FlexConnect mDNS profile name in the **Profile Name** field.
4. In the **Service Cache Update Timer** field, specify the service cache update time. The value range is between 1 and 100 minutes.
5. In the **Statistics Update Timer** field, specify the statistics update timer. The value range is between 1 and 100 minutes.
6. In the **VLANs** field, specify the VLAN ID. You can enter multiple VLAN IDs separated by commas or enter a range of VLAN IDs. The maximum number of VLANs allowed is 16.
7. Select a **Wired Service Policy** from the drop-down list to associate a Wired filter to mDNS Flex-Profile. In addition to filtering mDNS service queries based on the static default service list, wired filter will support filtering based on custom service lists.

   The new wired service-policy is added to flex-profile construct to support the custom wired service-policy. The AP applies this configuration for wired services and the respective IN and OUT filters are used for advertisements and queries only if the custom wired service-policy is configured in mDNS flex-profile.

   In case a custom service-policy is removed from the mDNS flex-profile, the AP removes the custom service-policy and apply the default service-policy for wired services. This feature is supported only in locally switched FlexConnect mode.

8. Click **Apply to Device**.

Configure an mDNS FlexConnect profile (CLI)

Configure an mDNS FlexConnect profile using commands to enable device discovery and service advertisement on FlexConnect sites.

Set up an mDNS FlexConnect profile, define service and statistics timers, specify wired VLAN range, and associate a wired service policy for efficient mDNS operation using commands.

Procedure

1. Enable the privileged EXEC mode.

   Example

   ```
   Device> enable
   ```

   Enter your password, if prompted.

2. Enter the global configuration mode.

   Example

   ```
   Device# configure terminal
   ```

3. Configure an mDNS FlexConnect profile.

Example

```
Device(config)# mdns-sd flex-profile mdns-flex-profile-name
```

4. Configure the mDNS update service cache timer for the FlexConnect profile.

   Example

   ```
   Device(config-mdns-flex-prof)# update-timer service-cache timer-value <1-100>
   ```

   The default value is 1 minute. The value range is between 1 minute and 100 minutes.

5. Configure the mDNS update statistics timer for the FlexConnect profile.

   Example

   ```
   Device(config-mdns-flex-prof)# update-timer statistics statistics timer-value
    <1-100>
   ```

   The default value is 1 minute. The valid range is from 1 to 100 minutes.

6. Configure the mDNS wired VLAN range for the FlexConnect profile.

   Example

   ```
   Device(config-mdns-flex-prof)# wired-vlan-range wired-vlan-range value
   ```

   The default value is 1 minute. The valid range is from 1 minute to 100 minutes.

7. Associate the wired service policy with mDNS FlexConnect profile.

   Example

   ```
   Device(config-mdns-flex-prof)# wired-service-policy service-policy-name
   ```

   > **Note**
   >
   > Here, *service-policy-name* refers to the mDNS service policy created earlier. For more information, refer to Creating Service Policy (CLI).

8. Return to the privileged EXEC mode.

   Example

   ```
   Device(config-mdns-flex-prof)# end
   ```

   Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Configure VLAN and MAC based wired service filtering (CLI)

Use this task to configure VLAN and MAC based wired service filtering for mDNS services using commands.

Enable precise filtering of wired services by matching specific MAC addresses and VLANs, improving service visibility and security using commands.

Procedure

1. Enable the privileged EXEC mode and enter the global configuration mode.

   Example

   ```
   Device> enable
   ```

   ```
   Device# configure terminal
   ```

   Enter your password, if prompted.

2. Configure an mDNS wired filter and match the wired filter with the MAC address of the wired service.

   Example

   ```
   Device(config)# mdns-sd wired-filter wired-filter-name
   ```

   ```
   Device(config-mdns-wired-filter)# match mac service-provider-mac-address1
   ```

3. Match the wired filter with the VLAN of the wired service and exit the mDNS gateway configuration mode.

   Example

   ```
   Device(config-mdns-wired-filter)# match vlan range
   ```

   ```
   Device(config-mdns-wired-filter)# exit
   ```

4. Enter the global configuration mode and configure mDNS service list for inbound filtering.

   Example

   ```
   Device# configure terminal
   ```

   ```
   Device(config)# mdns-sd service-list service-list-name IN
   ```

5. Match the service to the names of the services and configure the mDNS service list for outbound filtering.

   Example

   ```
   Device(config)# match service-definition-name
   ```

   ```
   Device(config)# mdns-sd service-list service-list-name OUT
   ```

   Here, *service-definition-name* refers to the names of services, such as, airplay, airserver, airtunes, and so on.

6. Match the Apple TV related wired filter and enable mDNS service policy.

Example

```
Device(config-mdns-sl-out)# match apple-tv wired-filter wired-filter-name
```

```
Device(config)# mdns-sd service-policy service-policy-name
```

7. Configure various service-list names for IN and OUT directions.

Example

```
Device(config-mdns-ser-pol)# service-list service-list-name {IN | OUT}
```

> **Note**
>
> If an administrator creates or uses a custom service policy, the policy must be configured with service-lists for both directions (IN and OUT). The mDNS Gateway will not work if there is no IN service-list, because it will not learn services. If there is no OUT service-list, the gateway will not reply to or announce services it has learned.

8. Configure AP location based filtering.

Example

```
Device(config-mdns-ser-pol)# location ap-group
```

9. Return to the privileged EXEC mode.

Example

```
Device(config-mdns-ser-pol)# end
```

Alternatively, you can also press **Ctrl-Z** to exit global configuration mode.

Verify mDNS-based wired service filtering

This topic lists the commands and outputs used to validate mDNS-based wired service filtering on FlexConnect APs, including verification of VLAN and MAC-based filtering, service policy support, and LSS configuration

To view the wired service list IN and OUT details, use this command:

```
Device# show mdns status
Global mDNS gateway:Enabled
vap_id      ssid mdns_mode
     0  myFisaiC    Bridge
     1 rguestcpC    Bridge
     2   RK-FLEX    Bridge
     3   RK-MDNS   Gateway
     4 GUHOAsaiC    Bridge
     5         -    Bridge
     6         -    Bridge
     7         -    Bridge
     8         -    Bridge
```

```
     9         -    Bridge
    10         -    Bridge
    11         -    Bridge
    12         -    Bridge
    13         -    Bridge
    14         -    Bridge
    15         -    Bridge
Active query interval:30
vap              service_list_in              service_list_out location
  0 default-mdns-service-list_IN default-mdns-service-list_OUT        0
  1 default-mdns-service-list_IN default-mdns-service-list_OUT        0
  2 default-mdns-service-list_IN default-mdns-service-list_OUT        0
  3 default-mdns-service-list_IN default-mdns-service-list_OUT        0
  4 default-mdns-service-list_IN default-mdns-service-list_OUT        0
Wired vlan configuration:
mdns stats timer: 1
mdns cache timer: 1
AP Sync VLAN: 1
Wired service list IN: RK-IN_IN
Wired service list OUT: RK-OUT_OUT
```

Note

Execute this command on the FlexConnect AP. This guidance also applies to the custom wired service policy support in FlexConnect mode.

To verify the VLAN and MAC based wired service filtering, use this command:

```
Device# show running-config mdns-sd wired-filter
mdns-sd wired-filter WIRED_FILTER_APPLE_TV
match mac a886.ddb2.05e9
match vlan 100
!
```

To verify the wired service policy support in FlexConnect Profile, use this command:

```
Device# show running-config mdns-sd flex-profile


               mdns-sd flex-profile custom_flex_profile
update-timer service-cache 15
update-timer statistics 10
wired-vlan-range 30
wired-service-policy custom_wired_policy
!
```

To verify whether LSS is configured or not, use this command:

```
Device# show running-config mdns-sd service-policy
mdns-sd service-policy custom_policy
service-list srvc_lst_in IN
service-list srvc_lst_out OUT
location lss
!
mdns-sd service-list srvc_lst_in IN
match apple-tv
```

```
!
mdns-sd service-list srvc_lst_out OUT
match apple-tv wired-filter WIRED_FILTER_APPLE_TV
!
```