

CLI Book 1: Cisco ASA Series General Operations CLI Configuration Guide, 9.17

First Published: 2021-12-01

Last Modified: 2026-07-19

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
http://www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 527-0883

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)

© 2021–2026 Cisco Systems, Inc. All rights reserved.



CONTENTS

PREFACE

About This Guide li

Document Objectives li

Related Documentation li

Document Conventions li

Communications, Services, and Additional Information liii

PART I

Getting Started with the ASA 55

CHAPTER 1

Introduction to the ASA 1

Hardware and Software Compatibility 1

VPN Compatibility 1

New Features 1

New Features in ASA 9.17(1) 2

Firewall Functional Overview 7

Security Policy Overview 7

Permitting or Denying Traffic with Access Rules 7

Applying NAT 7

Protecting from IP Fragments 7

Applying HTTP, HTTPS, or FTP Filtering 7

Applying Application Inspection 8

Applying QoS Policies 8

Applying Connection Limits and TCP Normalization 8

Enabling Threat Detection 8

Firewall Mode Overview 8

Stateful Inspection Overview 9

VPN Functional Overview 10

Security Context Overview 11
ASA Clustering Overview 11
Special and Legacy Services 11

CHAPTER 2 **Getting Started** 13
Access the Console for the Command-Line Interface 13
Access the ISA 3000 Console 13
Access the Firepower 2100 Platform Mode Console 14
Access the Firepower 1000, 2100 Appliance Mode, and Secure Firewall 3100 Console 16
Access the ASA Console on the Firepower 4100/9300 Chassis 18
Enable FIPS Mode 19
Configure ASDM Access 21
Use the Factory Default Configuration for ASDM Access 21
Customize ASDM Access 21
Start ASDM 24
Factory Default Configurations 25
Restore the Factory Default Configuration 26
Restore the ASAv Deployment Configuration 28
Firepower 1010 Default Configuration 29
Firepower 1100 Default Configuration 31
Firepower 2100 Platform Mode Default Configuration 32
Firepower 2100 Appliance Mode Default Configuration 33
Secure Firewall 3100 Default Configuration 34
Firepower 4100/9300 Chassis Default Configuration 35
ISA 3000 Default Configuration 36
ASAv Deployment Configuration 37
Set the Firepower 2100 to Appliance or Platform Mode 39
Work with the Configuration 41
About the Startup Configuration and the Running Configuration 41
Save Configuration Changes 42
Save Configuration Changes in Single Context Mode 42
Save Configuration Changes in Multiple Context Mode 42
Copy the Startup Configuration to the Running Configuration 44
View the Configuration 44

Clear and Remove Configuration Settings 45
Create Text Configuration Files Offline 46
Apply Configuration Changes to Connections 46
Reload the ASA 47

CHAPTER 3 **Licenses: Product Authorization Key Licensing for the ISA 3000** 49
About PAK Licenses 49
Preinstalled License 49
Permanent License 49
Time-Based Licenses 50
Time-Based License Activation Guidelines 50
How the Time-Based License Timer Works 50
How Permanent and Time-Based Licenses Combine 50
Stacking Time-Based Licenses 51
Time-Based License Expiration 51
License Notes 52
AnyConnect Plus, AnyConnect Apex, and AnyConnect VPN Only Licenses 52
Other VPN License 52
Total VPN Sessions Combined, All Types 52
VPN Load Balancing 53
Legacy VPN Licenses 53
Encryption License 53
Total TLS Proxy Sessions 53
VLANs, Maximum 54
Shared AnyConnect Client Premium Licenses (AnyConnect 3 and Earlier) 54
Failover 54
Failover License Requirements and Exceptions 54
How Failover Licenses Combine 55
Loss of Communication Between Failover Units 55
Upgrading Failover Pairs 55
No Payload Encryption Models 55
Licenses FAQ 56
Guidelines for PAK Licenses 57
Configure PAK Licenses 58

Order License PAKs and Obtain an Activation Key 58
Obtain a Strong Encryption License 59
Activate or Deactivate Keys 62
Configure a Shared License (AnyConnect Client 3 and Earlier) 63
About Shared Licenses 63
About the Shared Licensing Server and Participants 63
Communication Issues Between Participant and Server 64
About the Shared Licensing Backup Server 64
Failover and Shared Licenses 65
Maximum Number of Participants 66
Configure the Shared Licensing Server 67
Configure the Shared Licensing Backup Server (Optional) 68
Configure the Shared Licensing Participant 69
Supported Feature Licenses Per Model 70
Licenses Per Model 70
ISA 3000 License Features 70
Monitoring PAK Licenses 71
Viewing Your Current License 71
Monitoring the Shared License 79
History for PAK Licenses 81

CHAPTER 4 **Licenses: Smart Software Licensing** 87
About Smart Software Licensing 87
Smart Software Licensing for the ASA on the Firepower 4100/9300 Chassis 88
Smart Software Manager and Accounts 88
Offline Management 88
Permanent License Reservation 88
Smart Software Manager On-Prem 92
Licenses and Devices Managed per Virtual Account 92
Evaluation License 92
About Licenses by Type 94
AnyConnect Plus, AnyConnect Apex, and AnyConnect VPN Only Licenses 94
Other VPN Peers 94
Total VPN Peers Combined, All Types 94

Encryption License 95
Carrier License 97
Total TLS Proxy Sessions 97
VLANs, Maximum 98
Botnet Traffic Filter License 98
Failover or ASA Cluster Licenses 98
Failover Licenses for the ASAv 98
Failover Licenses for the Firepower 1010 98
Failover Licenses for the Firepower 1100 99
Failover Licenses for the Firepower 2100 101
Failover Licenses for the Secure Firewall 3100 102
Failover Licenses for the Firepower 4100/9300 103
ASA Cluster Licenses for the Secure Firewall 3100 104
ASA Cluster Licenses for the ASAv 106
ASA Cluster Licenses for the Firepower 4100/9300 106
Prerequisites for Smart Software Licensing 108
Smart Software Manager Regular and On-Prem Prerequisites 108
Permanent License Reservation Prerequisites 108
Guidelines for Smart Software Licensing 109
Defaults for Smart Software Licensing 109
ASAv: Configure Smart Software Licensing 109
ASAv: Configure Regular Smart Software Licensing 109
ASAv: Configure Smart Software Manager On-Prem Licensing 114
ASAv: Configure Utility (MSLA) Smart Software Licensing 117
ASAv: Configure Permanent License Reservation 122
Install the ASAv Permanent License 123
(Optional) Return the ASAv Permanent License 125
(Optional) Deregister the ASAv (Regular and On-Prem) 126
(Optional) Renew the ASAv ID Certificate or License Entitlement (Regular and On-Prem) 126
Firepower 4100/9300: Configure Smart Software Licensing 127
Other Models: Configure Smart Software Licensing 129
Other Models: Configure Regular Smart Software Licensing 130
Other Models: Configure Smart Software Manager On-Prem Licensing 134
Other Models: Configure Permanent License Reservation 137

Install the 1000/2100/3100 Permanent License 137
(Optional) Other Models: Return the Permanent License 140
(Optional) Other Models: Deregister the ASA (Regular and On-Prem) 141
(Optional) Other Models: Renew the ASA ID Certificate or License Entitlement (Regular and On-Prem) 141
Licenses Per Model 142
ASAv 142
Firepower 1010 146
Firepower 1100 Series 147
Firepower 2100 Series 148
Secure Firewall 3100 Series 150
Firepower 4100 151
Firepower 9300 153
Monitoring Smart Software Licensing 154
Viewing Your Current License 154
Viewing Smart License Status 155
Viewing the UDI 157
Debugging Smart Software Licensing 157
Smart Software Manager Communication 158
Device Registration and Tokens 158
Periodic Communication with the Smart Software Manager 158
Out-of-Compliance State 159
Smart Call Home Infrastructure 159
Smart License Certificate Management 159
History for Smart Software Licensing 160

CHAPTER 5 **Logical Devices for the Firepower 4100/9300 163**
About Interfaces 163
Chassis Management Interface 163
Interface Types 164
FXOS Interfaces vs. Application Interfaces 165
About Logical Devices 166
Standalone and Clustered Logical Devices 166
Requirements and Prerequisites for Hardware and Software Combinations 166

Guidelines and Limitations for Logical Devices 167
Guidelines and Limitations for Interfaces 167
General Guidelines and Limitations 168
Requirements and Prerequisites for High Availability 168
Configure Interfaces 168
Configure a Physical Interface 169
Add an EtherChannel (Port Channel) 171
Configure Logical Devices 174
Add a Standalone ASA 174
Add a High Availability Pair 179
Change an Interface on an ASA Logical Device 180
Connect to the Console of the Application 181
History for Logical Devices 183

CHAPTER 6 **Transparent or Routed Firewall Mode** 185
Firewall mode 185
Routed firewall mode 185
Transparent firewall mode 186
Transparent firewall network usage 186
Management Interface 186
Passing traffic for routed-mode features 187
Bridge groups 187
Bridge virtual interfaces 187
Bridge groups in transparent firewall mode 188
Bridge groups in routed firewall mode 188
Passing Traffic Not Allowed in Routed Mode 189
Allow layer 3 traffic 189
Allowed MAC addresses 190
BPDU Handling 190
MAC address and route lookups 190
Unsupported features for bridge groups in transparent mode 192
Unsupported features for bridge groups in routed mode 193
Default settings 194
Guidelines for firewall mode 194

Set the Firewall Mode 195
Examples for Firewall Mode 196
How Data Moves Through the ASA in Routed Firewall Mode 197
An Inside User Visits a Web Server 197
An Outside User Visits a Web Server on the DMZ 198
An Inside User Visits a Web Server on the DMZ 199
An Outside User Attempts to Access an Inside Host 200
A DMZ User Attempts to Access an Inside Host 200
How Data Moves Through the Transparent Firewall 201
An Inside User Visits a Web Server 202
An Inside User Visits a Web Server Using NAT 203
An Outside User Visits a Web Server on the Inside Network 205
An Outside User Attempts to Access an Inside Host 206
History for the Firewall Mode 207

PART II **High Availability and Scalability** 209

CHAPTER 7 **Multiple Context Mode** 211
About Security Contexts 211
Common Uses for Security Contexts 211
Context Configuration Files 212
Context Configurations 212
System Configuration 212
Admin Context Configuration 212
How the ASA Classifies Packets 212
Valid Classifier Criteria 212
Classification Examples 213
Cascading Security Contexts 215
Management Access to Security Contexts 216
System Administrator Access 216
Context Administrator Access 216
Management Interface Usage 216
About Resource Management 217
Resource Classes 217

Resource Limits 217
Default Class 218
Use Oversubscribed Resources 219
Use Unlimited Resources 219
About MAC Addresses 220
MAC Addresses in Multiple Context Mode 220
Automatic MAC Addresses 220
VPN Support 221
Licensing for Multiple Context Mode 221
Prerequisites for Multiple Context Mode 222
Guidelines for Multiple Context Mode 222
Defaults for Multiple Context Mode 224
Configure Multiple Contexts 224
Enable or Disable Multiple Context Mode 224
Enable Multiple Context Mode 224
Restore Single Context Mode 226
Configure a Class for Resource Management 226
Configure a Security Context 230
Assign MAC Addresses to Context Interfaces Automatically 234
Change Between Contexts and the System Execution Space 235
Manage Security Contexts 235
Remove a Security Context 236
Change the Admin Context 236
Change the Security Context URL 237
Reload a Security Context 238
Reload by Clearing the Configuration 239
Reload by Removing and Re-adding the Context 239
Monitoring Security Contexts 239
View Context Information 240
View Resource Allocation 241
View Resource Usage 244
Monitor SYN Attacks in Contexts 246
View Assigned MAC Addresses 249
View MAC Addresses in the System Configuration 249

View MAC Addresses Within a Context 250
Examples for Multiple Context Mode 251
History for Multiple Context Mode 252

CHAPTER 8 **Failover for High Availability** 257
About Failover 257
Failover Modes 257
Failover system requirements 258
Hardware requirements 258
Software requirements 258
License Requirements 259
Failover and stateful failover links 259
Failover link 260
Stateful failover links 262
Avoiding Interrupted Failover and Data Links 262
MAC addresses and IP addresses in Failover 264
Stateless and Stateful Failover 266
Stateless Failover 266
Stateful failover 266
Bridge Group Requirements for Failover 268
Bridge Group Requirements for Appliances, ASAv 268
Failover health monitoring 269
Unit health monitoring 269
Interface monitoring 270
Failover times 272
Configuration Synchronization 273
Running Configuration Replication 273
File Replication 274
Command Replication 274
Active/standby failover 275
Primary/secondary roles and active/standby status 275
Active unit determination at startup 275
Failover events 276
About Active/Active Failover 277

Active/Active Failover Overview 277
Primary/Secondary Roles and Active/Standby Status for a Failover Group 277
Active Unit Determination for Failover Groups at Startup 278
Failover Events 278
Licensing for Failover 279
Guidelines for Failover 280
Defaults for Failover 283
Configure Active/Standby Failover 283
Configure the Primary Unit for Active/Standby Failover 284
Configure the Secondary Unit for Active/Standby Failover 287
Configure Active/Active Failover 288
Configure the Primary Unit for Active/Active Failover 288
Configure the Secondary Unit for Active/Active Failover 293
Configure Optional Failover Parameters 294
Configure Failover Criteria and Other Settings 294
Configure Interface Monitoring 298
Configure Support for Asymmetrically Routed Packets (Active/Active Mode) 298
Manage Failover 302
Force Failover 302
Disable Failover 303
Restore a Failed Unit 304
Re-Sync the Configuration 304
Test the Failover Functionality 305
Remote Command Execution 305
Send a Command 306
Change Command Modes 306
Security Considerations 307
Limitations of Remote Command Execution 307
Monitoring Failover 308
Failover Messages 308
Failover Syslog Messages 308
Failover Debug Messages 308
SNMP Failover Traps 309
Monitoring Failover Status 309

History for Failover 309

CHAPTER 9 **Failover for High Availability in the Public Cloud 313**

About Failover in the Public Cloud 313

About Active/Backup Failover 314

Primary/Secondary Roles and Active/Backup Status 314

Failover Connection 314

Polling and Hello Messages 314

Active Unit Determination at Startup 315

Failover Events 315

Guidelines and Limitations 316

Licensing for Failover in the Public Cloud 317

Defaults for Failover in the Public Cloud 317

About ASAv High Availability in Microsoft Azure 318

About the Azure Service Principal 319

Configuration Requirements for ASAv High Availability in Azure 319

Configure Active/Backup Failover 320

Configure the Primary Unit for Active/Backup Failover 320

Configure the Secondary Unit for Active/Backup Failover 321

Configure Optional Failover Parameters 322

Configure Failover Criteria and Other Settings 322

Configure Authentication Credentials for an Azure Service Principal 324

Configure Azure Route Tables 325

Enable Active/Backup Failover 327

Enable the Primary Unit for Active/Backup Failover 327

Enable the Secondary Unit for Active/Backup Failover 328

Manage Failover in the Public Cloud 329

Force Failover 329

Update Routes 330

Validate Azure Authentication 330

Monitor Failover in the Public Cloud 331

Failover Status 331

Failover Messages 331

History for Failover in the Public Cloud 332

CHAPTER 10 **ASA Cluster for the Secure Firewall 3100** 333

About ASA Clustering 333

How a cluster fits into your network 333

Cluster Members 334

Bootstrap Configuration 334

Control and data node roles 334

Cluster interfaces 334

Cluster control links 334

Configuration replication 335

ASA Cluster Management 335

Management Network 335

Management Interface 335

Control Unit Management Vs. Data Unit Management 336

Crypto Key Replication 336

ASDM Connection Certificate IP Address Mismatch 336

Inter-Site Clustering 336

Licenses for ASA Clustering 337

Requirements and Prerequisites for ASA Clustering 338

Guidelines for Clustering 340

Configure ASA Clustering 345

Cable the Units and Configure Interfaces 345

Cluster interfaces 345

Cable the Cluster Units and Configure Upstream and Downstream Equipment 351

Configure the Cluster Interface Mode on Each Unit 352

Configure Interfaces on the Control Node 353

Create the Bootstrap Configuration 358

Configure the Control Node Bootstrap Settings 359

Configure Data Node Bootstrap Settings 363

Customize the Clustering Operation 366

Configure Basic ASA Cluster Parameters 366

Configure Health Monitoring and Auto-Rejoin Settings 366

Configure Connection Rebalancing and the Cluster TCP Replication Delay 370

Configure Inter-Site Features 371

Manage Cluster Nodes 378
Become an Inactive Node 378
Deactivate a Node 379
Rejoin the Cluster 379
Leave the Cluster 380
Change the Control Node 381
Execute a Command Cluster-Wide 382
Monitoring the ASA Cluster 383
Monitoring Cluster Status 383
Capturing Packets Cluster-Wide 387
Monitoring Cluster Resources 387
Monitoring Cluster Traffic 387
Monitoring Cluster Routing 392
Configuring Logging for Clustering 393
Monitoring Cluster Interfaces 393
Debugging Clustering 394
Examples for ASA Clustering 395
Sample ASA and Switch Configuration 395
ASA Configuration 395
Cisco IOS Switch Configuration 396
Firewall on a stick 397
Traffic segregation 400
OTV Configuration for Routed Mode Inter-Site Clustering 402
Examples for Inter-Site Clustering 405
Spanned EtherChannel Routed Mode Example with Site-Specific MAC and IP Addresses 405
Spanned EtherChannel Transparent Mode North-South Inter-Site Example 406
Spanned EtherChannel Transparent Mode East-West Inter-Site Example 407
Reference for Clustering 408
ASA Features and Clustering 408
Unsupported Features with Clustering 408
Centralized features for clustering 409
Features Applied to Individual Nodes 410
AAA for Network Access and Clustering 410
Connection settings and clustering 411

FTP and Clustering 411
ICMP Inspection and Clustering 411
Multicast Routing and Clustering 411
NAT and clustering 411
Dynamic Routing and Clustering 411
SCTP and Clustering 412
SIP inspection and clustering 413
SNMP and clustering 413
STUN and Clustering 413
Syslog and NetFlow and clustering 413
Cisco TrustSec and clustering 413
VPN and clustering 413
Performance scaling factor 414
Control node election 414
High availability within clusters 415
Node health monitoring 415
Interface monitoring 415
Status after failure 416
Rejoining the Cluster 416
Data path connection state replication 416
How the cluster manages connections 417
Connection roles 417
New connection ownership 419
Sample data flow for TCP 419
Sample data flows for ICMP and UDP 420
Rebalancing New TCP Connections Across the Cluster 421
History for ASA Clustering for the Secure Firewall 3100 422

CHAPTER 11 **ASA Cluster for the Firepower 4100/9300 423**
About Clustering on the Firepower 4100/9300 Chassis 423
Bootstrap Configuration 424
Cluster Members 424
Cluster Control Link 424
Size the cluster control link 425

Cluster control link redundancy 425
Cluster control link reliability 426
Cluster Control Link Network 426
Cluster control link MTU ping testing 426
Cluster Interfaces 427
Connecting to a Redundant Switch System 427
Configuration Replication 427
ASA Cluster Management 427
Management Network 427
Management Interface 427
Control Unit Management Vs. Data Unit Management 428
Crypto Key Replication 428
ASDM Connection Certificate IP Address Mismatch 428
Spanned EtherChannels (Recommended) 428
Inter-Site Clustering 429
Requirements and Prerequisites for Clustering on the Firepower 4100/9300 Chassis 430
Licenses for Clustering on the Firepower 4100/9300 Chassis 431
Clustering Guidelines and Limitations 433
Configure Clustering on the Firepower 4100/9300 Chassis 437
FXOS: Add an ASA Cluster 438
Create an ASA Cluster 438
Add More Cluster Members 447
ASA: Change the Firewall Mode and Context Mode 448
ASA: Configure Data Interfaces 448
ASA: Customize the Cluster Configuration 451
Configure Basic ASA Cluster Parameters 451
Configure Health Monitoring and Auto-Rejoin Settings 454
Configure Connection Rebalancing and the Cluster TCP Replication Delay 457
Configure Inter-Site Features 458
Configure Distributed Site-to-Site VPN 464
FXOS: Remove a Cluster Node 471
ASA: Manage Cluster Members 472
Become an Inactive Member 472
Deactivate a Unit 473

Rejoin the Cluster 474
Change the Control Unit 475
Execute a Command Cluster-Wide 475
ASA: Monitoring the ASA Cluster on the Firepower 4100/9300 chassis 477
Monitoring Cluster Status 477
Capturing Packets Cluster-Wide 481
Monitoring Cluster Resources 481
Monitoring Cluster Traffic 481
Monitoring Cluster Routing 486
Monitoring Distributed S2S VPN 486
Configuring Logging for Clustering 487
Debugging Clustering 487
Troubleshooting Distributed S2S VPN 487
Examples for ASA Clustering 489
Firewall on a stick 489
Traffic Segregation 492
OTV Configuration for Routed Mode Inter-Site Clustering 494
Examples for Inter-Site Clustering 496
Spanned EtherChannel Routed Mode Example with Site-Specific MAC and IP Addresses 497
Spanned EtherChannel Transparent Mode North-South Inter-Site Example 498
Spanned EtherChannel Transparent Mode East-West Inter-Site Example 499
Reference for Clustering 500
ASA Features and Clustering 500
Unsupported Features with Clustering 500
Centralized Features for Clustering 501
Features Applied to Individual Units 502
AAA for Network Access and Clustering 502
Connection Settings 503
FTP and Clustering 503
ICMP Inspection 503
Multicast Routing and Clustering 503
NAT and Clustering 503
Dynamic Routing and Clustering 506
SCTP and Clustering 507

SIP Inspection and Clustering 507
SNMP and Clustering 507
STUN and Clustering 508
Syslog and NetFlow and Clustering 508
Cisco TrustSec and Clustering 508
VPN and Clustering on the Secure Firewall eXtensible Operating System (FXOS) Chassis 508
Performance Scaling Factor 509
Control Unit Election 509
High Availability Within the Cluster 509
Chassis-Application Monitoring 509
Unit Health Monitoring 510
Interface Monitoring 510
Decorator Application Monitoring 510
Status After Failure 510
Rejoining the Cluster 511
Data Path Connection State Replication 511
How the Cluster Manages Connections 512
Connection Roles 512
New Connection Ownership 514
Sample Data Flow for TCP 514
Sample Data Flow for ICMP and UDP 515
Rebalancing New TCP Connections Across the Cluster 516
History for ASA Clustering on the Firepower 4100/9300 516

CHAPTER 12 **ASA Cluster for the ASAv for the Private Cloud** 523
About ASAv Clustering 523
How the Cluster Fits into Your Network 524
Cluster Nodes 524
Bootstrap Configuration 524
Control and data node roles 524
Individual Interfaces 525
Policy-based routing 525
Equal-cost multi-path routing 526
Cluster Control Link 526

Cluster Control Link Traffic Overview 527
Cluster Control Link Failure 527
Cluster control link MTU ping testing 527
Configuration Replication 528
ASAv Cluster Management 528
Management Network 528
Management Interface 528
Control Node Management Vs. Data Node Management 528
Crypto Key Replication 529
ASDM Connection Certificate IP Address Mismatch 529
Inter-Site Clustering 529
Licenses for ASAv Clustering 530
Requirements and Prerequisites for ASAv Clustering 530
Guidelines for ASAv Clustering 530
Configure the ASAv Clustering Using a Day0 Configuration 531
Configure ASAv Clustering after Deployment 534
Configure Interface Settings 534
Configure the Cluster Interface Mode on Each Node 535
Configure Individual Interfaces 536
Create the Bootstrap Configuration 539
Configure Control Node Bootstrap Settings 539
Configure Data Node Bootstrap Settings 544
Customize the Clustering Operation 547
Configure Basic ASA Cluster Parameters 547
Configure Health Monitoring and Auto-Rejoin Settings 548
Configure Connection Rebalancing and the Cluster TCP Replication Delay 550
Configure Inter-Site Features 551
Enable Director Localization 551
Enable Site Redundancy 553
Configure Cluster Flow Mobility 553
Manage Cluster Nodes 558
Become an Inactive Node 558
Deactivate a Data Node from the Control Node 558
Rejoin the Cluster 559

Leave the Cluster 560
Change the Control Node 561
Execute a Command Cluster-Wide 561
Monitoring the ASAv Cluster 562
Monitoring Cluster Status 562
Capturing Packets Cluster-Wide 566
Monitoring Cluster Resources 566
Monitoring Cluster Traffic 567
Monitoring Cluster Routing 572
Configuring Logging for Clustering 572
Monitoring Cluster Interfaces 572
Debugging Clustering 572
Examples for ASAv Clustering 573
Individual Interface Routed Mode North-South Inter-Site Example 573
Reference for Clustering 574
ASA Features and Clustering 574
Unsupported Features with Clustering 574
Centralized Features for Clustering 575
Features Applied to Individual Nodes 576
AAA for Network Access and Clustering 576
Connection Settings and Clustering 576
Dynamic Routing and Clustering 577
FTP and Clustering 578
ICMP Inspection and Clustering 578
Multicast Routing and Clustering 578
NAT and Clustering 578
SCTP and Clustering 580
SIP Inspection and Clustering 580
SNMP and Clustering 580
STUN and Clustering 580
Syslog and NetFlow and Clustering 580
Cisco TrustSec and Clustering 581
VPN and Clustering 581
Performance scaling factor 581

Control Node Election 581
High Availability Within the ASAv Cluster 582
Node health monitoring 582
Interface Monitoring 582
Status After Failure 583
Rejoining the Cluster 583
Data Path Connection State Replication 584
How the ASAv Cluster Manages Connections 584
Connection Roles 584
New connection ownership 586
Sample Data Flow for TCP 586
Sample Data Flow for ICMP and UDP 587
Rebalancing New TCP Connections Across the Cluster 588
History for ASAv Clustering 589

PART III **Interfaces** 591

CHAPTER 13 **Basic Interface Configuration** 593
About Basic Interface Configuration 593
Auto-MDI/MDIX Feature 593
Management Interface 594
Management Interface Overview 594
Management Slot/Port Interface 594
Use Any Interface for Management-Only Traffic 595
Management Interface for Transparent Mode 595
Guidelines for Basic Interface Configuration 596
Default Settings for Basic Interface Configuration 596
Enable the Physical Interface and Configure Ethernet Parameters 597
Enable Jumbo Frame Support (ASAv, ISA 3000) 600
Manage the Network Module for the Secure Firewall 3100 601
Add a Network Module 601
Hot Swap the Network Module 601
Replace the Network Module with a Different Type 602
Remove the Network Module 603

Monitoring Interfaces 604
Examples for Basic Interfaces 604
Physical Interface Parameters Example 605
Multiple Context Mode Example 605
History for Basic Interface Configuration 605

CHAPTER 14 **Basic Interface Configuration for Firepower 1010 Switch Ports** 607
About switch ports 607
Understanding switch ports and interfaces 607
Auto-MDI/MDIX feature 608
Prerequisites for switch ports 608
Guidelines for switch ports 608
Configure switch ports and Power Over Ethernet 610
Enable or disable switch port mode 610
Configure a VLAN interface 611
Configure switch ports as access ports 612
Configure switch ports as trunk ports 614
Configure Power Over Ethernet 616
Monitoring switch ports 618
Examples for Switch Ports 619
Routed Mode Example 619
Transparent Mode Example 619
Mixed Firewall Interface/Switch Port Example 620
Integrated Routing and Bridging Example 621
Failover Example 622
History for switch ports 623

CHAPTER 15 **EtherChannel Interfaces** 625
About EtherChannels 625
About EtherChannels 625
Channel Group Interfaces 626
Connecting to an EtherChannel on Another Device 626
Link Aggregation Control Protocol 627
Load Balancing 627

EtherChannel MAC Address 628
Guidelines for EtherChannels 628
Default Settings for EtherChannels Interfaces 630
Configure an EtherChannel 631
Add Interfaces to the EtherChannel 631
Customize the EtherChannel (ISA 3000) 633
Monitoring EtherChannels 634
Examples for EtherChannels 635
History for EtherChannels 636

CHAPTER 16 VLAN Subinterfaces 637
About VLAN Subinterfaces 637
Licensing for VLAN Subinterfaces 637
Guidelines and Limitations for VLAN Subinterfaces 638
Default Settings for VLAN Subinterfaces 639
Configure VLAN Subinterfaces and 802.1Q Trunking 639
Monitoring VLAN Subinterfaces 641
Examples for VLAN Subinterfaces 641
History for VLAN subinterfaces 642

CHAPTER 17 VXLAN Interfaces 645
About VXLAN Interfaces 645
Encapsulation 645
VXLAN Tunnel Endpoint 646
VTEP Source Interface 646
VNI Interfaces 646
VXLAN Packet Processing 647
Peer VTEP 647
VXLAN Use Cases 648
VXLAN Bridge or Gateway Overview 648
VXLAN Bridge 649
VXLAN Gateway (Routed Mode) 649
Router Between VXLAN Domains 649
AWS Gateway Load Balancer and Geneve Single-Arm Proxy 651

Requirements and Prerequisites for VXLAN Interfaces 651
Guidelines for VXLAN Interfaces 652
Default Settings for VXLAN Interfaces 652
Configure VXLAN Interfaces 653
Configure the VTEP Source Interface 653
Configure the VNI Interface 655
(Optional) Change the VXLAN UDP Port 656
Configure Geneve Interfaces 657
Configure the VTEP Source Interface for Geneve 657
Configure the VNI Interface for Geneve 658
Allow Gateway Load Balancer Health Checks 660
Monitoring VXLAN Interfaces 661
Examples for VXLAN Interfaces 664
Transparent VXLAN Gateway Example 664
VXLAN Routing Example 666
History for VXLAN Interfaces 667

CHAPTER 18 **Routed and Transparent Mode Interfaces 669**
About Routed and Transparent Mode Interfaces 669
Security Levels 669
Dual IP Stack (IPv4 and IPv6) 670
31-Bit Subnet Mask 670
31-Bit Subnet and Clustering 670
31-Bit Subnet and Failover 670
31-Bit Subnet and Management 671
31-Bit Subnet Unsupported Features 671
Guidelines and Limitations for Routed and Transparent Mode Interfaces 671
Configure Routed Mode Interfaces 673
Configure General Routed Mode Interface Parameters 673
Configure PPPoE 676
Configure Bridge Group Interfaces 677
Configure the Bridge Virtual Interface (BVI) 677
Configure General Bridge Group Member Interface Parameters 679
Configure a Management Interface for Transparent Mode 681

Configure IPv6 Addressing 683
About IPv6 683
IPv6 Addressing 683
Modified EUI-64 Interface IDs 683
Configure the IPv6 Prefix Delegation Client 684
About IPv6 Prefix Delegation 684
Enable the IPv6 Prefix Delegation Client 686
Configure a Global IPv6 Address 688
Configure IPv6 Neighbor Discovery 690
Monitoring Routed and Transparent Mode Interfaces 695
Interface Statistics and Information 695
DHCP Information 695
PPPoE 699
IPv6 Neighbor Discovery 699
Examples for Routed and Transparent Mode Interfaces 700
Transparent Mode Example with 2 Bridge Groups 700
Switched LAN Segment Example with 2 Bridge Groups 701
History for Routed and Transparent Mode Interfaces 703

CHAPTER 19 **Advanced Interface Configuration** 707
About Advanced Interface Configuration 707
About MAC Addresses 707
Default MAC Addresses 707
Automatic MAC Addresses 708
About the MTU 709
Path MTU Discovery 709
Default MTU 709
MTU and Fragmentation 709
MTU and Jumbo Frames 710
About the TCP MSS 710
Default TCP MSS 710
Suggested Maximum TCP MSS Setting 710
Inter-Interface Communication 711
Intra-Interface Communication (Routed Firewall Mode) 711

Manually Configure the MAC Address 711
Automatically Assign MAC Addresses 712
Configure the MTU and TCP MSS 713
Allow Same Security Level Communication 715
History for Advanced Interface Configuration 715

CHAPTER 20 **Traffic Zones** 717
About Traffic Zones 717
Non-Zoned Behavior 717
Why Use Zones? 717
Asymmetric Routing 718
Lost Route 718
Load Balancing 719
Per-Zone Connection and Routing Tables 720
ECMP Routing 720
Non-Zoned ECMP Support 720
Zoned ECMP Support 721
How Connections Are Load-Balanced 721
Falling Back to a Route in Another Zone 721
Interface-Based Security Policy 721
Supported Services for Traffic Zones 722
Security Levels 722
Primary and Current Interface for the Flow 722
Joining or Leaving a Zone 722
Intra-Zone Traffic 722
To- and From-the-Box Traffic 723
Overlapping IP Addresses Within a Zone 723
Prerequisites for Traffic Zones 723
Guidelines for Traffic Zones 724
Configure a Traffic Zone 726
Monitoring Traffic Zones 727
Zone Information 727
Zone Connections 727
Zone Routing 728

Example for Traffic Zones 729
History for Traffic Zones 732

PART IV **Basic Settings** 733

CHAPTER 21 **Basic Settings** 735
Set the Hostname, Domain Name, and the Enable and Telnet Passwords 735
Set the Date and Time 737
Set the Time Zone and Daylight Saving Dates 738
Set the Date and Time Using an NTP Server 740
Set the Date and Time Manually 741
Configure Precision Time Protocol (ISA 3000) 742
Configure the Master Passphrase 744
Add or Change the Master Passphrase 744
Disable the Master Passphrase 746
Remove the Master Passphrase 747
Configure the DNS Server 748
Configure the Hardware Bypass and Dual Power Supply (Cisco ISA 3000) 750
Adjust ASP (Accelerated Security Path) Performance and Behavior 752
Choose a Rule Engine Transactional Commit Model 752
Enable ASP Load Balancing 753
Monitoring the DNS Cache 754
History for Basic Settings 754

CHAPTER 22 **DHCP and DDNS Services** 759
About DHCP and DDNS services 759
DHCPv4 servers 759
DHCP options 759
DHCPv6 stateless server 760
DHCP relay agents 761
DHCP Relay Server Support on VTI 761
Guidelines for DHCP and DDNS services 762
Configure the DHCP Server 763
Enable the DHCPv4 Server 764

Configure Advanced DHCPv4 Options 766
Configure the DHCPv6 Stateless Server 767
Configure the DHCP Relay Agent 769
Configure the DHCPv4 Relay Agent 770
Configure the DHCPv6 Relay Agent 771
Configure Dynamic DNS 772
Monitoring DHCP and DDNS Services 778
Monitoring DHCP Services 778
Troubleshooting DHCP Relay over VTI 781
Monitoring DDNS Status 781
History for DHCP and DDNS Services 783

CHAPTER 23 **Digital Certificates** 787
About Digital Certificates 787
Public Key Cryptography 788
Certificate Scalability 788
Key Pairs 789
Trustpoints 789
Certificate Enrollment 790
Proxy for SCEP Requests 790
Revocation Checking 790
Supported CA Servers 790
CRLs 791
OCSP 792
Certificates and User Login Credentials 793
User Login Credentials 793
Certificates 793
Guidelines for Digital Certificates 794
Configure Digital Certificates 797
Configure Key Pairs 797
Configure Trustpoints 799
Configure CRLs for a Trustpoint 804
Export or Import a Trustpoint Configuration 807
Configure CA Certificate Map Rules 808

Configure Reference Identities 811
Obtain Certificates Manually 812
Obtain Certificates Automatically with SCEP 814
Configure Proxy Support for SCEP Requests 816
Configure CAs 818
Configure Auto-Import of Trustpool Certificates 818
Set a Certificate Expiration Alert (for Identity or CA Certificates) 819
Monitoring Digital Certificates 820
History for Certificate Management 822

CHAPTER 24 **ARP Inspection and the MAC Address Table** 825
About ARP Inspection and the MAC Address Table 825
ARP Inspection for Bridge Group Traffic 825
MAC Address Table 826
Default Settings 826
Guidelines for ARP Inspection and the MAC Address Table 826
Configure ARP Inspection and Other ARP Parameters 827
Add a Static ARP Entry and Customize Other ARP Parameters 827
Enable ARP Inspection 828
Customize the MAC Address Table for Bridge Groups 829
Add a Static MAC Address for Bridge Groups 829
Set the MAC Address Timeout 830
Configure MAC Address Learning 830
Monitoring ARP Inspection and the MAC Address Table 831
History for ARP Inspection and the MAC Address Table 831

PART V **IP Routing** 835

CHAPTER 25 **Routing Overview** 837
Path Determination 837
Supported Route Types 838
Static Versus Dynamic 838
Single-Path Versus Multipath 838
Flat Versus Hierarchical 838

Link-State Versus Distance Vector 839
Supported Internet Protocols for Routing 839
Routing GRE Traffic 840
Routing Table 840
How the Routing Table Is Populated 840
Administrative Distances for Routes 841
Backup Dynamic and Floating Static Routes 842
How Forwarding Decisions Are Made 842
Dynamic Routing and Failover 843
Dynamic Routing and Clustering 843
Dynamic Routing in Spanned EtherChannel Mode 843
Dynamic Routing in Individual Interface Mode 844
Dynamic Routing in Multiple Context Mode 845
Route Resource Management 846
Routing Table for Management Traffic 846
Management Interface Identification 847
Equal-Cost Multi-Path (ECMP) Routing 848
Disable Proxy ARP Requests 848
Display the Routing Table 849
History for Route Overview 849

CHAPTER 26 **Static and Default Routes** 851
About Static and Default Routes 851
Default Route 851
Static Routes 851
Route to null0 Interface to Drop Unwanted Traffic 852
Route Priorities 852
Transparent Firewall Mode and Bridge Group Routes 852
Static Route Tracking 852
Guidelines for Static and Default Routes 853
Configure Default and Static Routes 854
Configure a Default Route 854
Configure a Static Route 855
Configure Static Route Tracking 856

Monitoring a Static or Default Route 858
Examples for Static or Default Routes 858
History for Static and Default Routes 858

CHAPTER 27 **Policy Based Routing 861**
About Policy Based Routing 861
Why Use Policy Based Routing? 861
Equal-Access and Source-Sensitive Routing 862
Quality of Service 862
Cost Saving 862
Load Sharing 863
Implementation of PBR 863
Guidelines for Policy Based Routing 863
Configure Policy Based Routing 864
Examples for Policy Based Routing 868
Examples for Route Map Configuration 868
Example Configuration for PBR 869
Direct Internet Access using Software-Defined WAN 870
Policy Based Routing in Action 873
History for Policy Based Routing 877

CHAPTER 28 **Route Maps 879**
About Route Maps 879
Permit and Deny Clauses 880
Match and Set Clause Values 880
Guidelines for Route Maps 881
Define a Route Map 881
Customize a Route Map 881
Define a Route to Match a Specific Destination Address 881
Configure the Metric Values for a Route Action 883
Example for Route Maps 883
History for Route Maps 884

CHAPTER 29 **Bidirectional Forwarding Detection Routing 885**

About BFD Routing 885
BFD Asynchronous Mode and Echo Function 885
BFD Session Establishment 886
BFD Timer Negotiation 887
BFD Failure Detection 888
BFD Deployment Scenarios 888
Guidelines for BFD Routing 888
Configure BFD 889
Create the BFD Template 889
Configure BFD Interfaces 891
Configure BFD Maps 892
Monitoring for BFD 893
History for BFD Routing 894

CHAPTER 30 BGP 895
BGP 895
BGP usage scenarios 895
Routing table changes 896
BGP path selection 897
BGP multipath 898
Guidelines for BGP 899
Configure BGP 900
Enable BGP 900
Define the Best Path for a BGP Routing Process 902
Configure Policy Lists 903
Configure AS Path Filters 904
Configure Community Rules 904
Configure IPv4 Address Family Settings 905
Configure IPv4 Family General Settings 905
Configure IPv4 Family Aggregate Address Settings 908
Configure IPv4 Family Filtering Settings 909
Configure IPv4 Family BGP Neighbor Settings 910
Configure IPv4 Network Settings 915
Configure IPv4 Redistribution Settings 916

Configure IPv4 Route Injection Settings 917
Configure IPv6 Address Family Settings 918
Configure IPv6 Family General Settings 918
Configure IPv6 Family Aggregate Address Settings 919
Configure IPv6 Family BGP Neighbor Settings 920
Configure IPv6 Network Settings 926
Configure IPv6 Redistribution Settings 927
Configure IPv6 Route Injection Settings 928
Monitoring BGP 929
Example for BGP 931
History for BGP 934

CHAPTER 31 OSPF 937
OSPF routing 937
OSPF support for fast hello packets 939
OSPF fast hello packets configuration prerequisites 939
About OSPF Support for Fast Hello Packets 939
OSPFv2 and OSPFv3 implementation differences 940
Guidelines for OSPF 941
Configure OSPFv2 944
Configure a Key Chain for Authentication 945
Configure OSPFv2 Router ID 947
Manually Configure OSPF Router-ID 947
Router ID Behaviour while Migrating 948
Configure OSPF Fast Hello Packets 948
Customize OSPFv2 949
Redistribute Routes Into OSPFv2 949
Configure Route Summarization When Redistributing Routes Into OSPFv2 951
Add a Route Summary Address 951
Configure Route Summarization Between OSPFv2 Areas 952
Configure OSPFv2 Interface Parameters 953
Configure OSPFv2 Area Parameters 956
Configure OSPFv2 Filter Rules 957
Configure an OSPFv2 NSSA 957

Configure an IP Address Pool for Clustering (OSPFv2 and OSPFv3) 959
Define Static OSPFv2 Neighbors 959
Configure Route Calculation Timers 960
Log Neighbors Going Up or Down 961
Configure a Key Chain for Authentication 961
Configure OSPFv3 964
Enable OSPFv3 964
Configure OSPFv3 Interface Parameters 965
Configure OSPFv3 Router Parameters 970
Configure OSPFv3 Area Parameters 973
Configure OSPFv3 Passive Interfaces 975
Configure OSPFv3 Administrative Distance 976
Configure OSPFv3 Timers 976
Define Static OSPFv3 Neighbors 978
Reset OSPFv3 Default Parameters 979
Send Syslog Messages 980
Suppress Syslog Messages 981
Calculate Summary Route Costs 981
Generate a Default External Route into an OSPFv3 Routing Domain 982
Configure an IPv6 Summary Prefix 983
Redistribute IPv6 Routes 983
Configure Graceful Restart 985
Configure Capabilities 985
Configuring Graceful Restart for OSPFv2 986
Configure Cisco NSF Graceful Restart for OSPFv2 986
Configure IETF NSF Graceful Restart for OSPFv2 987
Configuring Graceful Restart for OSPFv3 987
Configuring Graceful Restart Wait Timer for OSPF 988
Remove the OSPFv2 Configuration 989
Remove the OSPFv3 Configuration 990
Example for OSPFv2 990
Examples for OSPFv3 991
Monitoring OSPF 993
History for OSPF 996

CHAPTER 32 IS-IS 999

About IS-IS 999

About NET 999

IS-IS Dynamic Hostname 1000

IS-IS PDU Types 1000

Operation of IS-IS on Multiaccess Circuits 1001

IS-IS Election of the Designated IS 1002

IS-IS LSPDB Synchronization 1003

IS-IS Shortest Path Calculation 1004

IS-IS Shutdown Protocol 1005

Prerequisites for IS-IS 1005

Guidelines for IS-IS 1005

Configure IS-IS 1006

Enable IS-IS Routing Globally 1006

Enable IS-IS Authentication 1010

Configure IS-IS LSP 1013

Configure IS-IS Summary Addresses 1017

Configure IS-IS Passive Interfaces 1018

Configure IS-IS Interfaces 1019

Configure IS-IS Interface Hello Padding 1023

Configure IS-IS IPv4 Address Family 1026

Configure IS-IS IPv6 Address Family 1031

Monitoring IS-IS 1036

History for IS-IS 1038

Examples for IS-IS 1038

CHAPTER 33 EIGRP 1049

About EIGRP 1049

Guidelines for EIGRP 1050

Configure EIGRP 1051

Enable EIGRP 1051

Enable EIGRP Stub Routing 1052

Customize EIGRP 1054

Define a Network for an EIGRP Routing Process 1054
Configure Interfaces for EIGRP 1054
Configure Passive Interfaces 1056
Configure the Summary Aggregate Addresses on Interfaces 1057
Change the Interface Delay Value 1058
Enable EIGRP Authentication on an Interface 1059
Define an EIGRP Neighbor 1060
Redistribute Routes Into EIGRP 1061
Filter Networks in EIGRP 1063
Customize the EIGRP Hello Interval and Hold Time 1064
Disable Automatic Route Summarization 1065
Configure Default Information in EIGRP 1066
Disable EIGRP Split Horizon 1067
Restart the EIGRP Process 1068
Monitoring for EIGRP 1068
Example for EIGRP 1069
History for EIGRP 1070

CHAPTER 34 **Multicast Routing** 1071
Multicast routing 1071
Stub multicast routing 1071
PIM multicast routing 1072
PIM sparse mode 1072
PIM source specific multicast support 1072
PIM bootstrap router (BSR) 1073
PIM bootstrap router (BSR) terminology 1073
Multicast group concept 1074
Multicast Addresses 1074
Clustering 1074
Guidelines for multicast routing 1075
Enable Multicast Routing 1075
Customize Multicast Routing 1076
Configure Stub Multicast Routing and Forward IGMP Messages 1076
Configure a Static Multicast Route 1077

Configure IGMP Features 1078
Disable IGMP on an Interface 1078
Configure IGMP Group Membership 1078
Configure a Statically Joined IGMP Group 1079
Control Access to Multicast Groups 1079
Limit the Number of IGMP States on an Interface 1080
Modify the Query Messages to Multicast Groups 1081
Change the IGMP Version 1082
Configure PIM Features 1082
Enable and Disable PIM on an Interface 1083
Configure a Static Rendezvous Point Address 1083
Configure the Designated Router Priority 1084
Configure and Filter PIM Register Messages 1084
Configure PIM Message Intervals 1085
Filter PIM Neighbors 1085
Configure a Bidirectional Neighbor Filter 1086
Configure the ASA as a Candidate BSR 1087
Configure a Multicast Boundary 1088
Monitoring for PIM 1088
Example for Multicast Routing 1089
History for Multicast Routing 1090

PART VI **AAA Servers and the Local Database** 1091

CHAPTER 35 **AAA and the Local Database** 1093
About AAA and the Local Database 1093
Authentication 1093
Authorization 1094
Accounting 1094
Interaction Between Authentication, Authorization, and Accounting 1094
AAA Servers and Server Groups 1094
About the Local Database 1096
Fallback Support 1097
How Fallback Works with Multiple Servers in a Group 1097

Guidelines for the Local Database 1098
Add a User Account to the Local Database 1098
Monitoring the Local Database 1100
History for the Local Database 1101

CHAPTER 36 RADIUS Servers for AAA 1105
About RADIUS Servers for AAA 1105
Supported Authentication Methods 1105
User Authorization of VPN Connections 1106
Supported Sets of RADIUS Attributes 1106
Supported RADIUS Authorization Attributes 1106
Supported IETF RADIUS Authorization Attributes 1114
RADIUS Accounting Disconnect Reason Codes 1115
Guidelines for RADIUS Servers for AAA 1116
Configure RADIUS Servers for AAA 1116
Configure RADIUS Server Groups 1117
Add a RADIUS Server to a Group 1120
Monitoring RADIUS Servers for AAA 1123
History for RADIUS Servers for AAA 1124

CHAPTER 37 TACACS+ Servers for AAA 1125
About TACACS+ Servers for AAA 1125
TACACS+ Attributes 1125
Guidelines for TACACS+ Servers for AAA 1126
Configure TACACS+ Servers 1127
Configure TACACS+ Server Groups 1127
Add a TACACS+ Server to a Group 1129
Monitoring TACACS+ Servers for AAA 1130
History for TACACS+ Servers for AAA 1130

CHAPTER 38 LDAP Servers for AAA 1133
About LDAP and the ASA 1133
How Authentication Works with LDAP 1133
LDAP Hierarchy 1134

Search the LDAP Hierarchy 1134
Bind to an LDAP Server 1135
LDAP Attribute Maps 1136
Guidelines for LDAP Servers for AAA 1137
Configure LDAP Servers for AAA 1137
Configure LDAP Attribute Maps 1138
Configure LDAP Server Groups 1139
Configure Authorization with LDAP for VPN 1142
Monitoring LDAP Servers for AAA 1143
History for LDAP Servers for AAA 1144

CHAPTER 39 **Kerberos Servers for AAA** 1145
Guidelines for Kerberos Servers for AAA 1145
Configure Kerberos Servers for AAA 1145
Configure Kerberos AAA Server Groups 1145
Add Kerberos Servers to a Kerberos Server Group 1147
Configure Kerberos Key Distribution Center Validation 1148
Monitor Kerberos Servers for AAA 1149
History for Kerberos Servers for AAA 1150

CHAPTER 40 **RSA SecurID Servers for AAA** 1153
About RSA SecurID Servers 1153
Guidelines for RSA SecurID Servers for AAA 1153
Configure RSA SecurID Servers for AAA 1154
Configure RSA SecurID AAA Server Groups 1154
Add RSA SecurID Servers to an SDI Server Group 1155
Import the SDI Node Secret File 1156
Monitor RSA SecurID Servers for AAA 1157
History for RSA SecurID Servers for AAA 1157

PART VII **System Administration** 1159

CHAPTER 41 **Management Access** 1161
Configure Management Remote Access 1161

Configure SSH Access 1161
Enable the SSH Server 1162
Configure SSH for Password Access 1165
Configure SSH for Public Key Access 1166
Configure Telnet Access 1170
Configure HTTPS Access for ASDM, Other Clients 1171
Configure HTTP Redirect for ASDM Access or Clientless SSL VPN 1174
Configure Management Access Over a VPN Tunnel 1175
Configure Management Access for FXOS on Firepower 2100 Platform Mode Data Interfaces 1176
Change the Console Timeout 1177
Customize a CLI Prompt 1178
Configure a Login Banner 1179
Set a Management Session Quota 1180
Configure AAA for System Administrators 1181
Configure Management Authentication 1181
About Management Authentication 1181
Configure Authentication for CLI and ASDM Access 1183
Configure Enable Authentication (Privileged EXEC Mode) 1184
Configure ASDM Certificate Authentication (Java Web Start) 1185
Control CLI and ASDM Access with Management Authorization 1187
Configure Command Authorization 1189
About Command Authorization 1189
Configure Local Command Authorization 1191
Configure Commands on the TACACS+ Server 1193
Configure TACACS+ Command Authorization 1196
Configure a Password Policy for Local Database Users 1197
Change Your Password 1199
Enable and View the Login History 1200
Configure Management Access Accounting 1201
Recover from a Lockout 1201
Monitoring Device Access 1202
History for Management Access 1205

CHAPTER 42 **Software and Configurations** 1213

Upgrade the Software 1213
Load an Image Using ROMMON (ISA 3000) 1213
Upgrade the ROMMON Image: ISA 3000 1215
Downgrade Your Software 1216
Guidelines and Limitations for Downgrading 1216
Incompatible Configuration Removed After Downgrading 1217
Downgrade the ASA Appliance 1217
Downgrade the Firepower 2100 in Platform Mode 1218
Downgrade the Firepower 4100/9300 1219
Downgrade the ISA 3000 1220
Manage Files 1221
View Files in Flash Memory 1221
Delete Files from Flash Memory 1221
Erase the Flash File System 1222
Configure File Access 1222
Configure the FTP Client Mode 1222
Configure the ASA Secure Copy Client 1223
Configure the ASA TFTP Client Path 1225
Copy a File to the ASA 1225
Copy a File to the Startup or Running Configuration 1228
Set the ASA Image, ASDM, and Startup Configuration 1230
Back Up and Restore Configurations or Other Files 1233
Perform a Complete System Backup or Restoration 1233
Before You Begin Backup or Restore 1233
Back Up the System 1235
Restore the Backup 1236
Configure Automatic Backup and Restore (ISA 3000) 1237
Configure Automatic Backup (ISA 3000) 1238
Configure Automatic Restore (ISA 3000) 1239
Back up the Single Mode Configuration or Multiple Mode System Configuration 1240
Back Up a Context Configuration or Other File in Flash Memory 1241
Back Up a Context Configuration within a Context 1243
Copy the Configuration from the Terminal Display 1243
Back Up Additional Files Using the Export and Import Commands 1244

Use a Script to Back Up and Restore Files 1244
Before You Begin Using Backup and Restore Scripts 1245
Run the Script 1245
Sample Script 1245
Hot Swap an SSD on the Secure Firewall 3100 1250
History for Software and Configurations 1253

CHAPTER 43 Response Automation for System Events 1255
About the EEM 1255
Supported Events 1255
Actions on Event Manager Applets 1256
Output Destinations 1256
Guidelines for the EEM 1256
Configure the EEM 1257
Create an Event Manager Applet and Configure Events 1257
Configure an Action and Destinations for Output from an Action 1259
Run an Event Manager Applet 1261
Track Memory Allocation and Memory Usage 1261
Examples for the EEM 1264
Monitoring the EEM 1265
History for the EEM 1266

CHAPTER 44 Testing and Troubleshooting 1267
Recover Enable and Telnet Passwords 1267
Recover Passwords on the ISA 3000 1267
Recover Passwords or Images on the ASAv 1269
Disable Password Recovery for ISA 3000 Hardware 1270
View Debugging Messages 1271
Packet Capture 1271
Guidelines for Packet Capture 1271
Capture Packets 1272
View a Packet Capture 1275
View the Crash Dump 1277
View the Coredump 1277

CPU Usage and Reporting 1277
vCPU Usage in the ASAv 1277
CPU Usage Example 1278
VMware CPU Usage Reporting 1278
ASAv and vCenter Graphs 1278
Amazon CloudWatch CPU Usage Reporting 1279
ASAv and Amazon CloudWatch Graphs 1279
Azure CPU Usage Reporting 1280
ASAv and Azure Graphs 1280
Hyper-V CPU Usage Reporting 1281
ASA Virtual and Hyper-V Graphs 1281
OCI CPU Usage Reporting 1282
ASAv and OCI Graphs 1282
Test Your Configuration 1282
Test Basic Connectivity: Pinging Addresses 1282
What You Can Test Using Ping 1282
Choosing Between ICMP and TCP Ping 1283
Enable ICMP 1283
Ping Hosts 1284
Test ASA Connectivity Systematically 1286
Trace Routes to Hosts 1289
Make the ASA Visible on Trace Routes 1289
Determine Packet Routes 1291
Using the Packet Tracer to Test Policy Configuration 1292
Monitoring Connections 1295
History for Testing and Troubleshooting 1295

PART VIII **Monitoring** 1297

CHAPTER 45 **Logging** 1299
About Logging 1299
Logging in Multiple Context Mode 1300
Syslog Message Analysis 1300
Syslog Message Format 1300

Syslog message severity levels 1301
Syslog message filters 1302
Syslog message classes and associated message ID numbers 1302
Custom Message Lists 1305
Clustering 1306
Guidelines for logging 1306
Configure Logging 1308
Enable Logging 1308
Configure an Output Destination 1309
Send Syslog Messages to an External Syslog Server 1309
Send Syslog Messages to the Internal Log Buffer 1312
Send Syslog Messages to an E-mail Address 1314
Send Syslog Messages to ASDM 1315
Send Syslog Messages to the Console Port 1316
Send Syslog Messages to an SNMP Server 1317
Send Syslog Messages to a Telnet or SSH Session 1317
Configure Syslog Messages 1318
Show or Hide Invalid Usernames in Syslogs 1318
Include the Date and Time in Syslog Messages 1318
Disable a Syslog Message 1319
Change the Severity Level of a Syslog Message 1319
Block Syslog Messages on a Standby Unit 1319
Include the Device ID in Non-EMBLEM Format Syslog Messages 1320
Create a Custom Event List 1321
Configure Logging Filters 1322
Send All Syslog Messages in a Class to a Specified Output Destination 1322
Limit the Rate of Syslog Message Generation 1323
Monitoring the Logs 1323
Examples for Logging 1324
History for Logging 1325

CHAPTER 46 SNMP 1327
About SNMP 1327
SNMP Terminology 1328

MIBs and Traps 1328
SNMP Object Identifiers 1330
Physical Vendor Type Values 1333
Supported Tables and Objects in MIBs 1333
Supported Traps (Notifications) 1335
Interface Types and Examples 1340
SNMP Version 3 Overview 1341
Security Models 1342
SNMP Groups 1342
SNMP Users 1342
SNMP Hosts 1342
Implementation Differences Between the ASA and Cisco IOS Software 1342
SNMP Syslog Messaging 1343
Application Services and Third-Party Tools 1343
Guidelines for SNMP 1343
Configure SNMP 1348
Enable the SNMP Agent and SNMP Server 1348
Configure SNMP Traps 1348
Configure a CPU Usage Threshold 1350
Configure a Physical Interface Threshold 1350
Configure Parameters for SNMP Version 1 or 2c 1351
Configure Parameters for SNMP Version 3 1352
Configure a Group of Users 1356
Associate Users with a Network Object 1356
Monitoring SNMP 1357
Examples for SNMP 1358
History for SNMP 1359

CHAPTER 47 **Cisco Success Network and Telemetry Data** 1365
About Cisco Success Network 1365
Supported Platforms and Required Configurations 1365
How Does ASA Telemetry Data Reach the SSE Cloud 1366
Enable or Disable Cisco Success Network 1366
View ASA Telemetry Data 1367

Cisco Success Network - Telemetry Data 1367
Debug Telemetry Data 1373

CHAPTER 48 Alarms for the Cisco ISA 3000 1375
About alarms 1375
Alarm input interfaces 1376
Alarm output interface 1376
Defaults for alarms 1377
Configure Alarms 1377
Monitoring Alarms 1380
History for Alarms 1382

CHAPTER 49 Anonymous Reporting and Smart Call Home 1383
About Anonymous Reporting 1383
DNS Requirement 1384
About Smart Call Home 1384
Subscribe to Alert Groups 1385
Attributes of Alert Groups 1385
Messages Sent to Cisco by Alert Groups 1386
Message Severity Threshold 1388
Subscription Profiles 1389
Guidelines for Anonymous Reporting and Smart Call Home 1390
Configure Anonymous Reporting and Smart Call Home 1391
Configure Anonymous Reporting 1391
Configure Smart Call Home 1392
Enable Smart Call Home 1392
Declare and Authenticate a Certificate Authority Trust Point 1393
Configure the Environment and Snapshot Alert Groups 1394
Configure Alert Group Subscription 1395
Configure Customer Contact Information 1396
Configure the Mail Server 1397
Configure Traffic Rate Limiting 1398
Send Smart Call Home Communications 1399
Configure a Destination Profile 1399

Copy a Destination Profile 1401
Rename a Destination Profile 1401
Monitoring Anonymous Reporting and Smart Call Home 1402
Examples for Smart Call Home 1403
History for Anonymous Reporting and Smart Call Home 1404

PART IX **Reference** 1407

CHAPTER 50 **Using the Command-Line Interface** 1409
Firewall Mode and Security Context Mode 1409
Command Modes and Prompts 1410
Syntax Formatting 1411
Abbreviate Commands 1412
Command-Line Editing 1412
Command Completion 1412
Command Help 1412
View the Running Configuration 1413
Filter show and more Command Output 1413
Redirecting and Appending show Command Output 1414
Getting a Line Count for show Command Output 1414
Command Output Paging 1415
Add Comments 1415
Text Configuration Files 1416
How Commands Correspond with Lines in the Text File 1416
Command-Specific Configuration Mode Commands 1416
Automatic Text Entries 1416
Line Order 1416
Commands Not Included in the Text Configuration 1417
Passwords 1417
Multiple Security Context Files 1417
Supported Character Sets 1417

CHAPTER 51 **Addresses, Protocols, and Ports** 1419
IPv4 Addresses and Subnet Masks 1419

Classes 1419
Private Networks 1420
Subnet Masks 1420
Determine the Subnet Mask 1420
Determine the Address to Use with the Subnet Mask 1421
IPv6 Addresses 1423
IPv6 Address Format 1423
IPv6 Address Types 1424
Unicast Addresses 1424
Multicast Address 1426
Anycast Address 1427
Required Addresses 1427
IPv6 Address Prefixes 1428
Protocols and Applications 1428
TCP and UDP Ports 1429
Local Ports and Protocols 1433
ICMP Types 1434



About This Guide

The following topics explain how to use this guide.

- Document Objectives, on page li
- Related Documentation, on page li
- Document Conventions, on page li
- Communications, Services, and Additional Information, on page liii

Document Objectives

The purpose of this guide is to help you configure general operations for the Cisco ASA series using the command-line interface. This guide does not cover every feature, but describes only the most common configuration scenarios.

You can also configure and monitor the ASA by using the Adaptive Security Device Manager (ASDM), a web-based GUI application. ASDM includes configuration wizards to guide you through some common configuration scenarios, and online help for less common scenarios.

Throughout this guide, the term "ASA" applies generically to supported models, unless specified otherwise.

Related Documentation

For more information, see *Navigating the Cisco ASA Series Documentation* at http://www.cisco.com/go/asadocs.

Document Conventions

This document adheres to the following text, display, and alert conventions.

Text Conventions

Convention	Indication
boldface	Commands, keywords, button labels, field names, and user-entered text appear in **boldface**. For menu-based commands, the full path to the command is shown.

Convention	Indication
italic	Variables, for which you supply values, are presented in an *italic* typeface. Italic type is also used for document titles, and for general emphasis.
`monospace`	Terminal sessions and information that the system displays appear in `monospace` type.
{x | y | z}	Required alternative keywords are grouped in braces and separated by vertical bars.
[ ]	Elements in square brackets are optional.
[x | y | z]	Optional alternative keywords are grouped in square brackets and separated by vertical bars.
[ ]	Default responses to system prompts are also in square brackets.
< >	Non-printing characters such as passwords are in angle brackets.
!, #	An exclamation point (!) or a number sign (#) at the beginning of a line of code indicates a comment line.

Reader Alerts

This document uses the following for reader alerts:



Note Means *reader take note*. Notes contain helpful suggestions or references to material not covered in the manual.



Tip Means *the following information will help you solve a problem.*



Caution Means *reader be careful*. In this situation, you might do something that could result in equipment damage or loss of data.



Timesaver Means *the described action saves time.* You can save time by performing the action described in the paragraph.



Warning **Means *reader be warned.* In this situation, you might perform an action that could result in bodily injury.**

Communications, Services, and Additional Information

- To receive timely, relevant information from Cisco, sign up at Cisco Profile Manager.
- To get the business results you're looking for with the technologies that matter, visit Cisco Services.
- To submit a service request, visit Cisco Support.
- To discover and browse secure, validated enterprise-class apps, products, solutions and services, visit Cisco Marketplace.
- To obtain general networking, training, and certification titles, visit Cisco Press.
- To find warranty information for a specific product or product family, access Cisco Warranty Finder.

Cisco Bug Search Tool

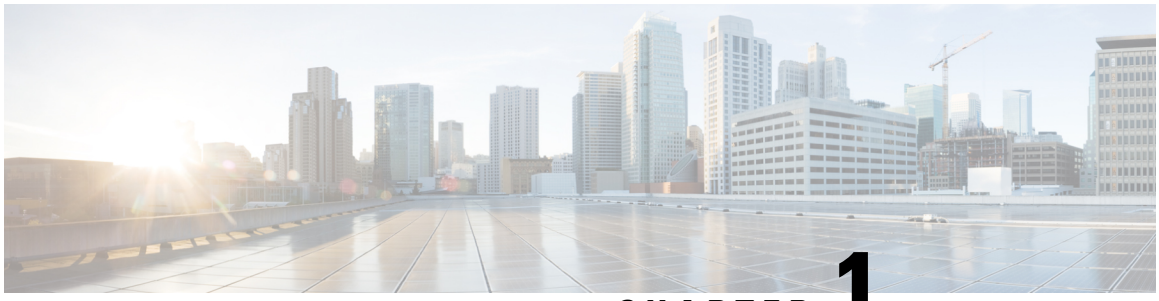
Cisco Bug Search Tool (BST) is a web-based tool that acts as a gateway to the Cisco bug tracking system that maintains a comprehensive list of defects and vulnerabilities in Cisco products and software. BST provides you with detailed defect information about your products and software.



PART I

Getting Started with the ASA

- Introduction to the ASA, on page 1
- Getting Started, on page 13
- Licenses: Product Authorization Key Licensing for the ISA 3000, on page 49
- Licenses: Smart Software Licensing, on page 87
- Logical Devices for the Firepower 4100/9300, on page 163
- Transparent or Routed Firewall Mode, on page 185



CHAPTER 1

Introduction to the ASA

The ASA provides advanced stateful firewall and VPN concentrator functionality in one device. The ASA includes many advanced features, such as multiple security contexts (similar to virtualized firewalls), clustering (combining multiple firewalls into a single firewall), transparent (Layer 2) firewall or routed (Layer 3) firewall operation, advanced inspection engines, IPsec VPN, SSL VPN, and clientless SSL VPN support, and many more features.

- Hardware and Software Compatibility, on page 1
- VPN Compatibility, on page 1
- New Features, on page 1
- Firewall Functional Overview, on page 7
- VPN Functional Overview, on page 10
- Security Context Overview, on page 11
- ASA Clustering Overview, on page 11
- Special and Legacy Services, on page 11

Hardware and Software Compatibility

For a complete list of supported hardware and software, see Cisco ASA Compatibility.

VPN Compatibility

See Supported VPN Platforms, Cisco ASA Series.

New Features

This section lists new features for each release.

Note New, changed, and deprecated syslog messages are listed in the syslog message guide.

New Features in ASA 9.17(1)

Released: December 1, 2021

Feature	Description
Platform Features	
Secure Firewall 3100	We introduced the ASA for the Secure Firewall 3110, 3120, 3130, and 3140. The Secure Firewall 3100 supports up to 8 units for Spanned EtherChannel clustering. You can hot swap a network module of the same type while the firewall is powered up without having to reboot; making other module changes requires a reboot. Secure Firewall 3100 25 Gbps interfaces support Forward Error Correction as well as speed detection based on the SFP installed. The SSDs are self-encrypting drives (SEDs), and if you have 2 SSDs, they form a software RAID. New/Modified commands: **fec, lacp rate, netmod, speed sfp-detect, raid, show raid, show ssd**
ASAv support for Autoscale	The ASAv now supports Autoscale for the following Public Cloud offerings: • Google Cloud Platform (GCP) • Oracle Cloud Infrastructure (OCI) Autoscaling increases or decreases the number of ASAv application instances based on capacity requirements.
ASAv for AWS expanded instance support	The ASAv on the AWS Public Cloud now supports AWS Nitro System instances from different Nitro instance families. ASAv for AWS adds support for these instances: • c5a.large, c5a.xlarge, c5a.2xlarge, c5a.4xlarge • c5d.large, c5d.xlarge, c5d.2xlarge, c5d.4xlarge • c5ad.large, c5ad.xlarge, c5ad.2xlarge, c5ad.4xlarge • m5n.large, m5n.xlarge, m5n.2xlarge, m5n.4xlarge • m5zn.large, m5zn.xlarge, m5zn.2xlarge For a detailed list of supported instances, see the Cisco Adaptive Security Virtual Appliance (ASAv) Data Sheet.
ASAv for Azure expanded instance support	ASAv on the Azure Public Cloud now supports these instances: • Standard_D8s_v3 • Standard_D16s_v3 • Standard_F8s_v2 • Standard_F16s_v2 For a detailed list of supported instances, see the Cisco Adaptive Security Virtual Appliance (ASAv) Data Sheet.

Feature	Description
Intel QuickAssist Technology (QAT) on ASAv	The ASAv supports hardware crypto acceleration for ASAv deployments that use the Intel QuickAssist (QAT) 8970 PCI adapter. Hardware crypto acceleration for the ASAv using QAT is supported on VMware ESXi and KVM only.
Single Root I/O Virtualization (SR-IOV) support for ASAv on OCI.	You can now implement Single Root Input/Output Virtualization (SR-IOV) for ASAv on OCI. SR-IOV can provide performance improvements for ASAv. Mellanox 5 as vNICs are not supported in SR-IOV mode.
Firewall Features	
Twice NAT support for fully-qualified domain name (FQDN) objects as the translated (mapped) destination	You can use an FQDN network object, such as one specifying www.example.com, as the translated (mapped) destination address in twice NAT rules. The system configures the rule based on the IP address returned from the DNS server.
Network-service objects and their use in policy-based routing and access control	You can configure network-service objects and use them in extended access control lists for use in policy-based routing route maps and access control groups. Network-service objects include IP subnet or DNS domain name specifications, and optionally protocol and port specifications, that essentially combine network and service objects. This feature also includes the ability to define trusted DNS servers, to ensure that any DNS domain name resolutions acquire IP addresses from trusted sources. We added or modified the following commands: **access-list extended**, **app-id**, **clear configure object network-service**, **clear configure object-group network-service**, **clear dns ip-cache**, **clear object**, **clear object-group**, **debug network-service**, **description**, **dns trusted-source**, **domain**, **network-service-member**, **network-service reload**, **object-group network-service**, **object network-service**, **policy-route cost**, **set adaptive-interface cost**, **show asp table classify**, **show asp table network-service**, **show dns trusted-source**, **show dns ip-cache**, **show object**, **show object-group**, **show running-config**, **subnet**.
High Availability and Scalability Features	
ASAv30, ASAv50, and ASAv100 clustering for VMware and KVM	ASAv clustering lets you group up to 16 ASAvs together as a single logical device. A cluster provides all the convenience of a single device (management, integration into a network) while achieving the increased throughput and redundancy of multiple devices. ASAv clustering supports Individual Interface mode in routed firewall mode; Spanned EtherChannels are not supported. The ASAv uses a VXLAN virtual interface (VNI) for the cluster control link. New/Modified commands: **cluster-interface vni**, **nve-only cluster**, **peer-group**, **show cluster info**, **show cluster info instance-type**, **show nve 1**
Clearing routes in a high availability group or cluster	In previous releases, the **clear route** command cleared the routing table on the unit only. Now, when operating in a high availability group or cluster, the command is available on the active or control unit only, and clears the routing table on all units in the group or cluster. We changed the **clear route** command.
Interface Features	

Feature	Description
Geneve interface support for the ASAv	Geneve encapsulation support was added for the ASAv30, ASAv50, and ASAv100 to support single-arm proxy for the AWS Gateway Load Balancer. New/Modified commands: **debug geneve**, **debug nve**, **debug vxlan**, **encapsulation**, **packet-tracer geneve**, **proxy single-arm**, **show asp drop**, **show capture**, **show interface**, **show nve**
Secure Firewall 3100 auto-negotiation can be enabled or disabled for 1Gigabit and higher interfaces.	Secure Firewall 3100 auto-negotiation can be enabled or disabled for 1Gigabit and higher interfaces. For other model SFP ports, the **no speed nonegotiate** option sets the speed to 1000 Mbps; the new command means you can set auto-negotiation and speed independently. New/Modified commands: **negotiate-auto**
Administrative and Troubleshooting Features	
Startup time and tmatch compilation status	The **show version** command now includes information on how long it took to start (boot) up the system. Note that the larger the configuration, the longer it takes to boot up the system. The new **show asp rule-engine** command shows status on tmatch compilation. Tmatch compilation is used for an access list that is used as an access group, the NAT table, and some other items. It is an internal process that can consume CPU resources and impact performance while in progress, if you have very large ACLs and NAT tables. Compilation time depends on the size of the access list, NAT table, and so forth.
Enhancements to **show access-list element-count** output and **show tech-support** content	The output of the **show access-list element-count** has be enhanced to show the following: • When used in the system context in multiple-context mode, the output shows the element count for all access lists across all the contexts. • When used with object-group search enabled, the output includes details about the number of object groups in the element count. In addition, the **show tech-support** output now includes the output **show access-list element-count** and **show asp rule-engine**.

Feature	Description
CiscoSSH stack	The ASA uses a proprietary SSH stack for SSH connections. You can now choose to use the CiscoSSH stack instead, which is based on OpenSSH. The default stack continues to be the ASA stack. Cisco SSH supports: • FIPS compliance • Regular updates, including updates from Cisco and the open source community Note that the CiscoSSH stack does not support: • SSH to a different interface over VPN (management-access) • EdDSA key pair • RSA key pair in FIPS mode If you need these features, you should continue to use the ASA SSH stack. There is a small change to SCP functionality with the CiscoSSH stack: to use the ASA **copy** command to copy a file to or from an SCP server, you have to enable SSH access on the ASA for the SCP server subnet/host using the **ssh** command. New/Modified commands: **ssh stack ciscossh**
PCAP support in packet tracer	You can replay a PCAP file in packet tracer tool and obtain the trace results. **pcap** and **force** are two new keywords that is used to support the usage of PCAP in packet tracer. New/Modified commands: **packet-tracer input** and **show packet-tracer**
Stronger local user and enable password requirements	For local users and the enable password, the following password requirements were added: • Password length—Minimum 8 characters. Formerly, the minimum was 3 characters. • Repetitive and sequential characters—Three or more consecutive sequential or repetitive ASCII characters are disallowed. For example, the following passwords will be rejected: • **abc**user1 • user**543** • user**aaaa** • user2**666** New/Modified commands: **enable password**, **username**
Local user lockout changes	The ASA can lock out local users after a configurable number of failed login attempts. This feature did not apply to users with privilege level 15. Also, a user would be locked out indefinitely until an admin unlocked their account. Now, users will be unlocked after 10 minutes unless an admin uses the **clear aaa local user lockout** command before then. Privilege level 15 users are also now affected by the lockout setting. New/Modified commands: **aaa local authentication attempts max-fail** , **show aaa local user**

Feature	Description
SSH and Telnet password change prompt	The first time a local user logs into the ASA using SSH or Telnet, they are prompted to change their password. They will also be prompted for the first login after an admin changes their password. If the ASA reloads, however, users will not be prompted even if it is their first login. Note that any service that uses the local user database, such as VPN, will also have to use the new password if it was changed during an SSH or Telnet login. New/Modified commands: **show aaa local user**
Change in DNS entry TTL behavior	Formerly, the configured value was added to the existing TTL of each entry (the default was 1 minute). Now, if the expiration timer is longer than the entry's TTL, the TTL is increased to the expire entry time value. If the TTL is longer than the expiration timer, the expire entry time value is ignored; no additional time is added to the TTL in this case. New/Modified commands: **expire-entry-timer minutes**
Monitoring Features	
SNMP now supports IPv6 when grouping multiple hosts in the form of a network object	The **host-group** command of **snmp-server** now supports IPv6 host, range, and subnet objects. New/Modified commands: **snmp-server host-group**
VPN Features	
Local tunnel id support for IKEv2	Support has been added for local Tunnel id configuration for IKEv2. New/Modified commands: **set ikev2 local-identity**
Support for SAML Attributes with DAP constraint	Support has been added for SAML assertion attributes which can be used to make DAP policy selections. It also introduces the ability for a group-policy to be specified by the *cisco_group_policy* attribute.
Multiple SAML trustpoints in IDP configuration	This feature supports adding multiple IDP trustpoints per SAML IDP configuration for applications that support multiple applications for the same Entity ID. New/Modified commands: **saml idp-trustpoint <trustpoint-name>**
AnyConnect Client VPN SAML External Browser	You can now configure VPN SAML External Browser to enable additional authentication choices, such as passwordless authentication, WebAuthN, FIDO2, SSO, U2F, and an improved SAML experience due to the persistence of cookies. When you use SAML as the primary authentication method for a remote access VPN connection profile, you can elect to have the AnyConnect Client use the client's local browser instead of the AnyConnect Client embedded browser to perform the web authentication. This option enables single sign-on (SSO) between your VPN authentication and other corporate logins. Also choose this option if you want to support web authentication methods, such as biometric authentication and Yubikeys, that cannot be performed in the embedded browser. New/Modified commands: **external-browser**
VPN Load balancing with SAML	ASA now supports VPN load balancing with SAML authentication.

Firewall Functional Overview

Firewalls protect inside networks from unauthorized access by users on an outside network. A firewall can also protect inside networks from each other, for example, by keeping a human resources network separate from a user network. If you have network resources that need to be available to an outside user, such as a web or FTP server, you can place these resources on a separate network behind the firewall, called a *demilitarized zone* (DMZ). The firewall allows limited access to the DMZ, but because the DMZ only includes the public servers, an attack there only affects the servers and does not affect the other inside networks. You can also control when inside users access outside networks (for example, access to the Internet), by allowing only certain addresses out, by requiring authentication or authorization, or by coordinating with an external URL filtering server.

When discussing networks connected to a firewall, the *outside* network is in front of the firewall, the *inside* network is protected and behind the firewall, and a *DMZ*, while behind the firewall, allows limited access to outside users. Because the ASA lets you configure many interfaces with varied security policies, including many inside interfaces, many DMZs, and even many outside interfaces if desired, these terms are used in a general sense only.

Security Policy Overview

A security policy determines which traffic is allowed to pass through the firewall to access another network. By default, the ASA allows traffic to flow freely from an inside network (higher security level) to an outside network (lower security level). You can apply actions to traffic to customize the security policy.

Permitting or Denying Traffic with Access Rules

You can apply access rules to limit traffic from inside to outside, or allow traffic from outside to inside. For bridge group interfaces, you can also apply an EtherType access rule to allow non-IP traffic.

Applying NAT

Some of the benefits of NAT include the following:

- You can use private addresses on your inside networks. Private addresses are not routable on the Internet.
- NAT hides the local addresses from other networks, so attackers cannot learn the real address of a host.
- NAT can resolve IP routing problems by supporting overlapping IP addresses.

Protecting from IP Fragments

The ASA provides IP fragment protection. This feature performs full reassembly of all ICMP error messages and virtual reassembly of the remaining IP fragments that are routed through the ASA. Fragments that fail the security check are dropped and logged. Virtual reassembly cannot be disabled.

Applying HTTP, HTTPS, or FTP Filtering

Although you can use access lists to prevent outbound access to specific websites or FTP servers, configuring and managing web usage this way is not practical because of the size and dynamic nature of the Internet.

You can configure Cloud Web Security on the ASA. You can also use the ASA in conjunction with an external product such as the Cisco Web Security Appliance (WSA).

Applying Application Inspection

Inspection engines are required for services that embed IP addressing information in the user data packet or that open secondary channels on dynamically assigned ports. These protocols require the ASA to do a deep packet inspection.

Applying QoS Policies

Some network traffic, such as voice and streaming video, cannot tolerate long latency times. QoS is a network feature that lets you give priority to these types of traffic. QoS refers to the capability of a network to provide better service to selected network traffic.

Applying Connection Limits and TCP Normalization

You can limit TCP and UDP connections and embryonic connections. Limiting the number of connections and embryonic connections protects you from a DoS attack. The ASA uses the embryonic limit to trigger TCP Intercept, which protects inside systems from a DoS attack perpetrated by flooding an interface with TCP SYN packets. An embryonic connection is a connection request that has not finished the necessary handshake between source and destination.

TCP normalization is a feature consisting of advanced TCP connection settings designed to drop packets that do not appear normal.

Enabling Threat Detection

You can configure scanning threat detection and basic threat detection, and also how to use statistics to analyze threats.

Basic threat detection detects activity that might be related to an attack, such as a DoS attack, and automatically sends a system log message.

A typical scanning attack consists of a host that tests the accessibility of every IP address in a subnet (by scanning through many hosts in the subnet or sweeping through many ports in a host or subnet). The scanning threat detection feature determines when a host is performing a scan. Unlike IPS scan detection that is based on traffic signatures, the ASA scanning threat detection feature maintains an extensive database that contains host statistics that can be analyzed for scanning activity.

The host database tracks suspicious activity such as connections with no return activity, access of closed service ports, vulnerable TCP behaviors such as non-random IPID, and many more behaviors.

You can configure the ASA to send system log messages about an attacker or you can automatically shun the host.

Firewall Mode Overview

The ASA runs in two different firewall modes:

- Routed
- Transparent

In routed mode, the ASA is considered to be a router hop in the network.

In transparent mode, the ASA acts like a “bump in the wire,” or a “stealth firewall,” and is not considered a router hop. The ASA connects to the same network on its inside and outside interfaces in a "bridge group".

You might use a transparent firewall to simplify your network configuration. Transparent mode is also useful if you want the firewall to be invisible to attackers. You can also use a transparent firewall for traffic that would otherwise be blocked in routed mode. For example, a transparent firewall can allow multicast streams using an EtherType access list.

Routed mode supports Integrated Routing and Bridging, so you can also configure bridge groups in routed mode, and route between bridge groups and regular interfaces. In routed mode, you can replicate transparent mode functionality; if you do not need multiple context mode or clustering, you might consider using routed mode instead.

Stateful Inspection Overview

All traffic that goes through the ASA is inspected using the Adaptive Security Algorithm and either allowed through or dropped. A simple packet filter can check for the correct source address, destination address, and ports, but it does not check that the packet sequence or flags are correct. A filter also checks *every* packet against the filter, which can be a slow process.



Note The TCP state bypass feature allows you to customize the packet flow.

A stateful firewall like the ASA, however, takes into consideration the state of a packet:

- Is this a new connection?

  If it is a new connection, the ASA has to check the packet against access lists and perform other tasks to determine if the packet is allowed or denied. To perform this check, the first packet of the session goes through the “session management path,” and depending on the type of traffic, it might also pass through the “control plane path.”

  The session management path is responsible for the following tasks:

  - Performing the access list checks
  - Performing route lookups
  - Allocating NAT translations (xlates)
  - Establishing sessions in the “fast path”

  The ASA creates forward and reverse flows in the fast path for TCP traffic; the ASA also creates connection state information for connectionless protocols like UDP, ICMP (when you enable ICMP inspection), so that they can also use the fast path.



Note For other IP protocols, like SCTP, the ASA does not create reverse path flows. As a result, ICMP error packets that refer to these connections are dropped.

Some packets that require Layer 7 inspection (the packet payload must be inspected or altered) are passed on to the control plane path. Layer 7 inspection engines are required for protocols that have two or more

channels: a data channel, which uses well-known port numbers, and a control channel, which uses different port numbers for each session. These protocols include FTP, H.323, and SNMP.

- Is this an established connection?

  If the connection is already established, the ASA does not need to re-check packets; most matching packets can go through the "fast" path in both directions. The fast path is responsible for the following tasks:

  - IP checksum verification
  - Session lookup
  - TCP sequence number check
  - NAT translations based on existing sessions
  - Layer 3 and Layer 4 header adjustments

  Data packets for protocols that require Layer 7 inspection can also go through the fast path.

  Some established session packets must continue to go through the session management path or the control plane path. Packets that go through the session management path include HTTP packets that require inspection or content filtering. Packets that go through the control plane path include the control packets for protocols that require Layer 7 inspection.

VPN Functional Overview

A VPN is a secure connection across a TCP/IP network (such as the Internet) that appears as a private connection. This secure connection is called a tunnel. The ASA uses tunneling protocols to negotiate security parameters, create and manage tunnels, encapsulate packets, transmit or receive them through the tunnel, and unencapsulate them. The ASA functions as a bidirectional tunnel endpoint: it can receive plain packets, encapsulate them, and send them to the other end of the tunnel where they are unencapsulated and sent to their final destination. It can also receive encapsulated packets, unencapsulate them, and send them to their final destination. The ASA invokes various standard protocols to accomplish these functions.

The ASA performs the following functions:

- Establishes tunnels
- Negotiates tunnel parameters
- Authenticates users
- Assigns user addresses
- Encrypts and decrypts data
- Manages security keys
- Manages data transfer across the tunnel
- Manages data transfer inbound and outbound as a tunnel endpoint or router

The ASA invokes various standard protocols to accomplish these functions.

Security Context Overview

You can partition a single ASA into multiple virtual devices, known as security contexts. Each context is an independent device, with its own security policy, interfaces, and administrators. Multiple contexts are similar to having multiple standalone devices. Many features are supported in multiple context mode, including routing tables, firewall features, IPS, and management; however, some features are not supported. See the feature chapters for more information.

In multiple context mode, the ASA includes a configuration for each context that identifies the security policy, interfaces, and almost all the options you can configure on a standalone device. The system administrator adds and manages contexts by configuring them in the system configuration, which, like a single mode configuration, is the startup configuration. The system configuration identifies basic settings for the ASA. The system configuration does not include any network interfaces or network settings for itself; rather, when the system needs to access network resources (such as downloading the contexts from the server), it uses one of the contexts that is designated as the admin context.

The admin context is just like any other context, except that when a user logs into the admin context, then that user has system administrator rights and can access the system and all other contexts.

ASA Clustering Overview

ASA Clustering lets you group multiple ASAs together as a single logical device. A cluster provides all the convenience of a single device (management, integration into a network) while achieving the increased throughput and redundancy of multiple devices.

You perform all configuration (aside from the bootstrap configuration) on the control unit only; the configuration is then replicated to the member units.

Special and Legacy Services

For some services, documentation is located outside of the main configuration guides and online help.

Special Services Guides

Special services allow the ASA to interoperate with other Cisco products; for example, by providing a security proxy for phone services (Unified Communications), or by providing Botnet traffic filtering in conjunction with the dynamic database from the Cisco update server, or by providing WCCP services for the Cisco Web Security Appliance. Some of these special services are covered in separate guides:

- Cisco ASA Botnet Traffic Filter Guide
- Cisco ASA NetFlow Implementation Guide
- Cisco ASA Unified Communications Guide
- Cisco ASA WCCP Traffic Redirection Guide
- SNMP Version 3 Tools Implementation Guide

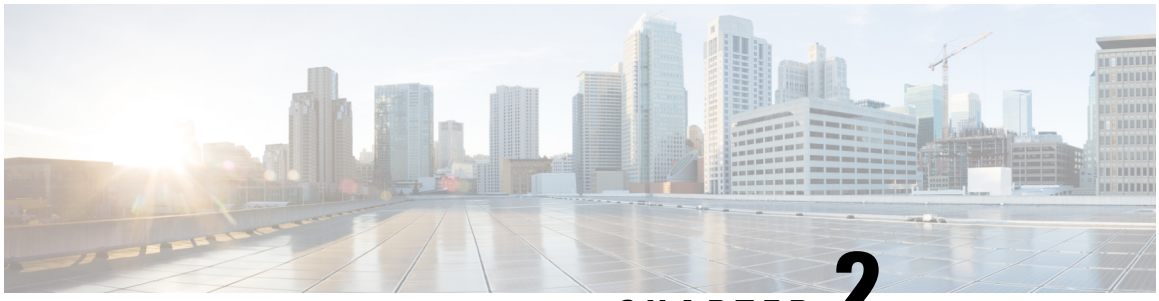
Legacy Services Guide

Legacy services are still supported on the ASA, however there may be better alternative services that you can use instead. Legacy services are covered in a separate guide:

Cisco ASA Legacy Feature Guide

This guide includes the following chapters:

- Configuring RIP
- AAA Rules for Network Access
- Using Protection Tools, which includes Preventing IP Spoofing (**ip verify reverse-path**), Configuring the Fragment Size (**fragment**), Blocking Unwanted Connections (**shun**), Configuring TCP Options (for ASDM), and Configuring IP Audit for Basic IPS Support (**ip audit**).
- Configuring Filtering Services



CHAPTER 2

Getting Started

This chapter describes how to get started with your ASA.

- Access the Console for the Command-Line Interface, on page 13
- Enable FIPS Mode, on page 19
- Configure ASDM Access, on page 21
- Start ASDM, on page 24
- Factory Default Configurations, on page 25
- Set the Firepower 2100 to Appliance or Platform Mode, on page 39
- Work with the Configuration, on page 41
- Apply Configuration Changes to Connections, on page 46
- Reload the ASA, on page 47

Access the Console for the Command-Line Interface

For initial configuration, access the CLI directly from the console port. Later, you can configure remote access using Telnet or SSH according to Management Access, on page 1161. If your system is already in multiple context mode, then accessing the console port places you in the system execution space.

If you get unreadable characters when connecting to the console port, verify the port settings. If they are correct, try the cable with another device using the same settings. If the cable is good, you might need to replace the hardware for the console port. Also consider trying a different workstation to make the connection.

Note For ASAv console access, see the ASAv quick start guide.

Access the ISA 3000 Console

Follow these steps to access the appliance console.

Procedure

Step 1 Connect a computer to the console port using the provided console cable, and connect to the console using a terminal emulator set for 9600 baud, 8 data bits, no parity, 1 stop bit, no flow control.

See the hardware guide for your ASA for more information about the console cable.

Step 2 Press the **Enter** key to see the following prompt:

```
ciscoasa>
```

This prompt indicates that you are in user EXEC mode. Only basic commands are available from user EXEC mode.

Step 3 Access privileged EXEC mode.

enable

You are prompted to change the password the first time you enter the **enable** command:

Example:

```
ciscoasa> enable
Password:
The enable password is not set. Please set it now.
Enter Password: ******
Repeat Password: ******
ciscoasa#
```

All non-configuration commands are available in privileged EXEC mode. You can also enter configuration mode from privileged EXEC mode.

To exit privileged mode, enter the **disable**, **exit**, or **quit** command.

Step 4 Access global configuration mode.

configure terminal

Example:

```
ciscoasa# configure terminal
ciscoasa(config)#
```

You can begin to configure the ASA from global configuration mode. To exit global configuration mode, enter the **exit**, **quit**, or **end** command.

Access the Firepower 2100 Platform Mode Console

The Firepower 2100 console port connects you to the Secure Firewall eXtensible Operating System (FXOS CLI). From the FXOS CLI, you can then connect to the ASA console, and back again. If you SSH to FXOS, you can also connect to the ASA CLI; a connection from SSH is not a console connection, so you can have

multiple ASA connections from an FXOS SSH connection. Similarly, if you SSH to the ASA, you can connect to the FXOS CLI.

Before you begin

You can only have one console connection at a time. When you connect to the ASA console from the FXOS console, this connection is a persistent console connection, not like a Telnet or SSH connection.

Procedure

Step 1 Connect your management computer to the console port. The Firepower 2100 ships with a DB-9 to RJ-45 serial cable, so you will need a third party serial-to-USB cable to make the connection. Be sure to install any necessary USB serial drivers for your operating system. Use the following serial settings:

- 9600 baud
- 8 data bits
- No parity
- 1 stop bit

You connect to the FXOS CLI. Enter the user credentials; by default, you can log in with the **admin** user and the default password, **Admin123**.

Step 2 Connect to the ASA:

connect asa

Example:

```
firepower-2100# connect asa
Attaching to Diagnostic CLI ... Press 'Ctrl+a then d' to detach.
Type help or '?' for a list of available commands.
ciscoasa>
```

Step 3 Access privileged EXEC mode.

enable

You are prompted to change the password the first time you enter the **enable** command.

Example:

```
ciscoasa> enable
Password:
The enable password is not set. Please set it now.
Enter Password: ******
Repeat Password: ******
ciscoasa#
```

All non-configuration commands are available in privileged EXEC mode. You can also enter configuration mode from privileged EXEC mode.

To exit privileged mode, enter the **disable**, **exit**, or **quit** command.

Step 4 Access global configuration mode.

configure terminal

Example:

```
ciscoasa# configure terminal
ciscoasa(config)#
```

You can begin to configure the ASA from global configuration mode. To exit global configuration mode, enter the **exit**, **quit**, or **end** command.

Step 5 To return to the FXOS console, enter **Ctrl+a**, **d**.

Step 6 If you SSH to the ASA (after you configure SSH access in the ASA), connect to the FXOS CLI.

connect fxos

You are prompted to authenticate for FXOS; use the default username: **admin** and password: **Admin123**. To return to the ASA CLI, enter **exit** or type **Ctrl-Shift-6**, **x**.

Example:

```
ciscoasa# connect fxos
Connecting to fxos.
Connected to fxos. Escape character sequence is 'CTRL-^X'.

FXOS 2.2(2.32) kp2110

kp2110 login: admin
Password: Admin123
Last login: Sat Jan 23 16:20:16 UTC 2017 on pts/1
Successful login attempts for user 'admin' : 4
Cisco Firepower Extensible Operating System (FX-OS) Software

[…]

kp2110#
kp2110# exit
Remote card closed command session. Press any key to continue.
Connection with fxos terminated.
Type help or '?' for a list of available commands.
ciscoasa#
```

Access the Firepower 1000, 2100 Appliance Mode, and Secure Firewall 3100 Console

The Firepower 1000, 2100 Appliance Mode, and Secure Firewall 3100 console port connects you to the ASA CLI (unlike the Firepower 2100 Platform mode console, which connects you to the FXOS CLI). From the ASA CLI, you can then connect to the FXOS CLI using Telnet for troubleshooting purposes.

Procedure

Step 1 Connect your management computer to the console port. Be sure to install any necessary serial drivers for your operating system. Use the following serial settings:

- 9600 baud
- 8 data bits
- No parity
- 1 stop bit

You connect to the ASA CLI. There are no user credentials required for console access by default.

Step 2 Access privileged EXEC mode.

enable

You are prompted to change the password the first time you enter the **enable** command.

Example:

```
ciscoasa> enable
Password:
The enable password is not set. Please set it now.
Enter Password: ******
Repeat Password: ******
ciscoasa#
```

The enable password that you set on the ASA is also the FXOS **admin** user password if the ASA fails to boot up, and you enter FXOS failsafe mode.

All non-configuration commands are available in privileged EXEC mode. You can also enter configuration mode from privileged EXEC mode.

To exit privileged EXEC mode, enter the **disable**, **exit**, or **quit** command.

Step 3 Access global configuration mode.

configure terminal

Example:

```
ciscoasa# configure terminal
ciscoasa(config)#
```

You can begin to configure the ASA from global configuration mode. To exit global configuration mode, enter the **exit**, **quit**, or **end** command.

Step 4 (Optional) Connect to the FXOS CLI.

connect fxos [**admin**]

- **admin**—Provides admin-level access. Without this option, users have read-only access. Note that no configuration commands are available even in admin mode.

You are not prompted for user credentials. The current ASA username is passed through to FXOS, and no additional login is required. To return to the ASA CLI, enter **exit** or type **Ctrl-Shift-6**, **x**.

Within FXOS, you can view user activity using the **scope security/show audit-logs** command.

Example:

```
ciscoasa# connect fxos admin
Connecting to fxos.
Connected to fxos. Escape character sequence is 'CTRL-^X'.
firepower#
firepower# exit
Connection with FXOS terminated.
Type help or '?' for a list of available commands.
ciscoasa#
```

Access the ASA Console on the Firepower 4100/9300 Chassis

For initial configuration, access the command-line interface by connecting to the Firepower 4100/9300 chassis supervisor (either to the console port or remotely using Telnet or SSH) and then connecting to the ASA security module.

Procedure

Step 1 Connect to the Firepower 4100/9300 chassis supervisor CLI (console or SSH), and then session to the ASA:

connect module *slot* {**console** | **telnet**}

The benefits of using a Telnet connection is that you can have multiple sessions to the module at the same time, and the connection speed is faster.

The first time you access the module, you access the FXOS module CLI. You must then connect to the ASA application.

connect asa

Example:

```
Firepower# connect module 1 console
Firepower-module1> connect asa

asa>
```

Step 2 Access privileged EXEC mode, which is the highest privilege level.

enable

You are prompted to change the password the first time you enter the **enable** command.

Example:

```
asa> enable
Password:
```

```
The enable password is not set. Please set it now.
Enter Password: ******
Repeat Password: ******
asa#
```

All non-configuration commands are available in privileged EXEC mode. You can also enter configuration mode from privileged EXEC mode.

To exit privileged mode, enter the **disable**, **exit**, or **quit** command.

Step 3 Enter global configuration mode.

configure terminal

Example:

```
asa# configure terminal
asa(config)#
```

To exit global configuration mode, enter the **disable**, **exit**, or **quit** command.

Step 4 Exit the application console to the FXOS module CLI by entering **Ctrl-a, d**

You might want to use the FXOS module CLI for troubleshooting purposes.

Step 5 Return to the supervisor level of the FXOS CLI.

Exit the console:

a) Enter **~**

You exit to the Telnet application.

b) To exit the Telnet application, enter:

telnet>**quit**

Exit the Telnet session:

a) Enter **Ctrl-], .**

Enable FIPS Mode

FIPS, or Federal Information Processing Standard, is a U.S. and Canadian government certification standard. It defines requirements that cryptographic modules must follow. Certain versions of ASA are FIPS 140-3 compliant, in accordance with the U.S. National Institute of Standards (NIST).

In addition to enabling FIPS on the ASA, you must also configure FIPS-compliant ciphers.

You cannot enable FIPS for clustering.

Before you begin

- FIPS mode is only supported on releases that have been through FIPS compliance. You should disable FIPS mode before you upgrade to a non-FIPS compliance version.

For information about which releases are FIPS compliant and to view their certifications, see https://www.cisco.com/c/en/us/solutions/industries/government/global-government-certifications/fips-140.html?flt0_general-table0=asa.

- For failover, both units must be in FIPS mode.

Procedure

Step 1 Enable FIPS on the ASA.

fips enable

Example:

```
ciscoasa(config)# fips enable
WARNING: FIPS mode change will not take effect until you save configuration and reboot the
 device
```

When FIPS is enabled, FIPS POST will run and print the following console message at startup:

```
Copyright (c) 1996-2005 by Cisco Systems, Inc.
Restricted Rights Legend
Use, duplication, or disclosure by the Government is subject to restrictions as set forth
in subparagraph (c) of the Commercial Computer Software - Restricted Rights clause at FAR
sec. 52.227-19 and subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer
Software clause at DFARS sec. 252.227-7013.

               Cisco Systems, Inc.
               170 West Tasman Drive
               San Jose, California 95134-1706

....
Cryptochecksum (unchanged): 6c6d2f77 ef13898e 682c9f94 9c2d5ba9

INFO: FIPS Power-On Self-Test in process.  Estimated completion in 90 seconds.
.................................................
INFO: FIPS Power-On Self-Test complete.
Type help or '?' for a list of available commands.
ciscoasa>
```

Step 2 Save your configuration and reload.

write memory

reload

Step 3 (Optional) Execute power-on self-tests.

fips self-test poweron

Tests include the cryptographic algorithm test, software integrity test, and critical functions test.

Configure ASDM Access

This section describes how to access ASDM with a default configuration and how to configure access if you do not have a default configuration.

Use the Factory Default Configuration for ASDM Access

With a factory default configuration, ASDM connectivity is pre-configured with default network settings.

Procedure

Connect to ASDM using the following interface and network settings:

- The management interface depends on your model:
  - Firepower 1010—Management 1/1 (192.168.45.1), or inside Ethernet 1/2 through 1/8 (192.168.1.1). Management hosts are limited to the 192.168.45.0/24 network, and inside hosts are limited to the 192.168.1.0/24 network.
  - Firepower 1100, 2100 in Appliance Mode, Secure Firewall 3100—Inside Ethernet 1/2 (192.168.1.1), or Management 1/1 (from DHCP). Inside hosts are limited to the 192.168.1.0/24 network. Management hosts are allowed from any network.
  - Firepower 2100 in Platform Mode—Management 1/1 (192.168.45.1). Management hosts are limited to the 192.168.45.0/24 network.
  - Firepower 4100/9300—The Management type interface and IP address of your choice defined when you deployed. Management hosts are allowed from any network.
  - ASAv—Management 0/0 (set during deployment). Management hosts are limited to the management network.
  - ISA 3000—Management 1/1 (192.168.1.1). Management hosts are limited to the 192.168.1.0/24 network.

Note
If you change to multiple context mode, you can access ASDM from the admin context using the network settings above.

Related Topics

Factory Default Configurations, on page 25
Enable or Disable Multiple Context Mode, on page 224
Start ASDM, on page 24

Customize ASDM Access

Use this procedure if *one or more* of the following conditions applies:

- You do not have a factory default configuration
- You want to change the management IP address
- You want to change to transparent firewall mode
- You want to change to multiple context mode

For routed, single mode, for quick and easy ASDM access, we recommend applying the factory default configuration with the option to set your own management IP address. Use the procedure in this section only if you have special needs such as setting transparent or multiple context mode, or if you have other configuration that you need to preserve.



Note For the ASAv, you can configure transparent mode when you deploy, so this procedure is primarily useful after you deploy if you need to clear your configuration, for example.

Procedure

Step 1 Access the CLI at the console port.

Step 2 (Optional) Enable transparent firewall mode:

This command clears your configuration.

firewall transparent

Step 3 Configure the management interface:

```
interface interface_id
   nameif name
   security-level level
   no shutdown
   ip address ip_address mask
```

Example:

```
ciscoasa(config)# interface management 0/0
ciscoasa(config-if)# nameif management
ciscoasa(config-if)# security-level 100
ciscoasa(config-if)# no shutdown
ciscoasa(config-if)# ip address 192.168.1.1 255.255.255.0
```

The **security-level** is a number between 1 and 100, where 100 is the most secure.

Step 4 (For directly-connected management hosts) Set the DHCP pool for the management network:

```
dhcpd address ip_address-ip_address interface_name
dhcpd enable interface_name
```

Example:

```
ciscoasa(config)# dhcpd address 192.168.1.2-192.168.1.254 management
ciscoasa(config)# dhcpd enable management
```

Make sure you do not include the interface address in the range.

Step 5 (For remote management hosts) Configure a route to the management hosts:

route *management_ifc management_host_ip mask gateway_ip* **1**

Example:

```
ciscoasa(config)# route management 10.1.1.0 255.255.255.0 192.168.1.50 1
```

Step 6 Enable the HTTP server for ASDM:

http server enable

Step 7 Allow the management host(s) to access ASDM:

http *ip_address mask interface_name*

Example:

```
ciscoasa(config)# http 192.168.1.0 255.255.255.0 management
```

Step 8 Save the configuration:

write memory

Step 9 (Optional) Set the mode to multiple mode:

mode multiple

When prompted, confirm that you want to convert the existing configuration to be the admin context. You are then prompted to reload the ASA.

Examples

The following configuration converts the firewall mode to transparent mode, configures the Management 0/0 interface, and enables ASDM for a management host:

```
firewall transparent
interface management 0/0

ip address 192.168.1.1 255.255.255.0
nameif management
security-level 100
no shutdown

dhcpd address 192.168.1.2-192.168.1.254 management
dhcpd enable management
http server enable
http 192.168.1.0 255.255.255.0 management
```

Related Topics

Restore the Factory Default Configuration, on page 26
Set the Firewall Mode, on page 195
Access the ISA 3000 Console, on page 13
Start ASDM, on page 24

Start ASDM

You can start ASDM using two methods:

- ASDM Launcher—The Launcher is an application downloaded from the ASA using a web browser that you can use to connect to any ASA IP address. You do not need to re-download the launcher if you want to connect to other ASAs.
- Java Web Start—For each ASA that you manage, you need to connect with a web browser and then save or launch the Java Web Start application. You can optionally save the shortcut to your computer; however you need separate shortcuts for each ASA IP address.



Note If you use web start, clear the Java cache or you might lose changes to some pre-login policies such as Hostscan. This problem does not occur if you use the launcher.

Within ASDM, you can choose a different ASA IP address to manage; the difference between the Launcher and Java Web Start functionality rests primarily in how you initially connect to the ASA and launch ASDM.

This section describes how to connect to ASDM initially, and then launch ASDM using the Launcher or the Java Web Start.

ASDM stores files in the local \Users\<user_id>\.asdm directory, including cache, log, and preferences, and also in the Temp directory, including AnyConnect Client profiles.

Procedure

Step 1 On the computer that you specified as the ASDM client, enter the following URL:

https://*asa_ip_address***/admin**

Note
Be sure to specify **https://**, and not **http://** or just the IP address (which defaults to HTTP); the ASA does not automatically forward an HTTP request to HTTPS.

The ASDM launch page appears with the following buttons:

- **Install ASDM Launcher and Run ASDM**
- **Run ASDM**
- **Run Startup Wizard**

Step 2 To download the Launcher and start ASDM:

a) Click **Install ASDM Launcher and Run ASDM**.
b) Leave the username and password fields empty (for a new installation), and click **OK**.

With no HTTPS authentication configured, you can gain access to ASDM with no username and the **enable** password, which is blank by default. When you enter the **enable** command at the CLI for the first time, you are prompted to change the password; this behavior is not enforced when you log into ASDM. We suggest that you change the enable password as soon as possible so that it does not remain blank; see Set the Hostname, Domain Name, and the Enable and Telnet Passwords, on page 735. **Note**: If you enabled HTTPS authentication, enter your username and associated password. Even without authentication, if you enter a username and password at the login screen (instead of leaving the username blank), ASDM checks the local database for a match.

c) Save the installer to your computer, and then start the installer. The ASDM Launcher opens automatically after installation is complete.
d) Enter the management IP address, the same username and password (blank for a new installation), and then click **OK**.

Step 3 To use Java Web Start:

a) Click **Run ASDM** or **Run Startup Wizard**.
b) Save the shortcut to your computer when prompted. You can optionally open it instead of saving it.
c) Start Java Web Start from the shortcut.
d) Accept any certificates according to the dialog boxes that appear. The Cisco ASDM-IDM Launcher appears.
e) Leave the username and password fields empty (for a new installation), and click **OK**. With no HTTPS authentication configured, you can gain access to ASDM with no username and the **enable** password, which is blank by default. When you enter the **enable** command at the CLI for the first time, you are prompted to change the password; this behavior is not enforced when you log into ASDM. We suggest that you change the enable password as soon as possible so that it does not remain blank; see Set the Hostname, Domain Name, and the Enable and Telnet Passwords, on page 735. **Note**: If you enabled HTTPS authentication, enter your username and associated password. Even without authentication, if you enter a username and password at the login screen (instead of leaving the username blank), ASDM checks the local database for a match.

Factory Default Configurations

The factory default configuration is the configuration applied by Cisco to new ASAs.

- Firepower 1010—The factory default configuration enables a functional inside/outside configuration. You can manage the ASA using ASDM from either the management interface or the inside switch ports.
- Firepower 1100—The factory default configuration enables a functional inside/outside configuration. You can manage the ASA using ASDM from either the management interface or the inside interface.
- Firepower 2100—Platform mode (the default):The factory default configuration enables a functional inside/outside configuration. You can manage the ASA using the Secure Firewall Chassis Manager and ASDM from the management interface.

  Appliance mode—If you change to appliance mode, the factory default configuration enables a functional inside/outside configuration. You can manage the ASA using ASDM from either the management interface or the inside interface.

- Secure Firewall 3100—The factory default configuration enables a functional inside/outside configuration. You can manage the ASA using ASDM from either the Management 1/1 interface or the inside interface.
- Firepower 4100/9300 chassis—When you deploy the standalone or cluster of ASAs, the factory default configuration configures an interface for management so that you can connect to it using ASDM, with which you can then complete your configuration.
- ASAv—Depending on your hypervisor, as part of deployment, the deployment configuration (the initial virtual deployment settings) configures an interface for management so that you can connect to it using ASDM, with which you can then complete your configuration. You can also configure failover IP addresses. You can also apply a "factory default" configuration if desired.
- ISA 3000—The factory default configuration is an almost-complete transparent firewall mode configuration with all inside and outside interfaces on the same network; you can connect to the management interface with ASDM to set the IP address of your network. Hardware bypass is enabled for two interface pairs.

For appliances, the factory default configuration is available only for routed firewall mode and single context mode, except for the ISA 3000, where the factory default configuration is only available in transparent mode. For the ASAv and the Firepower 4100/9300 chassis, you can choose transparent or routed mode at deployment.



Note In addition to the image files and the (hidden) default configuration, the following folders and files are standard in flash memory: log/, crypto_archive/, and coredumpinfo/coredump.cfg. The date on these files may not match the date of the image files in flash memory. These files aid in potential troubleshooting; they do not indicate that a failure has occurred.

Restore the Factory Default Configuration

This section describes how to restore the factory default configuration. For the ASAv, this procedure erases the deployment configuration and applies the following configuration:

```
interface management 0/0
  ip address 192.168.1.1 255.255.255.0
  nameif management
  security-level 100
  no shutdown
!
asdm logging informational
asdm history enable
!
http server enable
http 192.168.1.0 255.255.255.0 management
!
dhcpd address 192.168.1.2-192.168.1.254 management
dhcpd enable management
```



Note On the Firepower 4100/9300, restoring the factory default configuration simply erases the configuration; to restore the default configuration, you must re-deploy the ASA from the supervisor.

Before you begin

This feature is available only in routed firewall mode, except for the ISA 3000, where this command is only supported in transparent mode. In addition, this feature is available only in single context mode; an ASA with a cleared configuration does not have any defined contexts to configure automatically using this feature.

Procedure

Step 1 Restore the factory default configuration:

configure factory-default [*ip_address* [*mask*]]

Example:

```
ciscoasa(config)# configure factory-default 10.1.1.1 255.255.255.0
```

Note
This command does not clear the currently-set mode, Appliance or Platform, for the Firepower 2100.

If you specify the *ip_address*, then you set the inside or management interface IP address, depending on your model, instead of using the default IP address. See the following model guidelines for which interface is set by the *ip_address* option:

- Firepower 1010—Sets the **management** interface IP address.
- Firepower 1100—Sets the **inside** interface IP address.
- Firepower 2100 in Appliance mode—Sets the **inside** interface IP address.
- Firepower 2100 in Platform mode—Sets the **management** interface IP address.
- Secure Firewall 3100—Sets the **inside** interface IP address.
- Firepower 4100/9300—No effect.
- ASAv—Sets the **management** interface IP address.
- ISA 3000—Sets the **management** interface IP address.

The **http** command uses the subnet you specify. Similarly, the **dhcpd address** command range consists of all available addresses higher than the IP address you specify. For example, if you specify 10.5.6.78 with a subnet mask of 255.255.255.0, then the DHCP address range will be 10.5.6.79-10.5.6.254.

For the Firepower 1000, and the Firepower 2100 in Appliance mode, and the Secure Firewall 3100, 6100: This command clears the **boot system** command, if present, along with the rest of the configuration. This configuration change does not affect the image at bootup: the currently-loaded image continues to be used.

For the Firepower 2100 in Platform mode: This model does not use the **boot system** command; packages are managed by FXOS.

For all other models: This command clears the **boot system** command, if present, along with the rest of the configuration. The **boot system** command lets you boot from a specific image. The next time you reload the ASA after restoring the factory configuration, it boots from the first image in internal flash memory; if you do not have an image in internal flash memory, the ASA does not boot.

Example:

```
docs-bxb-asa3(config)# configure factory-default 10.86.203.151 255.255.254.0
Based on the management IP address and mask, the DHCP address
pool size is reduced to 103 from the platform limit 256

WARNING: The boot system configuration will be cleared.
The first image found in disk0:/ will be used to boot the
system on the next reload.
Verify there is a valid image on disk0:/ or the system will
not boot.

Begin to apply factory-default configuration:
Clear all configuration
WARNING: The new maximum-session limit will take effect after the running-config is saved
and the system boots next time. Command accepted
WARNING: Local user database is empty and there are still 'aaa' commands for 'LOCAL'.
Executing command: interface management0/0
Executing command: nameif management
INFO: Security level for "management" set to 0 by default.
Executing command: ip address 10.86.203.151 255.255.254.0
Executing command: security-level 100
Executing command: no shutdown
Executing command: exit
Executing command: http server enable
Executing command: http 10.86.202.0 255.255.254.0 management
Executing command: dhcpd address 10.86.203.152-10.86.203.254 management
Executing command: dhcpd enable management
Executing command: logging asdm informational
Factory-default configuration is completed
ciscoasa(config)#
```

Step 2 Save the default configuration to flash memory:

write memory

This command saves the running configuration to the default location for the startup configuration, even if you previously configured the **boot** config command to set a different location; when the configuration was cleared, this path was also cleared.

Restore the ASAv Deployment Configuration

This section describes how to restore the ASAv deployment (Day 0) configuration.

Procedure

Step 1 For failover, power off the standby unit.

To prevent the standby unit from becoming active, you must power it off. If you leave it on, when you erase the active unit configuration, then the standby unit becomes active. When the former active unit reloads and reconnects over the failover link, the old configuration will sync from the new active unit, wiping out the deployment configuration you wanted.

Step 2 Restore the deployment configuration after you reload. For failover, enter this command on the active unit:

write erase

Note
The ASAv boots the current running image, so you are not reverted to the original boot image. To use the original boot image, see the **boot image** command.

Do not save the configuration.

Step 3 Reload the ASAv and load the deployment configuration:

reload

Step 4 For failover, power on the standby unit.

After the active unit reloads, power on the standby unit. The deployment configuration will sync to the standby unit.

Firepower 1010 Default Configuration

The default factory configuration for the Firepower 1010 configures the following:

- **Hardware switch**—Ethernet 1/2 through 1/8 belong to VLAN 1
- **inside→outside** traffic flow—Ethernet 1/1 (outside), VLAN1 (inside)
- **management**—Management 1/1 (management), IP address 192.168.45.1
- **outside IP address** from DHCP, **inside IP address**—192.168.1.1
- **DHCP server** on inside interface, management interface
- **Default route** from outside DHCP
- **ASDM** access—Management and inside hosts allowed. Management hosts are limited to the 192.168.45.0/24 network, and inside hosts are limited to the 192.168.1.0/24 network.
- **NAT**—Interface PAT for all traffic from inside to outside.
- **DNS** servers—OpenDNS servers are pre-configured.

The configuration consists of the following commands:

```
interface Vlan1
nameif inside
security-level 100
ip address 192.168.1.1 255.255.255.0
no shutdown
!
interface Management1/1
managment-only
nameif management
no shutdown
security-level 100
ip address 192.168.45.1 255.255.255.0
!
interface Ethernet1/1
nameif outside
ip address dhcp setroute
no shutdown
```

```
!
interface Ethernet1/2
no shutdown
switchport
switchport mode access
switchport access vlan 1
!
interface Ethernet1/3
no shutdown
switchport
switchport mode access
switchport access vlan 1
!
interface Ethernet1/4
no shutdown
switchport
switchport mode access
switchport access vlan 1
!
interface Ethernet1/5
no shutdown
switchport
switchport mode access
switchport access vlan 1
!
interface Ethernet1/6
no shutdown
switchport
switchport mode access
switchport access vlan 1
!
interface Ethernet1/7
no shutdown
switchport
switchport mode access
switchport access vlan 1
!
interface Ethernet1/8
no shutdown
switchport
switchport mode access
switchport access vlan 1
!
object network obj_any
   subnet 0.0.0.0 0.0.0.0
   nat (any,outside) dynamic interface
!
dhcpd auto_config outside
dhcpd address 192.168.1.20-192.168.1.254 inside
dhcpd address 192.168.45.10-192.168.45.12 management
dhcpd enable inside
dhcpd enable management
!
http server enable
http 192.168.45.0 255.255.255.0 management
http 192.168.1.0 255.255.255.0 inside
!
dns domain-lookup outside
dns server-group DefaultDNS
   name-server 208.67.222.222 outside
   name-server 208.67.220.220 outside
!
```

Firepower 1100 Default Configuration

The default factory configuration for the Firepower 1100 configures the following:

- **inside→outside traffic flow**—Ethernet 1/1 (outside), Ethernet 1/2 (inside)
- **outside IP address** from DHCP, **inside IP address**—192.168.1.1
- **management**—Management 1/1 (management), IP address from DHCP
- **DHCP server** on inside interface
- **Default routes** from outside DHCP, management DHCP
- **ASDM** access—Management and inside hosts allowed. Inside hosts are limited to the 192.168.1.0/24 network.
- **NAT**—Interface PAT for all traffic from inside to outside.
- **DNS** servers—OpenDNS servers are pre-configured.

The configuration consists of the following commands:

```
interface Management1/1
  management-only
  nameif management
  security-level 100
  ip address dhcp setroute
  no shutdown
!
interface Ethernet1/1
  nameif outside
  security-level 0
  ip address dhcp setroute
  no shutdown
!
interface Ethernet1/2
  nameif inside
  security-level 100
  ip address 192.168.1.1 255.255.255.0
  no shutdown
!
object network obj_any
  subnet 0.0.0.0 0.0.0.0
  nat (any,outside) dynamic interface
!
http server enable
http 0.0.0.0 0.0.0.0 management
http 192.168.1.0 255.255.255.0 inside
!
dhcpd auto_config outside
dhcpd address 192.168.1.20-192.168.1.254 inside
dhcpd enable inside
!
dns domain-lookup outside
dns server-group DefaultDNS
   name-server 208.67.222.222 outside
   name-server 208.67.220.220 outside
!
```

Firepower 2100 Platform Mode Default Configuration

You can set the Firepower 2100 to run in Platform mode; Appliance mode is the default.

Note For pre-9.13(1) versions, Platform mode was the default and only option. If you upgrade from Platform mode, this mode is maintained.

ASA Configuration

The default factory configuration for the ASA on the Firepower 2100 configures the following:

- **inside→outside traffic flow**—Ethernet 1/1 (outside), Ethernet 1/2 (inside)
- **outside IP address** from DHCP, inside IP address—192.168.1.1
- **DHCP server** on inside interface
- **Default route** from outside DHCP
- **management**—Management 1/1 (management), IP address 192.168.45.1
- **ASDM** access—Management hosts allowed.
- **NAT**—Interface PAT for all traffic from inside to outside.
- **FXOS management** traffic initiation—The FXOS chassis can initiate management traffic on the ASA outside interface.
- **DNS** servers—OpenDNS servers are pre-configured.

The configuration consists of the following commands:

```
interface Management1/1
  management-only
  nameif management
  security-level 100
  ip address 192.168.45.1 255.255.255.0
  no shutdown
!
interface Ethernet1/1
  nameif outside
  security-level 0
  ip address dhcp setroute
  no shutdown
!
interface Ethernet1/2
  nameif inside
  security-level 100
  ip address 192.168.1.1 255.255.255.0
  no shutdown
!
object network obj_any
  subnet 0.0.0.0 0.0.0.0
  nat (any,outside) dynamic interface
!
http server enable
http 192.168.45.0 255.255.255.0 management
```

```
!
dhcpd auto_config outside
dhcpd address 192.168.1.20-192.168.1.254 inside
dhcpd enable inside
!
ip-client outside
!
dns domain-lookup outside
dns server-group DefaultDNS
   name-server 208.67.222.222 outside
   name-server 208.67.220.220 outside
```

FXOS Configuration

The default factory configuration for FXOS on the Firepower 2100 configures the following:

- **Management 1/1**—IP address 192.168.45.45
- **Default gateway**—ASA data interfaces
- **Firewall Chassis Manager and SSH access**—From the management network only.
- **Default Username**—**admin**, with the default password **Admin123**
- **DHCP server**—Client IP address range 192.168.45.10-192.168.45.12
- **NTP** server—Cisco NTP servers: 0.sourcefire.pool.ntp.org, 1.sourcefire.pool.ntp.org, 2.sourcefire.pool.ntp.org
- **DNS** Servers—OpenDNS: 208.67.222.222, 208.67.220.220
- **Ethernet 1/1 and Ethernet 1/2**—Enabled

Firepower 2100 Appliance Mode Default Configuration

The Firepower 2100 runs in Appliance mode by default.

Note For pre-9.13(1) versions, Platform mode was the default and only option. If you upgrade from Platform mode, Platform mode is maintained.

The default factory configuration for the Firepower 2100 in Appliance mode configures the following:

- **inside→outside traffic flow**—Ethernet 1/1 (outside), Ethernet 1/2 (inside)
- **outside IP address** from DHCP, **inside IP address**—192.168.1.1
- **management IP address** from DHCP—Management 1/1 (management)
- **DHCP server** on inside interface
- **Default routes** from outside DHCP, management DHCP
- **ASDM** access—Management and inside hosts allowed. Inside hosts are limited to the 192.168.1.0/24 network.

- **NAT**—Interface PAT for all traffic from inside to outside.
- **DNS** servers—OpenDNS servers are pre-configured.

The configuration consists of the following commands:

```
interface Management1/1
  management-only
  nameif management
  security-level 100
  ip address dhcp setroute
  no shutdown
!
interface Ethernet1/1
  nameif outside
  security-level 0
  ip address dhcp setroute
  no shutdown
!
interface Ethernet1/2
  nameif inside
  security-level 100
  ip address 192.168.1.1 255.255.255.0
  no shutdown
!
object network obj_any
  subnet 0.0.0.0 0.0.0.0
  nat (any,outside) dynamic interface
!
http server enable
http 0.0.0.0 0.0.0.0 management
http 192.168.1.0 255.255.255.0 management
!
dhcpd auto_config outside
dhcpd address 192.168.1.20-192.168.1.254 inside
dhcpd enable inside
!
dns domain-lookup outside
dns server-group DefaultDNS
   name-server 208.67.222.222 outside
   name-server 208.67.220.220 outside
!
```

Secure Firewall 3100 Default Configuration

The default factory configuration for the Secure Firewall 3100 configures the following:

- **inside→outside traffic flow**—Ethernet 1/1 (outside), Ethernet 1/2 (inside)
- **outside IP address** from DHCP, **inside IP address**—192.168.1.1
- **management**—Management 1/1 (management), IP address from DHCP
- **DHCP server** on inside interface
- **Default routes** from outside DHCP, management DHCP
- **ASDM** access—Management and inside hosts allowed. Inside hosts are limited to the 192.168.1.0/24 network.

- **NAT**—Interface PAT for all traffic from inside to outside.
- **DNS** servers—OpenDNS servers are pre-configured.

The configuration consists of the following commands:

```
interface Management1/1
  management-only
  nameif management
  security-level 100
  ip address dhcp setroute
  no shutdown
!
interface Ethernet1/1
  nameif outside
  security-level 0
  ip address dhcp setroute
  no shutdown
!
interface Ethernet1/2
  nameif inside
  security-level 100
  ip address 192.168.1.1 255.255.255.0
  no shutdown
!
object network obj_any
  subnet 0.0.0.0 0.0.0.0
  nat (any,outside) dynamic interface
!
http server enable
http 0.0.0.0 0.0.0.0 management
http 192.168.1.0 255.255.255.0 inside
!
dhcpd auto_config outside
dhcpd address 192.168.1.20-192.168.1.254 inside
dhcpd enable inside
!
dns domain-lookup outside
dns server-group DefaultDNS
   name-server 208.67.222.222 outside
   name-server 208.67.220.220 outside
!
```

Firepower 4100/9300 Chassis Default Configuration

When you deploy the ASA on the Firepower 4100/9300 chassis, you can pre-set many parameters that let you connect to the Management interface using ASDM. A typical configuration includes the following settings:

- Management interface:
  - Management type interface of your choice defined on the Firepower 4100/9300 Chassis supervisor
  - Named “management”
  - IP address of your choice
  - Security level 0
  - Management-only

- Default route through the management interface
- ASDM access—All hosts allowed.

The configuration for a standalone unit consists of the following commands. For additional configuration for clustered units, see Create an ASA Cluster, on page 438.

```
interface <management_ifc>
  management-only
  ip address <ip_address> <mask>
  ipv6 address <ipv6_address>
  ipv6 enable
  nameif management
  security-level 0
  no shutdown
!
http server enable
http 0.0.0.0 0.0.0.0 management
http ::/0 management
!
route management 0.0.0.0 0.0.0.0 <gateway_ip> 1
ipv6 route management ::/0 <gateway_ipv6>
```

ISA 3000 Default Configuration

The default factory configuration for the ISA 3000 configures the following:

- **Transparent firewall mode**—A transparent firewall is a Layer 2 firewall that acts like a "bump in the wire," or a "stealth firewall," and is not seen as a router hop to connected devices.
- **1 Bridge Virtual Interface**—All member interfaces are in the same network (**IP address *not* pre-configured; you must set to match your network**): GigabitEthernet 1/1 (outside1), GigabitEthernet 1/2 (inside1), GigabitEthernet 1/3 (outside2), GigabitEthernet 1/4 (inside2)
- All **inside and outside** interfaces can communicate with each other.
- **Management 1/1** interface—192.168.1.1/24 for ASDM access.
- **DHCP** for clients on management.
- **ASDM** access—Management hosts allowed.
- **Hardware bypass** is enabled for the following interface pairs: GigabitEthernet 1/1 & 1/2; GigabitEthernet 1/3 & 1/4



Note When the ISA 3000 loses power and goes into hardware bypass mode, only the above interface pairs can communicate; inside1 and inside2, and outside1 and outside2 can no longer communicate. Any existing connections between these interfaces will be lost. When the power comes back on, there is a brief connection interruption as the ASA takes over the flows.

The configuration consists of the following commands:

```
firewall transparent

interface GigabitEthernet1/1
  bridge-group 1
  nameif outside1
  security-level 0
  no shutdown
interface GigabitEthernet1/2
  bridge-group 1
  nameif inside1
  security-level 100
  no shutdown
interface GigabitEthernet1/3
  bridge-group 1
  nameif outside2
  security-level 0
  no shutdown
interface GigabitEthernet1/4
  bridge-group 1
  nameif inside2
  security-level 100
  no shutdown
interface Management1/1
  management-only
  no shutdown
  nameif management
  security-level 100
  ip address 192.168.1.1 255.255.255.0
interface BVI1
  no ip address

access-list allowAll extended permit ip any any
access-group allowAll in interface outside1
access-group allowAll in interface outside2

same-security-traffic permit inter-interface

hardware-bypass GigabitEthernet 1/1-1/2
hardware-bypass GigabitEthernet 1/3-1/4

http server enable
http 192.168.1.0 255.255.255.0 management

dhcpd address 192.168.1.5-192.168.1.254 management
dhcpd enable management
```

ASAv Deployment Configuration

When you deploy the ASAv, you can pre-set many parameters that let you connect to the Management 0/0 interface using ASDM. A typical configuration includes the following settings:

- Routed or Transparent firewall mode
- Management 0/0 interface:
  - Named “management”
  - IP address or DHCP

    - Security level 0
- Static route for the management host IP address (if it is not on the management subnet)
- HTTP server enabled or disabled
- HTTP access for the management host IP address
- (Optional) Failover link IP addresses for GigabitEthernet 0/8, and the Management 0/0 standby IP address
- DNS server
- Smart licensing ID token
- Smart licensing Throughput Level and Standard Feature Tier
- (Optional) Smart Call Home HTTP Proxy URL and port
- (Optional) SSH management settings:
    - Client IP addresses
    - Local username and password
    - Authentication required for SSH using the LOCAL database
- (Optional) REST API enabled or disabled



Note To successfully register the ASAv with the Cisco Licensing Authority, the ASAv requires Internet access. You might need to perform additional configuration after deployment to achieve Internet access and successful license registration.

See the following sample configuration for a standalone unit:

```
interface Management0/0
  nameif management
  security-level 0
  ip address ip_address

  no shutdown
http server enable
http managemment_host_IP mask management
route management management_host_IP mask gateway_ip 1
dns server-group DefaultDNS
  name-server ip_address
call-home
  http-proxy ip_address port port
license smart
  feature tier standard
  throughput level {100M | 1G | 2G}
  license smart register idtoken id_token
aaa authentication ssh console LOCAL
username username password password
ssh source_IP_address mask management
rest-api image boot:/path
```

```
rest-api agent
```

Note The Essentials license used to be called “Standard” license.

See the following sample configuration for a primary unit in a failover pair:

```
nameif management
  security-level 0
  ip address ip_address standby standby_ip

  no shutdown
route management management_host_IP mask gateway_ip 1
http server enable
http managemment_host_IP mask management
dns server-group DefaultDNS
  name-server ip_address
call-home
  http-proxy ip_address port port
license smart
  feature tier standard
  throughput level {100M | 1G | 2G}
  license smart register idtoken id_token
aaa authentication ssh console LOCAL
username username password password
ssh source_IP_address mask management
rest-api image boot:/path
rest-api agent
failover
failover lan unit primary
failover lan interface fover gigabitethernet0/8
failover link fover gigabitethernet0/8
failover interface ip fover primary_ip mask standby standby_ip
```

Set the Firepower 2100 to Appliance or Platform Mode

The Firepower 2100 runs an underlying operating system called the FXOS. You can run the Firepower 2100 in the following modes:

- Appliance mode (the default)—Appliance mode lets you configure all settings in the ASA. Only advanced troubleshooting commands are available from the FXOS CLI.
- Platform mode—When in Platform mode, you must configure basic operating parameters and hardware interface settings in FXOS. These settings include enabling interfaces, establishing EtherChannels, NTP, image management, and more. You can use the Firewall Chassis Manager web interface or FXOS CLI. You can then configure your security policy in the ASA operating system using ASDM or the ASA CLI.

This procedure tells you how to change the mode.

Caution When you change the mode, you need to reload the system, and the configuration is cleared. The default configuration is applied upon reload. Be sure to keep a copy of the original configuration to refer to.

Note that the **clear configure all** and **configure factory-default** commands do not clear the current mode.

Before you begin

You can only change the mode at the CLI.

Procedure

Step 1 (Optional) Back up your current configuration. See Back Up and Restore Configurations or Other Files, on page 1233.

Although there are slight differences between an Appliance mode configuration and a Platform mode configuration, a copy of the old configuration can be a good starting point. For example, for Platform mode, the NTP, DNS, and EtherChannel configuration is not part of the ASA configuration, so it will not be included in your backup, but most other ASA settings are valid for both modes.

Step 2 View the current mode.

show fxos mode

Example:

```
ciscoasa(config)# show fxos mode
Mode is currently set to appliance
```

Step 3 Set the mode to Platform mode.

no fxos mode appliance

write memory

reload

After you set the mode, you need to save the configuration and reload the device. Prior to reloading, you can set the mode back to the original value without any disruption.

Caution
When you reload, the configuration is cleared. The default configuration is applied upon reload.

Example:

```
ciscoasa(config)# no fxos mode appliance
Mode set to platform mode
WARNING: This command will take effect after the running-config is saved and the system has
 been rebooted. Command accepted.
ciscoasa(config)# write memory
Building configuration...
Cryptochecksum: c0532471 648dc7c2 4f2b4175 1f162684

23736 bytes copied in 1.520 secs (23736 bytes/sec)
```

```
[OK]
ciscoasa(config)# reload
Proceed with reload? [confirm]
```

Step 4 Set the mode to Appliance mode.

fxos mode appliance

write memory

reload

After you set the mode, you need to save the configuration and reload the device. Prior to reloading, you can set the mode back to the original value without any disruption.

Caution
When you reload, the configuration is cleared. The default configuration is applied upon reload.

Example:

```
ciscoasa(config)# fxos mode appliance
Mode set to appliance mode
WARNING: This command will take effect after the running-config is saved and the system has
 been rebooted. Command accepted.
ciscoasa(config)# write memory
Building configuration...
Cryptochecksum: c0532471 648dc7c2 4f2b4175 1f162684

23736 bytes copied in 1.520 secs (23736 bytes/sec)
[OK]
ciscoasa(config)# reload
Proceed with reload? [confirm]
```

Work with the Configuration

This section describes how to work with the configuration. The information in this section applies to both single and multiple security contexts, except where noted.

About the Startup Configuration and the Running Configuration

Startup Configuration

When the ASA starts up, it loads the configuration from a text file called the startup configuration. This file resides by default as a hidden file in internal flash memory. You can, however, specify a different file for the startup configuration that resides in the visible file system. Use the following command to specify a new startup configuration:

boot config {**disk0:/** | **disk1:/**} [*path*/]*filename*

Save the new location:

write memory

For example:

```
ciscoasa (config)# boot config disk0:/startup.cfg
ciscoasa (config)# write memory
```

Working With Large Configurations

The hidden startup directory has limited space. If your configuration is very large (for example, over 16 MB), then you will not be able to save the startup configuration. In this case, you must use the **boot config** command to save the startup configuration to the visible file system. For example, if you load a large configuration into running memory and try to save it, you may see the following error message if you enter **write memory** and the configuration is too large:

```
%Error writing. nvram:/startup-config (No space left on device:)
```

In this case, be sure to resave the running configuration to a new file location before you reload the ASA. Otherwise, the ASA may not load a complete configuration.

Running Configuration

When you enter a command, the change is made only to the running configuration in memory. You must manually save the running configuration to the startup configuration for your changes to remain after a reload.

Save Configuration Changes

This section describes how to save your configuration.

Save Configuration Changes in Single Context Mode

To save the running configuration to the startup configuration, perform the following procedure.

Procedure

Save the running configuration to the startup configuration:

write memory

Note
The **copy running-config startup-config** command is equivalent to the **write memory** command.

Save Configuration Changes in Multiple Context Mode

You can save each context (and system) configuration separately, or you can save all context configurations at the same time.

Save Each Context and System Separately

Use the following procedure to save the system or context configuration.

Procedure

From within the context or the system, save the running configuration to the startup configuration:

write memory

For multiple context mode, context startup configurations can reside on external servers. In this case, the ASA saves the configuration back to the server you identified in the context URL, except for an HTTP or HTTPS URL, which do not let you save the configuration to the server.

Note
The **copy running-config startup-config** command is equivalent to the **write memory** command.

Save All Context Configurations at the Same Time

Use the following procedure to save all context configurations at the same time, as well as the system configuration.

Procedure

From the system execution space, save the running configuration to the startup configuration for all contexts and the system configuration:

write memory all [**/noconfirm**]

If you do not enter the **/noconfirm** keyword, you see the following prompt:

```
Are you sure [Y/N]:
```

After you enter **Y**, the ASA saves the system configuration and each context. Context startup configurations can reside on external servers. In this case, the ASA saves the configuration back to the server you identified in the context URL, except for an HTTP or HTTPS URL, which do not let you save the configuration to the server.

After the ASA saves each context, the following message appears:

```
‘Saving context ‘b’ ... ( 1/3 contexts saved ) ’
```

Sometimes, a context is not saved because of an error. See the following information for errors:

- For contexts that are not saved because of low memory, the following message appears:

  ```
  The context 'context a' could not be saved due to Unavailability of resources
  ```

- For contexts that are not saved because the remote destination is unreachable, the following message appears:

```
The context 'context a' could not be saved due to non-reachability of destination
```

- For contexts that are not saved because the context is locked, the following message appears:

```
Unable to save the configuration for the following contexts as these contexts are locked.
context ‘a’ , context ‘x’ , context ‘z’ .
```

  A context is only locked if another user is already saving the configuration or in the process of deleting the context.

- For contexts that are not saved because the startup configuration is read-only (for example, on an HTTP server), the following message report is printed at the end of all other messages:

```
Unable to save the configuration for the following contexts as these contexts have
read-only config-urls:
context ‘a’ , context ‘b’ , context ‘c’ .
```

- For contexts that are not saved because of bad sectors in the flash memory, the following message appears:

```
The context 'context a' could not be saved due to Unknown errors
```

Copy the Startup Configuration to the Running Configuration

Use one of the following commands to copy a new startup configuration to the running configuration:

- **copy startup-config running-config**

  Merges the startup configuration with the running configuration. A merge adds any new commands from the new configuration to the running configuration. If the configurations are the same, no changes occur. If commands conflict or if commands affect the running of the context, then the effect of the merge depends on the command. You might get errors, or you might have unexpected results.

- **reload**

  Reloads the ASA, which loads the startup configuration and discards the running configuration.

- **clear configure all** and then **copy startup-config running-config**

  Loads the startup configuration and discards the running configuration without requiring a reload.

View the Configuration

The following commands let you view the running and startup configurations:

- **show running-config**

  Views the running configuration.

- **show running-config** *command*

Views the running configuration of a specific command.

- **show startup-config**

  Views the startup configuration.

Clear and Remove Configuration Settings

To erase settings, enter one of the following commands:

- **clear configure** *configurationcommand* [*level2configurationcommand*]

  Clears all the configuration for a specified command. If you only want to clear the configuration for a specific version of the command, you can enter a value for *level2configurationcommand*.

  For example, to clear the configuration for all **aaa** commands, enter the following command:

  ```
  ciscoasa(config)# clear configure aaa
  ```

  To clear the configuration for only **aaa authentication** commands, enter the following command:

  ```
  ciscoasa(config)# clear configure aaa authentication
  ```

- **no** *configurationcommand* [*level2configurationcommand*] *qualifier*

  Disables the specific parameters or options of a command. In this case, you use the **no** command to remove the specific configuration identified by *qualifier*.

  For example, to remove a specific **access-list** command, enter enough of the command to identify it uniquely; you may have to enter the entire command:

  ```
  ciscoasa(config)# no access-list abc extended permit icmp any any object-group obj_icmp_1
  ```

- **write erase**

  Erases the startup configuration.

Note For the ASAv, this command restores the deployment configuration after a reload. To erase the configuration completely, use the **clear configure all** command.

- **clear configure all**

  Erases the running configuration.

Note In multiple context mode, if you enter **clear configure all** from the system configuration, you also remove all contexts and stop them from running. The context configuration files are not erased, and remain in their original location.

Note For the Firepower 1000, and the Firepower 2100 in Appliance mode, and the Secure Firewall 3100: This command clears the **boot system** command, if present, along with the rest of the configuration. This configuration change does not affect the image at bootup: the currently-loaded image continues to be used.

For the Firepower 2100 in Platform mode: This model does not use the **boot system** command; packages are managed by FXOS.

For all other models: This command clears the **boot system** command, if present, along with the rest of the configuration. The **boot** system command lets you boot from a specific image, including an image on the external flash memory card. The next time you reload the ASA, it boots from the first image in internal flash memory; if you do not have an image in internal flash memory, the ASA does not boot.

Note This command does not clear the currently-set mode, Appliance or Platform, for the Firepower 2100.

Create Text Configuration Files Offline

This guide describes how to use the CLI to configure the ASA; when you save commands, the changes are written to a text file. Instead of using the CLI, however, you can edit a text file directly on your computer and paste a configuration at the configuration mode command-line prompt in its entirety, or line by line. Alternatively, you can download a text file to the ASA internal flash memory. See Software and Configurations, on page 1213 for information on downloading the configuration file to the ASA.

In most cases, commands described in this guide are preceded by a CLI prompt. The prompt in the following example is "ciscoasa(config)#":

```
ciscoasa(config)# context a
```

In the text configuration file you are not prompted to enter commands, so the prompt is omitted as follows:

```
context a
```

For additional information about formatting the file, see Using the Command-Line Interface, on page 1409.

Apply Configuration Changes to Connections

When you make security policy changes to the configuration, all *new* connections use the new security policy. Existing connections continue to use the policy that was configured at the time of the connection establishment. **show** command output for old connections reflect the old configuration, and in some cases will not include data about the old connections.

For example, if you remove a QoS **service-policy** from an interface, then re-add a modified version, then the **show service-policy** command only displays QoS counters associated with new connections that match the new service policy; existing connections on the old policy no longer show in the command output.

To ensure that all connections use the new policy, you need to disconnect the current connections so that they can reconnect using the new policy.

To disconnect connections, enter the following command:

- **clear conn** [**all**] [**protocol** {**tcp** | **udp**}] [**address** *src_ip* [*-src_ip*] [**netmask** *mask*]] [**port** *src_port* [*-src_port*]] [**address** *dest_ip* [*-dest_ip*] [**netmask** *mask*]] [**port** *dest_port* [*-dest_port*]]

  This command terminates connections in any state. See the **show conn** command to view all current connections.

  With no arguments, this command clears all through-the-box connections. To also clear to-the-box connections (including your current management session), use the **all** keyword. To clear specific connections based on the source IP address, destination IP address, port, and/or protocol, you can specify the desired options.

Reload the ASA

To reload the ASA, complete the following procedure.

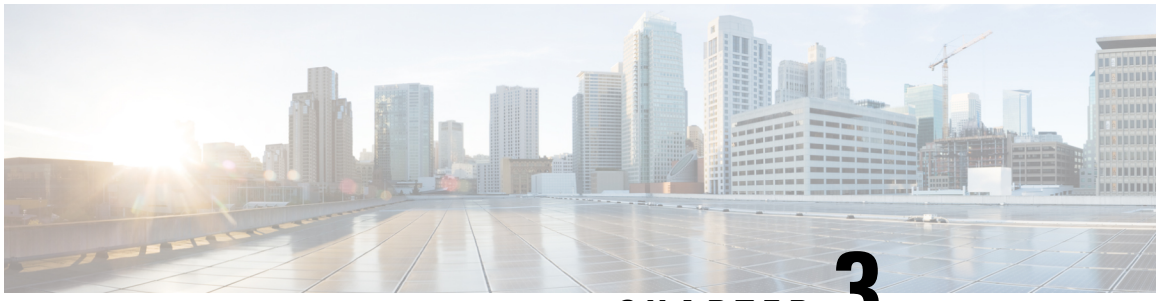
The **reload** command is not replicated to data nodes for clustering or to the standby/secondary unit for failover.

In multiple context mode, you can only reload from the system execution space.

Procedure

Reload the ASA.

reload



CHAPTER 3

Licenses: Product Authorization Key Licensing for the ISA 3000

A license specifies the options that are enabled on a given ASA. This document describes product authorization key (PAK) licenses for the ISA 3000. For other models, see Licenses: Smart Software Licensing, on page 87.

- About PAK Licenses, on page 49
- Guidelines for PAK Licenses, on page 57
- Configure PAK Licenses, on page 58
- Configure a Shared License (AnyConnect Client 3 and Earlier), on page 63
- Supported Feature Licenses Per Model, on page 70
- Monitoring PAK Licenses, on page 71
- History for PAK Licenses, on page 81

About PAK Licenses

A license specifies the options that are enabled on a given ASA. It is represented by an activation key that is a 160-bit (5 32-bit words or 20 bytes) value. This value encodes the serial number (an 11 character string) and the enabled features.

Preinstalled License

By default, your ASA ships with a license already installed. This license might be the Base License, to which you want to add more licenses, or it might already have all of your licenses installed, depending on what you ordered and what your vendor installed for you.

Related Topics

Monitoring PAK Licenses, on page 71

Permanent License

You can have one permanent activation key installed. The permanent activation key includes all licensed features in a single key. If you also install time-based licenses, the ASA combines the permanent and time-based licenses into a running license.

Related Topics

How Permanent and Time-Based Licenses Combine, on page 50

Time-Based Licenses

In addition to permanent licenses, you can purchase time-based licenses or receive an evaluation license that has a time-limit. For example, you might buy a time-based AnyConnect Client Premium license to handle short-term surges in the number of concurrent SSL VPN users.

Time-Based License Activation Guidelines

- You can install multiple time-based licenses, including multiple licenses for the same feature. However, only one time-based license per feature can be *active* at a time. The inactive license remains installed, and ready for use. For example, if you install a 1000-session AnyConnect Client Premium license, and a 2500-session AnyConnect Client Premium license, then only one of these licenses can be active.
- If you activate an evaluation license that has multiple features in the key, then you cannot also activate another time-based license for one of the included features.

How the Time-Based License Timer Works

- The timer for the time-based license starts counting down when you activate it on the ASA.
- If you stop using the time-based license before it times out, then the timer halts. The timer only starts again when you reactivate the time-based license.
- If the time-based license is active, and you shut down the ASA, then the timer stops counting down. The time-based license only counts down when the ASA is running. The system clock setting does not affect the license; only ASA uptime counts towards the license duration.

How Permanent and Time-Based Licenses Combine

When you activate a time-based license, then features from both permanent and time-based licenses combine to form the running license. How the permanent and time-based licenses combine depends on the type of license. The following table lists the combination rules for each feature license.

Note Even when the permanent license is used, if the time-based license is active, it continues to count down.

Table 1: Time-Based License Combination Rules

Time-Based Feature	Combined License Rule
AnyConnect Client Premium Sessions	The higher value is used, either time-based or permanent. For example, if the permanent license is 1000 sessions, and the time-based license is 2500 sessions, then 2500 sessions are enabled. Typically, you will not install a time-based license that has less capability than the permanent license, but if you do so, then the permanent license is used.

Time-Based Feature	Combined License Rule
Unified Communications Proxy Sessions	The time-based license sessions are added to the permanent sessions, up to the platform limit. For example, if the permanent license is 2500 sessions, and the time-based license is 1000 sessions, then 3500 sessions are enabled for as long as the time-based license is active.
All Others	The higher value is used, either time-based or permanent. For licenses that have a status of enabled or disabled, then the license with the enabled status is used. For licenses with numerical tiers, the higher value is used. Typically, you will not install a time-based license that has less capability than the permanent license, but if you do so, then the permanent license is used.

Related Topics

Monitoring PAK Licenses, on page 71

Stacking Time-Based Licenses

In many cases, you might need to renew your time-based license and have a seamless transition from the old license to the new one. For features that are only available with a time-based license, it is especially important that the license not expire before you can apply the new license. The ASA allows you to *stack* time-based licenses so that you do not have to worry about the license expiring or about losing time on your licenses because you installed the new one early.

When you install an identical time-based license as one already installed, then the licenses are combined, and the duration equals the combined duration.

For example:

1. You install an 8-week 1000-session AnyConnect Client Premium license, and use it for 2 weeks (6 weeks remain).
2. You then install another 8-week 1000-session license, and the licenses combine to be 1000-sessions for 14 weeks (8 weeks plus 6 weeks).

If the licenses are not identical (for example, a 1000-session AnyConnect Client Premium license vs. a 2500-session license), then the licenses are *not* combined. Because only one time-based license per feature can be active, only one of the licenses can be active.

Although non-identical licenses do not combine, when the current license expires, the ASA automatically activates an installed license of the same feature if available.

Related Topics

Activate or Deactivate Keys, on page 62
Time-Based License Expiration, on page 51

Time-Based License Expiration

When the current license for a feature expires, the ASA automatically activates an installed license of the same feature if available. If there are no other time-based licenses available for the feature, then the permanent license is used.

If you have more than one additional time-based license installed for a feature, then the ASA uses the first license it finds; which license is used is not user-configurable and depends on internal operations. If you prefer to use a different time-based license than the one the ASA activated, then you must manually activate the license you prefer.

For example, you have a time-based 2500-session AnyConnect Client Premium license (active), a time-based 1000-session AnyConnect Client Premium license (inactive), and a permanent 500-session AnyConnect Client Premium license. While the 2500-session license expires, the ASA activates the 1000-session license. After the 1000-session license expires, the ASA uses the 500-session permanent license.

Related Topics

Activate or Deactivate Keys, on page 62

License Notes

The following sections include additional information about licenses.

AnyConnect Plus, AnyConnect Apex, and AnyConnect VPN Only Licenses

The AnyConnect Plus or Apex license is a multi-use license that you can apply to multiple ASAs, all of which share a user pool as specified by the license. TheAnyConnect VPN Only license applies to a specific ASA. See https://www.cisco.com/go/license, and assign the PAK separately to each ASA. When you apply the resulting activation key to an ASA, it toggles on the VPN features to the maximum allowed, but the actual number of unique users across all ASAs sharing the license should not exceed the license limit. For more information, see:

- Cisco AnyConnect Client Ordering Guide
- AnyConnect Client Licensing Frequently Asked Questions (FAQ)



Note The AnyConnect Apex license is the only AnyConnect Apex license supported for multiple context mode. Moreover, in multiple context mode, this license must be applied to each unit in a failover pair; the license is not aggregated.

Other VPN License

Other VPN peers include the following VPN types:

- IPsec remote access VPN using IKEv1
- IPsec site-to-site VPN using IKEv1
- IPsec site-to-site VPN using IKEv2

This license is included in the Base license.

Total VPN Sessions Combined, All Types

- The Total VPN Peers is the maximum VPN peers allowed of both AnyConnect Client and Other VPN peers combined. For example, if the total is 1000, you can allow 500 AnyConnect Client and 500 Other VPN peers simultaneously; or 700 AnyConnect Client and 300 Other VPN; or use all 1000 for AnyConnect

Client. If you exceed the total VPN peers, you can overload the ASA, so be sure to size your network appropriately.

VPN Load Balancing

VPN load balancing requires a Strong Encryption (3DES/AES) License.

Legacy VPN Licenses

Refer to the Supplemental end User License Agreement for AnyConnect Client for all relevant information on licensing.

Note The AnyConnect Apex license is the only AnyConnect Client license supported for multiple context mode; you cannot use the default or legacy license.

Encryption License

If you have Strong Encryption enabled, you cannot use DES.

Total TLS Proxy Sessions

Each TLS proxy session for Encrypted Voice Inspection is counted against the TLS license limit.

Other applications that use TLS proxy sessions do not count toward the TLS limit, for example, Mobility Advantage Proxy (which does not require a license).

Some applications might use multiple sessions for a connection. For example, if you configure a phone with a primary and backup Cisco Unified Communications Manager, there are 2 TLS proxy connections.

You independently set the TLS proxy limit using the **tls-proxy maximum-sessions** command or in ASDM, using the **Configuration > Firewall > Unified Communications > TLS Proxy** pane. To view the limits of your model, enter the **tls-proxy maximum-sessions ?** command. When you apply a TLS proxy license that is higher than the default TLS proxy limit, the ASA automatically sets the TLS proxy limit to match the license. The TLS proxy limit takes precedence over the license limit; if you set the TLS proxy limit to be less than the license, then you cannot use all of the sessions in your license.

Note For license part numbers ending in “K8” (for example, licenses under 250 users), TLS proxy sessions are limited to 1000. For license part numbers ending in “K9” (for example, licenses 250 users or larger), the TLS proxy limit depends on the configuration, up to the model limit. K8 and K9 refer to whether the license is restricted for export: K8 is unrestricted, and K9 is restricted.

If you clear the configuration (using the **clear configure all** command, for example), then the TLS proxy limit is set to the default for your model; if this default is lower than the license limit, then you see an error message to use the **tls-proxy maximum-sessions** command to raise the limit again (in ASDM, use the **TLS Proxy** pane). If you use failover and enter the **write standby** command or in ASDM, use **File > Save Running Configuration to Standby Unit** on the primary unit to force a configuration synchronization, the **clear configure all** command is generated on the secondary unit automatically, so you may see the warning message on the secondary unit. Because the configuration synchronization restores the TLS proxy limit set on the primary unit, you can ignore the warning.

You might also use SRTP encryption sessions for your connections:

- For K8 licenses, SRTP sessions are limited to 250.
- For K9 licenses, there is no limit.

Note Only calls that require encryption/decryption for media are counted toward the SRTP limit; if passthrough is set for the call, even if both legs are SRTP, they do not count toward the limit.

VLANs, Maximum

For an interface to count against the VLAN limit, you must assign a VLAN to it. For example:

```
interface gigabitethernet 0/0.100
vlan 100
```

Shared AnyConnect Client Premium Licenses (AnyConnect 3 and Earlier)

Note The shared license feature on the ASA is not supported with AnyConnect 4 and later licensing. AnyConnect Client licenses are shared and no longer require a shared server or participant license.

A shared license lets you purchase a large number of AnyConnect Client Premium sessions and share the sessions as needed among a group of ASAs by configuring one of the ASAs as a shared licensing server, and the rest as shared licensing participants.

Failover

With some exceptions, failover units do not require the same license on each unit. For earlier versions, see the licensing document for your version.

Failover License Requirements and Exceptions

Model	License Requirement
ISA 3000	Security Plus license on both units. **Note** Each unit must have the same encryption license.

Note A valid permanent key is required; in rare instances on the ISA 3000, your PAK authentication key can be removed. If your key consists of all 0's, then you need to reinstall a valid authentication key before failover can be enabled.

How Failover Licenses Combine

For failover pairs, the licenses on each unit are combined into a single running cluster license. If you buy separate licenses for each unit, then the combined license uses the following rules:

- For licenses that have numerical tiers, such as the number of sessions, the values from each unit's licenses are combined up to the platform limit. If all licenses in use are time-based, then the licenses count down simultaneously.

  For example, for failover:

  - You have two ASAs with 10 TLS Proxy sessions installed on each; the licenses will be combined for a total of 20 TLS Proxy sessions.
  - You have an ASA with 1000 TLS Proxy sessions, and another with 2000 sessions; because the platform limit is 2000, the combined license allows 2000 TLS Proxy sessions.

- For licenses that have a status of enabled or disabled, then the license with the enabled status is used.
- For time-based licenses that are enabled or disabled (and do not have numerical tiers), the duration is the combined duration of all licenses. The primary/control unit counts down its license first, and when it expires, the secondary/data unit(s) start counting down its license, and so on.

Related Topics

Monitoring PAK Licenses, on page 71

Loss of Communication Between Failover Units

If the units lose communication for more than 30 days, then each unit reverts to the license installed locally. During the 30-day grace period, the combined running license continues to be used by all units.

If you restore communication during the 30-day grace period, then for time-based licenses, the time elapsed is subtracted from the primary/control license; if the primary/control license becomes expired, only then does the secondary/data license start to count down.

If you do not restore communication during the 30-day period, then for time-based licenses, time is subtracted from all unit licenses, if installed. They are treated as separate licenses and do not benefit from the combined license. The time elapsed includes the 30-day grace period.

Upgrading Failover Pairs

Because failover pairs do not require the same license on both units, you can apply new licenses to each unit without any downtime. If you apply a permanent license that requires a reload, then you can fail over to the other unit while you reload. If both units require reloading, then you can reload them separately so that you have no downtime.

Related Topics

Activate or Deactivate Keys, on page 62

No Payload Encryption Models

You can purchase some models with No Payload Encryption. For export to some countries, payload encryption cannot be enabled on the ASA series. The ASA software senses a No Payload Encryption model, and disables the following features:

- Unified Communications
- VPN

You can still install the Strong Encryption (3DES/AES) license for use with management connections. For example, you can use ASDM HTTPS/SSL, SSHv2, Telnet and SNMPv3.

When you view the license, VPN and Unified Communications licenses will not be listed.

Related Topics

Monitoring PAK Licenses, on page 71

Licenses FAQ

Can I activate multiple time-based licenses?

Yes. You can use one time-based license per feature at a time.

Can I "stack" time-based licenses so that when the time limit runs out, it will automatically use the next license?

Yes. For identical licenses, the time limit is combined when you install multiple time-based licenses. For non-identical licenses (for example, a 1000-session AnyConnect Client Premium license and a 2500-session license), the ASA automatically activates the next time-based license it finds for the feature.

Can I install a new permanent license while maintaining an active time-based license?

Yes. Activating a permanent license does not affect time-based licenses.

For failover, can I use a shared licensing server as the primary unit, and the shared licensing backup server as the secondary unit?

No. The secondary unit has the same running license as the primary unit; in the case of the shared licensing server, they require a server license. The backup server requires a participant license. The backup server can be in a separate failover pair of two backup servers.

Do I need to buy the same licenses for the secondary unit in a failover pair?

No. Starting with Version 8.3(1), you do not have to have matching licenses on both units. Typically, you buy a license only for the primary unit; the secondary unit inherits the primary license when it becomes active. In the case where you also have a separate license on the secondary unit (for example, if you purchased matching licenses for pre-8.3 software), the licenses are combined into a running failover cluster license, up to the model limits.

Can I use a time-based or permanent AnyConnect Client Premium license in addition to a shared AnyConnect Premium license?

Yes. The shared license is used only after the sessions from the locally installed license (time-based or permanent) are used up.



Note On the shared licensing server, the permanent AnyConnect Client Premium license is not used; you can however use a time-based license at the same time as the shared licensing server license. In this case, the time-based license sessions are available for local AnyConnect Client Premium sessions only; they cannot be added to the shared licensing pool for use by participants.

Guidelines for PAK Licenses

Context Mode Guidelines

In multiple context mode, apply the activation key in the system execution space.

Failover Guidelines

See Failover, on page 54.

Model Guidelines

- Smart Licensing is supported on the ASAv only.
- Shared licenses are not supported on the ASAv, ASA 5506-X, ASA 5508-X, and ASA 5516-X.
- The ASA 5506-X and ASA 5506W-X do not support time-based licenses.

Upgrade and Downgrade Guidelines

Your activation key remains compatible if you upgrade to the latest version from any previous version. However, you might have issues if you want to maintain downgrade capability:

- Downgrading to Version 8.1 or earlier—After you upgrade, if you activate additional feature licenses that were introduced *before 8.2*, then the activation key continues to be compatible with earlier versions if you downgrade. However if you activate feature licenses that were introduced in *8.2 or later*, then the activation key is not backwards compatible. If you have an incompatible license key, then see the following guidelines:
  - If you previously entered an activation key in an earlier version, then the ASA uses that key (without any of the new licenses you activated in Version 8.2 or later).
  - If you have a new system and do not have an earlier activation key, then you need to request a new activation key compatible with the earlier version.
- Downgrading to Version 8.2 or earlier—Version 8.3 introduced more robust time-based key usage as well as failover license changes:
  - If you have more than one time-based activation key active, when you downgrade, only the most recently activated time-based key can be active. Any other keys are made inactive. If the last time-based license is for a feature introduced in 8.3, then that license still remains the active license even though it cannot be used in earlier versions. Reenter the permanent key or a valid time-based key.
  - If you have mismatched licenses on a failover pair, then downgrading will disable failover. Even if the keys are matching, the license used will no longer be a combined license.
  - If you have one time-based license installed, but it is for a feature introduced in 8.3, then after you downgrade, that time-based license remains active. You need to reenter the permanent key to disable the time-based license.

Additional Guidelines

- The activation key is not stored in your configuration file; it is stored as a hidden file in flash memory.
- The activation key is tied to the serial number of the device. Feature licenses cannot be transferred between devices (except in the case of a hardware failure). If you have to replace your device due to a hardware failure, and it is covered by Cisco TAC, contact the Cisco Licensing Team to have your existing license transferred to the new serial number. The Cisco Licensing Team will ask for the Product Authorization Key reference number and existing serial number.
- The serial number used for licensing is the one seen in the **show version** output. This serial number is different from the chassis serial number printed on the outside of your hardware. The chassis serial number is used for technical support, but not for licensing.
- Once purchased, you cannot return a license for a refund or for an upgraded license.
- On a single unit, you cannot add two separate licenses for the same feature together; for example, if you purchase a 25-session SSL VPN license, and later purchase a 50-session license, you cannot use 75 sessions; you can use a maximum of 50 sessions. (You may be able to purchase a larger license at an upgrade price, for example from 25 sessions to 75 sessions; this kind of upgrade should be distinguished from adding two separate licenses together).
- Although you can activate all license types, some features are incompatible with each other. In the case of the AnyConnect Essentials license, the license is incompatible with the following licenses: AnyConnect Premium license, shared AnyConnect Premium license, and Advanced Endpoint Assessment license. By default, if you install the AnyConnect Essentials license (if it is available for your model), it is used instead of the above licenses. You can disable the AnyConnect Essentials license in the configuration to restore use of the other licenses using the **webvpn**, and then the **no anyconnect-essentials** command.

Configure PAK Licenses

This section describes how to obtain an activation key and how to active it. You can also deactivate a key.

Order License PAKs and Obtain an Activation Key

To install a license on the ASA, you need Product Authorization Keys, which you can then register with Cisco.com to obtain an activation key. You can then enter the activation key on the ASA. You need a separate Product Authorization Key for each feature license. The PAKs are combined to give you a single activation key. You may have received all of your license PAKs in the box with your device. The ASA has the Base or Security Plus license pre-installed, along with the Strong Encryption (3DES/AES) license if you qualify for its use. If you need to manually request the Strong Encryption license (which is free), see http://www.cisco.com/go/license.

Before you begin

When you purchase 1 or more licenses for the device, you manage them in the Cisco Smart Software Manager:

https://software.cisco.com/#module/SmartLicensing

If you do not yet have an account, set up a new account. The Smart Software Manager lets you create a master account for your organization.

Procedure

Step 1 To purchase additional licenses, see http://www.cisco.com/go/ccw. See the following AnyConnect Client ordering guide and FAQ:

- Cisco AnyConnect Client Ordering Guide
- AnyConnect Client Licensing Frequently Asked Questions (FAQ)

After you order a license, you will then receive an email with a Product Authorization Key (PAK). For the AnyConnect Client licenses, you receive a multi-use PAK that you can apply to multiple ASAs that use the same pool of user sessions. The PAK email can take several days in some cases.

Step 2 Obtain the serial number for your ASA by entering the following command.

show version | grep Serial

The serial number used for licensing is different from the chassis serial number printed on the outside of your hardware. The chassis serial number is used for technical support, but not for licensing.

Step 3 To obtain the activation key, go to the following licensing website:

http://www.cisco.com/go/license

Step 4 Enter the following information, when prompted:

- Product Authorization Key (if you have multiple keys, enter one of the keys first. You have to enter each key as a separate process.)
- The serial number of your ASA
- Your e-mail address

An activation key is automatically generated and sent to the e-mail address that you provide. This key includes all features you have registered so far for permanent licenses. For time-based licenses, each license has a separate activation key.

Step 5 If you have additional Product Authorization Keys, repeat the process for each Product Authorization Key. After you enter all of the Product Authorization Keys, the final activation key provided includes all of the permanent features you registered.

Step 6 Install the activation key according to Activate or Deactivate Keys, on page 62.

Obtain a Strong Encryption License

To use ASDM (and many other features), you need to install the Strong Encryption (3DES/AES) license. If your ASA did not come with the Strong Encryption license pre-installed, you can request one for free. You must qualify for a Strong Encryption license based on your country.

Procedure

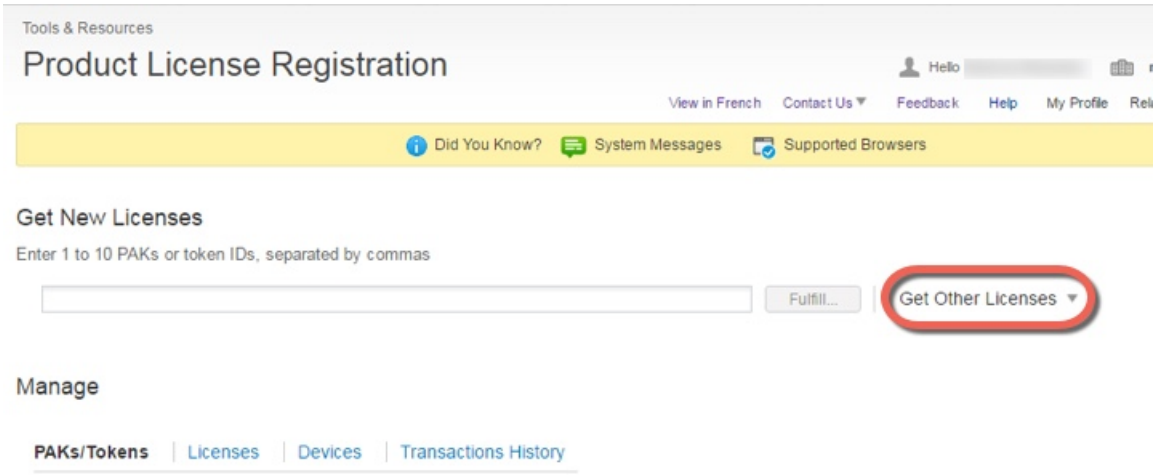
Step 1 Obtain the serial number for your ASA by entering the following command:

show version | grep Serial

This serial number is different from the chassis serial number printed on the outside of your hardware. The chassis serial number is used for technical support, but not for licensing.

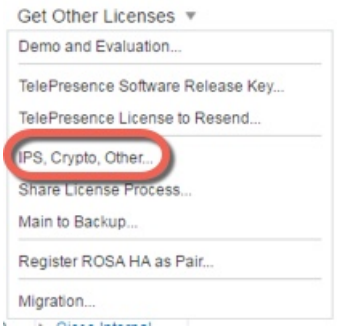
Step 2 See https://www.cisco.com/go/license, and click **Get Other Licenses**.

Figure 1: Get Other Licenses



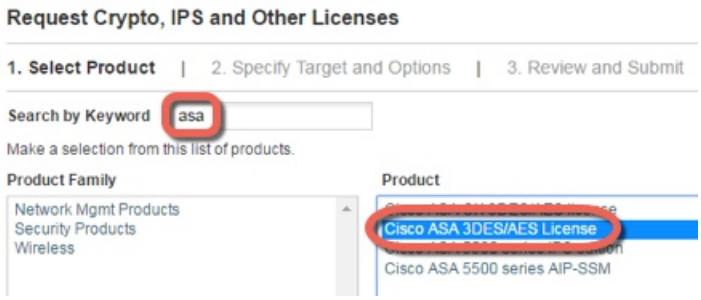
Step 3 Choose **IPS, Crypto, Other**.

Figure 2: IPS, Crypto, Other



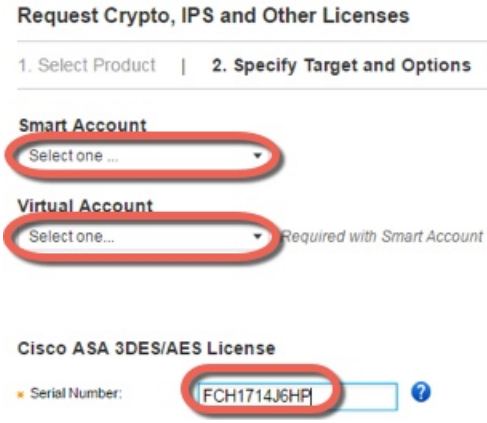
Step 4 In the **Search by Keyword** field, enter **asa**, and select **Cisco ASA 3DES/AES License**.

Figure 3: Cisco ASA 3DES/AES License



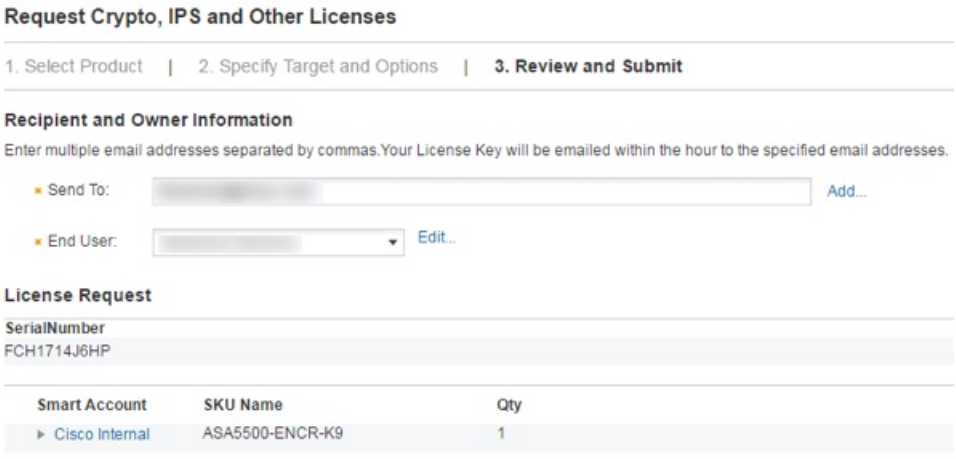
Step 5 Select your **Smart Account**, **Virtual Account**, enter the ASA **Serial Number**, and click **Next**.

Figure 4: Smart Account, Virtual Account, and Serial Number



Step 6 Your Send To email address and End User name are auto-filled; enter additional email addresses if needed. Check the **I Agree** check box, and click **Submit**.

Figure 5: Submit



Step 7 You will then receive an email with the activation key, but you can also download the key right away from the **Manage** > **Licenses** area.

Step 8 Apply the activation key according to Activate or Deactivate Keys, on page 62.

Activate or Deactivate Keys

This section describes how to enter a new activation key, and how to activate and deactivate time-based keys.

Before you begin

- If you are already in multiple context mode, enter the activation key in the system execution space.
- Some permanent licenses require you to reload the ASA after you activate them. The following table lists the licenses that require reloading.

Table 2: Permanent License Reloading Requirements

Model	License Action Requiring Reload
All models	Downgrading the Encryption license.

Procedure

Step 1 Apply an activation key to the ASA:

activation-key *key* [**activate** | **deactivate**]

Example:

```
ciscoasa# activation-key 0xd11b3d48 0xa80a4c0a 0x48e0fd1c 0xb0443480 0x843fc490
```

The *key* is a five-element hexadecimal string with one space between each element. The leading 0x specifier is optional; all values are assumed to be hexadecimal.

You can install one permanent key, and multiple time-based keys. If you enter a new permanent key, it overwrites the already installed one.

The **activate** and **deactivate** keywords are available for time-based keys only. If you do not enter any value, **activate** is the default. The last time-based key that you activate for a given feature is the active one. To deactivate any active time-based key, enter the **deactivate** keyword. If you enter a key for the first time, and specify **deactivate**, then the key is installed on the ASA in an inactive state.

Step 2 (Might be required.) Reload the ASA:

reload

Some permanent licenses require you to reload the ASA after entering the new activation key. If you need to reload, you will see the following message:

```
WARNING: The running activation key was not updated with the requested key.
The flash activation key was updated with the requested key, and will become
```

```
active after the next reload.
```

Related Topics

Time-Based Licenses, on page 50

Configure a Shared License (AnyConnect Client 3 and Earlier)

Note The shared license feature on the ASA is not supported with AnyConnect Client 4 and later licensing. AnyConnect Client licenses are shared and no longer require a shared server or participant license.

This section describes how to configure the shared licensing server and participants.

About Shared Licenses

A shared license lets you purchase a large number of AnyConnect Client Premium sessions and share the sessions as needed among a group of ASAs by configuring one of the ASAs as a shared licensing server, and the rest as shared licensing participants.

About the Shared Licensing Server and Participants

The following steps describe how shared licenses operate:

1. Decide which ASA should be the shared licensing server, and purchase the shared licensing server license using that device serial number.
2. Decide which ASAs should be shared licensing participants, including the shared licensing backup server, and obtain a shared licensing participant license for each device, using each device serial number.
3. (Optional) Designate a second ASA as a shared licensing backup server. You can only specify one backup server.

Note The shared licensing backup server only needs a participant license.

4. Configure a shared secret on the shared licensing server; any participants with the shared secret can use the shared license.
5. When you configure the ASA as a participant, it registers with the shared licensing server by sending information about itself, including the local license and model information.

Note The participant needs to be able to communicate with the server over the IP network; it does not have to be on the same subnet.

6. The shared licensing server responds with information about how often the participant should poll the server.
7. When a participant uses up the sessions of the local license, it sends a request to the shared licensing server for additional sessions in 50-session increments.
8. The shared licensing server responds with a shared license. The total sessions used by a participant cannot exceed the maximum sessions for the platform model.



Note The shared licensing server can also participate in the shared license pool. It does not need a participant license as well as the server license to participate.

   a. If there are not enough sessions left in the shared license pool for the participant, then the server responds with as many sessions as available.
   b. The participant continues to send refresh messages requesting more sessions until the server can adequately fulfill the request.

9. When the load is reduced on a participant, it sends a message to the server to release the shared sessions.



Note The ASA uses SSL between the server and participant to encrypt all communications.

Communication Issues Between Participant and Server

See the following guidelines for communication issues between the participant and server:

- If a participant fails to send a refresh after 3 times the refresh interval, then the server releases the sessions back into the shared license pool.
- If the participant cannot reach the license server to send the refresh, then the participant can continue to use the shared license it received from the server for up to 24 hours.
- If the participant is still not able to communicate with a license server after 24 hours, then the participant releases the shared license, even if it still needs the sessions. The participant leaves existing connections established, but cannot accept new connections beyond the license limit.
- If a participant reconnects with the server before 24 hours expires, but after the server expired the participant sessions, then the participant needs to send a new request for the sessions; the server responds with as many sessions as can be reassigned to that participant.

About the Shared Licensing Backup Server

The shared licensing backup server must register successfully with the main shared licensing server before it can take on the backup role. When it registers, the main shared licensing server syncs server settings as well as the shared license information with the backup, including a list of registered participants and the current license usage. The main server and backup server sync the data at 10 second intervals. After the initial sync, the backup server can successfully perform backup duties, even after a reload.

When the main server goes down, the backup server takes over server operation. The backup server can operate for up to 30 continuous days, after which the backup server stops issuing sessions to participants, and existing sessions time out. Be sure to reinstate the main server within that 30-day period. Critical-level syslog messages are sent at 15 days, and again at 30 days.

When the main server comes back up, it syncs with the backup server, and then takes over server operation.

When the backup server is not active, it acts as a regular participant of the main shared licensing server.

Note When you first launch the main shared licensing server, the backup server can only operate independently for 5 days. The operational limit increases day-by-day, until 30 days is reached. Also, if the main server later goes down for any length of time, the backup server operational limit decrements day-by-day. When the main server comes back up, the backup server starts to increment again day-by-day. For example, if the main server is down for 20 days, with the backup server active during that time, then the backup server will only have a 10-day limit left over. The backup server "recharges" up to the maximum 30 days after 20 more days as an inactive backup. This recharging function is implemented to discourage misuse of the shared license.

Failover and Shared Licenses

This section describes how shared licenses interact with failover.

Failover and Shared License Servers

This section describes how the main server and backup server interact with failover. Because the shared licensing server is also performing normal duties as the ASA, including performing functions such as being a VPN gateway and firewall, then you might need to configure failover for the main and backup shared licensing servers for increased reliability.

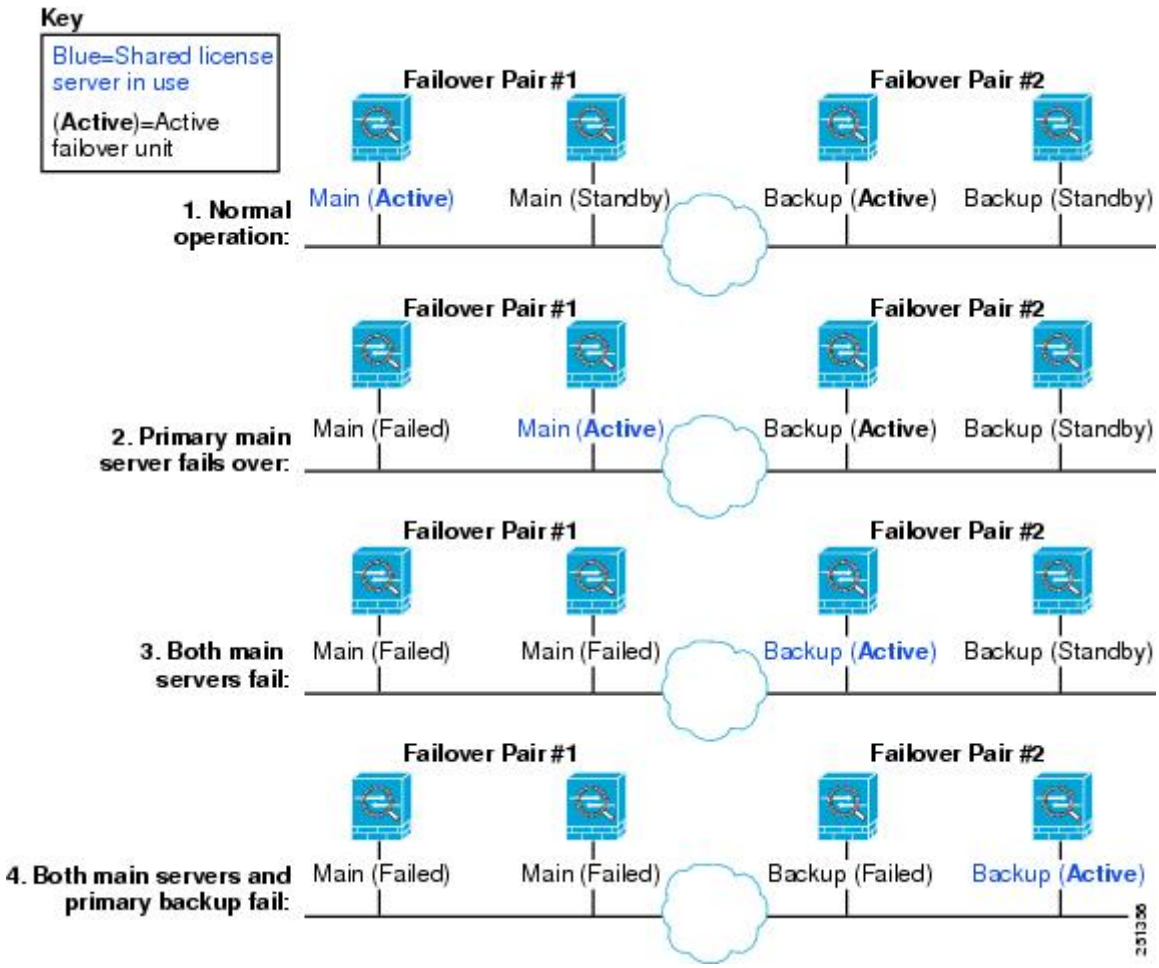
Note The backup server mechanism is separate from, but compatible with, failover.

Shared licenses are supported only in single context mode, so Active/Active failover is not supported.

For Active/Standby failover, the primary unit acts as the main shared licensing server, and the standby unit acts as the main shared licensing server after failover. The standby unit does *not* act as the backup shared licensing server. Instead, you can have a second pair of units acting as the backup server, if desired.

For example, you have a network with 2 failover pairs. Pair #1 includes the main licensing server. Pair #2 includes the backup server. When the primary unit from Pair #1 goes down, the standby unit immediately becomes the new main licensing server. The backup server from Pair #2 never gets used. Only if both units in Pair #1 go down does the backup server in Pair #2 come into use as the shared licensing server. If Pair #1 remains down, and the primary unit in Pair #2 goes down, then the standby unit in Pair #2 comes into use as the shared licensing server (see the following figure).

Figure 6: Failover and Shared License Servers



The standby backup server shares the same operating limits as the primary backup server; if the standby unit becomes active, it continues counting down where the primary unit left off.

Related Topics

About the Shared Licensing Backup Server, on page 64

Failover and Shared License Participants

For participant pairs, both units register with the shared licensing server using separate participant IDs. The active unit syncs its participant ID with the standby unit. The standby unit uses this ID to generate a transfer request when it switches to the active role. This transfer request is used to move the shared sessions from the previously active unit to the new active unit.

Maximum Number of Participants

The ASA does not limit the number of participants for the shared license; however, a very large shared network could potentially affect the performance on the licensing server. In this case, you can increase the delay between participant refreshes, or you can create two shared networks.

Configure the Shared Licensing Server

This section describes how to configure the ASA to be a shared licensing server.

Before you begin

The server must have a shared licensing server key.

Procedure

Step 1 Set the shared secret:

license-server secret *secret*

Example:

```
ciscoasa(config)# license-server secret farscape
```

The *secret* is a string between 4 and 128 ASCII characters. Any participant with this secret can use the licensing server.

Step 2 (Optional) Set the refresh interval:

license-server refresh-interval *seconds*

Example:

```
ciscoasa(config)# license-server refresh-interval 100
```

The interval is between 10 and 300 seconds; this value is provided to participants to set how often they should communicate with the server. The default is 30 seconds.

Step 3 (Optional) Set the port on which the server listens for SSL connections from participants:

license-server port *port*

Example:

```
ciscoasa(config)# license-server port 40000
```

The *port* is between 1 and 65535. The default is TCP port 50554.

Step 4 (Optional) Identify the backup server IP address and serial number:

license-server backup *address* **backup-id** *serial_number* [**ha-backup-id** *ha_serial_number*]

Example:

```
ciscoasa(config)# license-server backup 10.1.1.2 backup-id JMX0916L0Z4 ha-backup-id
JMX1378N0W3
```

If the backup server is part of a failover pair, identify the standby unit serial number as well. You can only identify 1 backup server and its optional standby unit.

Step 5 Enable this unit to be the shared licensing server:

license-server enable *interface_name*

Example:

```
ciscoasa(config)# license-server enable inside
```

Specify the interface on which participants contact the server. You can repeat this command for as many interfaces as desired.

Examples

The following example sets the shared secret, changes the refresh interval and port, configures a backup server, and enables this unit as the shared licensing server on the inside interface and dmz interface:

```
ciscoasa(config)# license-server secret farscape
ciscoasa(config)# license-server refresh-interval 100
ciscoasa(config)# license-server port 40000
ciscoasa(config)# license-server backup 10.1.1.2 backup-id JMX0916L0Z4 ha-backup-id
JMX1378N0W3
ciscoasa(config)# license-server enable inside
ciscoasa(config)# license-server enable dmz
```

Configure the Shared Licensing Backup Server (Optional)

This section enables a shared license participant to act as the backup server if the main server goes down.

Before you begin

The backup server must have a shared licensing participant key.

Procedure

Step 1 Identify the shared licensing server IP address and shared secret:

license-server address *address* **secret** *secret* [**port** *port*]

Example:

```
ciscoasa(config)# license-server address 10.1.1.1 secret farscape
```

If you changed the default port in the server configuration, set the port for the backup server to match.

Step 2 Enable this unit to be the shared licensing backup server:

license-server backup enable *interface_name*

Example:

```
ciscoasa(config)# license-server backup enable inside
```

Specify the interface on which participants contact the server. You can repeat this command for as many interfaces as desired.

Examples

The following example identifies the license server and shared secret, and enables this unit as the backup shared license server on the inside interface and dmz interface:

```
ciscoasa(config)# license-server address 10.1.1.1 secret farscape
ciscoasa(config)# license-server backup enable inside
ciscoasa(config)# license-server backup enable dmz
```

Configure the Shared Licensing Participant

This section configures a shared licensing participant to communicate with the shared licensing server.

Before you begin

The participant must have a shared licensing participant key.

Procedure

Step 1 Identify the shared licensing server IP address and shared secret:

license-server address *address* **secret** *secret* [**port** *port*]

Example:

```
ciscoasa(config)# license-server address 10.1.1.1 secret farscape
```

If you changed the default port in the server configuration, set the port for the participant to match.

Step 2 (Optional) If you configured a backup server, enter the backup server address:

license-server backup address *address*

Example:

```
ciscoasa(config)# license-server backup address 10.1.1.2
```

Examples

The following example sets the license server IP address and shared secret, as well as the backup license server IP address:

```
ciscoasa(config)# license-server address 10.1.1.1 secret farscape
ciscoasa(config)# license-server backup address 10.1.1.2
```

Supported Feature Licenses Per Model

This section describes the licenses available for each model as well as important notes about licenses.

Licenses Per Model

This section lists the feature licenses available for each model:

Items that are in *italics* are separate, optional licenses that can replace the Base (or Security Plus, and so on) license version. You can mix and match optional licenses.

Note Some features are incompatible with each other. See the individual feature chapters for compatibility information.

If you have a No Payload Encryption model, then some of the features below are not supported. See No Payload Encryption Models, on page 55 for a list of unsupported features.

For detailed information about licenses, see License Notes, on page 52.

ISA 3000 License Features

The following table shows the licensed features for the ISA 3000.

Licenses	Base License	Security Plus License
Firewall Licenses		
Botnet Traffic Filter	No support	No Support
Firewall Conns, Concurrent	20,000	50,000
Carrier	No Support	No Support
Total TLS Proxy Sessions	160	160
VPN Licenses		

Licenses	Base License		Security Plus License	
AnyConnect Client peers	Disabled	*Optional AnyConnect Plus, AnyConnect Apex, or AnyConnect VPN Only license: 25 maximum*	Disabled	*Optional AnyConnect Plus, AnyConnect Apex, or AnyConnect VPN Only license: 25 maximum*
Other VPN Peers	10		50	
Total VPN Peers, combined all types	25		50	
VPN Load Balancing	No support		No support	
General Licenses				
Encryption	Base (DES)	*Opt. lic.: Strong (3DES/AES)*	Base (DES)	*Opt. lic.: Strong (3DES/AES)*
Failover	No support		Active/Standby	
Security Contexts	No support		No Support	
Clustering	No Support		No Support	
VLANs, Maximum	5		25	

Monitoring PAK Licenses

This section describes how to view license information.

Viewing Your Current License

This section describes how to view your current license, and for time-based activation keys, how much time the license has left.

Before you begin

If you have a No Payload Encryption model, then you view the license, VPN and Unified Communications licenses will not be listed. See No Payload Encryption Models, on page 55 for more information.

Procedure

Show the permanent license, active time-based licenses, and the running license, which is a combination of the permanent license and active time-based licenses:

show activation-key [**detail**]

The **detail** keyword also shows inactive time-based licenses.

For failover or cluster units, this command also shows the “cluster” license, which is the combined keys of all units.

Examples

Example 1: Standalone Unit Output for the show activation-key command

The following is sample output from the **show activation-key** command for a standalone unit that shows the running license (the combined permanent license and time-based licenses), as well as each active time-based license:

```
ciscoasa# show activation-key

Serial Number:  JMX1232L11M
Running Permanent Activation Key: 0xce06dc6b 0x8a7b5ab7 0xa1e21dd4 0xd2c4b8b8 0xc4594f9c
Running Timebased Activation Key: 0xa821d549 0x35725fe4 0xc918b97b 0xce0b987b 0x47c7c285
Running Timebased Activation Key: 0xyadayad2 0xyadayad2 0xyadayad2 0xyadayad2 0xyadayad2

Licensed features for this platform:
Maximum Physical Interfaces       : Unlimited      perpetual
Maximum VLANs                     : 150            perpetual
Inside Hosts                      : Unlimited      perpetual
Failover                          : Active/Active  perpetual
VPN-DES                           : Enabled        perpetual
VPN-3DES-AES                      : Enabled        perpetual
Security Contexts                 : 10             perpetual
GTP/GPRS                          : Enabled        perpetual
AnyConnect Premium Peers          : 2              perpetual
AnyConnect Essentials             : Disabled       perpetual
Other VPN Peers                   : 750            perpetual
Total VPN Peers                   : 750            perpetual
Shared License                    : Enabled        perpetual
  Shared AnyConnect Premium Peers : 12000          perpetual
AnyConnect for Mobile             : Disabled       perpetual
AnyConnect for Cisco VPN Phone    : Disabled       perpetual
Advanced Endpoint Assessment      : Disabled       perpetual
UC Phone Proxy Sessions           : 12             62 days
Total UC Proxy Sessions           : 12             62 days
Botnet Traffic Filter             : Enabled        646 days
Intercompany Media Engine         : Disabled       perpetual

This platform has a Base license.

The flash permanent activation key is the SAME as the running permanent key.

Active Timebased Activation Key:
0xa821d549 0x35725fe4 0xc918b97b 0xce0b987b 0x47c7c285
Botnet Traffic Filter        : Enabled    646 days

0xyadayad2 0xyadayad2 0xyadayad2 0xyadayad2 0xyadayad2
Total UC Proxy Sessions      : 10         62 days
```

Example 2: Standalone Unit Output for show activation-key detail

The following is sample output from the **show activation-key detail** command for a standalone unit that shows the running license (the combined permanent license and time-based licenses), as well as the permanent license and each installed time-based license (active and inactive):

```
ciscoasa# show activation-key detail

Serial Number:  88810093382
Running Permanent Activation Key: 0xce06dc6b 0x8a7b5ab7 0xa1e21dd4 0xd2c4b8b8 0xc4594f9c
Running Timebased Activation Key: 0xa821d549 0x35725fe4 0xc918b97b 0xce0b987b 0x47c7c285

Licensed features for this platform:
Maximum Physical Interfaces    : 8              perpetual
VLANs                          : 20             DMZ Unrestricted
Dual ISPs                      : Enabled        perpetual
VLAN Trunk Ports               : 8              perpetual
Inside Hosts                   : Unlimited      perpetual
Failover                       : Active/Standby perpetual
VPN-DES                        : Enabled        perpetual
VPN-3DES-AES                   : Enabled        perpetual
AnyConnect Premium Peers          : 2              perpetual
AnyConnect Essentials             : Disabled       perpetual
Other VPN Peers                   : 25             perpetual
Total VPN Peers                   : 25              perpetual
AnyConnect for Mobile             : Disabled       perpetual
AnyConnect for Cisco VPN Phone    : Disabled       perpetual
Advanced Endpoint Assessment      : Disabled       perpetual
UC Phone Proxy Sessions        : 2              perpetual
Total UC Proxy Sessions        : 2              perpetual
Botnet Traffic Filter          : Enabled        39 days
Intercompany Media Engine      : Disabled       perpetual

This platform has an ASA 5512-X Security Plus license.

Running Permanent Activation Key: 0xce06dc6b 0x8a7b5ab7 0xa1e21dd4 0xd2c4b8b8 0xc4594f9c

Licensed features for this platform:
Maximum Physical Interfaces    : 8              perpetual
VLANs                          : 20             DMZ Unrestricted
Dual ISPs                      : Enabled        perpetual
VLAN Trunk Ports               : 8              perpetual
Inside Hosts                   : Unlimited      perpetual
Failover                       : Active/Standby perpetual
VPN-DES                        : Enabled        perpetual
VPN-3DES-AES                   : Enabled        perpetual
AnyConnect Premium Peers          : 2              perpetual
AnyConnect Essentials             : Disabled       perpetual
Other VPN Peers                   : 25             perpetual
Total VPN Peers                   : 25              perpetual
AnyConnect for Mobile             : Disabled       perpetual
AnyConnect for Cisco VPN Phone    : Disabled       perpetual
Advanced Endpoint Assessment      : Disabled       perpetual
UC Phone Proxy Sessions        : 2              perpetual
Total UC Proxy Sessions        : 2              perpetual
Botnet Traffic Filter          : Enabled        39 days
Intercompany Media Engine      : Disabled       perpetual

The flash permanent activation key is the SAME as the running permanent key.

Active Timebased Activation Key:
0xa821d549 0x35725fe4 0xc918b97b 0xce0b987b 0x47c7c285
Botnet Traffic Filter           : Enabled    39 days

Inactive Timebased Activation Key:
0xyadayada3 0xyadayada3 0xyadayada3 0xyadayada3 0xyadayada3
AnyConnect Premium Peers                  : 25     7 days
```

Example 3: Primary Unit Output in a Failover Pair for show activation-key detail

The following is sample output from the **show activation-key detail** command for the primary failover unit that shows:

- The primary unit license (the combined permanent license and time-based licenses).
- The "Failover Cluster" license, which is the combined licenses from the primary and secondary units. This is the license that is actually running on the ASA. The values in this license that reflect the combination of the primary and secondary licenses are in bold.
- The primary unit permanent license.
- The primary unit installed time-based licenses (active and inactive).

```
ciscoasa# show activation-key detail

Serial Number:  P3000000171
Running Permanent Activation Key: 0xce06dc6b 0x8a7b5ab7 0xa1e21dd4 0xd2c4b8b8 0xc4594f9c
Running Timebased Activation Key: 0xa821d549 0x35725fe4 0xc918b97b 0xce0b987b 0x47c7c285

Licensed features for this platform:
Maximum Physical Interfaces    : Unlimited      perpetual
Maximum VLANs                  : 150            perpetual
Inside Hosts                   : Unlimited      perpetual
Failover                       : Active/Active  perpetual
VPN-DES                        : Enabled        perpetual
VPN-3DES-AES                   : Enabled        perpetual
Security Contexts              : 12             perpetual
GTP/GPRS                       : Enabled        perpetual
AnyConnect Premium Peers          : 2              perpetual
AnyConnect Essentials             : Disabled       perpetual
Other VPN Peers                   : 750            perpetual
Total VPN Peers                   : 750            perpetual
Shared License                    : Disabled        perpetual
AnyConnect for Mobile             : Disabled       perpetual
AnyConnect for Cisco VPN Phone    : Disabled       perpetual
Advanced Endpoint Assessment      : Disabled       perpetual
UC Phone Proxy Sessions        : 2              perpetual
Total UC Proxy Sessions        : 2              perpetual
Botnet Traffic Filter          : Enabled        33 days
Intercompany Media Engine      : Disabled       perpetual

This platform has an ASA 5520 VPN Plus license.

Failover cluster licensed features for this platform:
Maximum Physical Interfaces    : Unlimited      perpetual
Maximum VLANs                  : 150            perpetual
Inside Hosts                   : Unlimited      perpetual
Failover                       : Active/Active  perpetual
VPN-DES                        : Enabled        perpetual
VPN-3DES-AES                   : Enabled        perpetual
Security Contexts              : 12             perpetual
GTP/GPRS                       : Enabled        perpetual
AnyConnect Premium Peers       : 4           perpetual
AnyConnect Essentials             : Disabled       perpetual
Other VPN Peers                   : 750            perpetual
Total VPN Peers                   : 750            perpetual
Shared License                    : Disabled       perpetual
AnyConnect for Mobile             : Disabled       perpetual
AnyConnect for Cisco VPN Phone    : Disabled       perpetual
Advanced Endpoint Assessment      : Disabled       perpetual
```

```
UC Phone Proxy Sessions        : 4              perpetual
Total UC Proxy Sessions        : 4              perpetual
Botnet Traffic Filter          : Enabled        33 days
Intercompany Media Engine      : Disabled       perpetual

This platform has an ASA 5520 VPN Plus license.

Running Permanent Activation Key: 0xce06dc6b 0x8a7b5ab7 0xa1e21dd4 0xd2c4b8b8 0xc4594f9c

Licensed features for this platform:
Maximum Physical Interfaces    : Unlimited      perpetual
Maximum VLANs                  : 150            perpetual
Inside Hosts                   : Unlimited      perpetual
Failover                       : Active/Active  perpetual
VPN-DES                        : Enabled        perpetual
VPN-3DES-AES                   : Disabled       perpetual
Security Contexts              : 2              perpetual
GTP/GPRS                       : Disabled       perpetual
AnyConnect Premium Peers          : 2              perpetual
AnyConnect Essentials             : Disabled       perpetual
Other VPN Peers                   : 750            perpetual
Total VPN Peers                   : 750            perpetual
Shared License                    : Disabled        perpetual
AnyConnect for Mobile             : Disabled       perpetual
AnyConnect for Cisco VPN Phone    : Disabled       perpetual
Advanced Endpoint Assessment      : Disabled       perpetual
UC Phone Proxy Sessions        : 2              perpetual
Total UC Proxy Sessions        : 2              perpetual
Botnet Traffic Filter          : Disabled       perpetual
Intercompany Media Engine      : Disabled       perpetual

The flash permanent activation key is the SAME as the running permanent key.

Active Timebased Activation Key:
0xa821d549 0x35725fe4 0xc918b97b 0xce0b987b 0x47c7c285
Botnet Traffic Filter           : Enabled    33 days

Inactive Timebased Activation Key:
0xyadayad3 0xyadayad3 0xyadayad3 0xyadayad3 0xyadayad3
Security Contexts               : 2          7 days
AnyConnect Premium Peers                   : 100        7 days

0xyadayad4 0xyadayad4 0xyadayad4 0xyadayad4 0xyadayad4
Total UC Proxy Sessions         : 100        14 days
```

Example 4: Secondary Unit Output in a Failover Pair for show activation-key detail

The following is sample output from the **show activation-key detail** command for the secondary failover unit that shows:

- The secondary unit license (the combined permanent license and time-based licenses).
- The “Failover Cluster” license, which is the combined licenses from the primary and secondary units. This is the license that is actually running on the ASA. The values in this license that reflect the combination of the primary and secondary licenses are in bold.
- The secondary unit permanent license.
- The secondary installed time-based licenses (active and inactive). This unit does not have any time-based licenses, so none display in this sample output.

```
ciscoasa# show activation-key detail

Serial Number:  P3000000011
Running Activation Key: 0xyadayad1 0xyadayad1 0xyadayad1 0xyadayad1 0xyadayad1

Licensed features for this platform:
Maximum Physical Interfaces    : Unlimited      perpetual
Maximum VLANs                  : 150            perpetual
Inside Hosts                   : Unlimited      perpetual
Failover                       : Active/Active  perpetual
VPN-DES                        : Enabled        perpetual
VPN-3DES-AES                   : Disabled       perpetual
Security Contexts              : 2              perpetual
GTP/GPRS                       : Disabled       perpetual
AnyConnect Premium Peers          : 2              perpetual
AnyConnect Essentials             : Disabled       perpetual
Other VPN Peers                   : 750            perpetual
Total VPN Peers                   : 750            perpetual
Shared License                    : Disabled        perpetual
AnyConnect for Mobile             : Disabled       perpetual
AnyConnect for Cisco VPN Phone    : Disabled       perpetual
Advanced Endpoint Assessment      : Disabled       perpetual
UC Phone Proxy Sessions        : 2              perpetual
Total UC Proxy Sessions        : 2              perpetual
Botnet Traffic Filter          : Disabled       perpetual
Intercompany Media Engine      : Disabled       perpetual

This platform has an ASA 5520 VPN Plus license.

Failover cluster licensed features for this platform:
Maximum Physical Interfaces    : Unlimited      perpetual
Maximum VLANs                  : 150            perpetual
Inside Hosts                   : Unlimited      perpetual
Failover                       : Active/Active  perpetual
VPN-DES                        : Enabled        perpetual
VPN-3DES-AES                   : Enabled        perpetual
Security Contexts              : 10             perpetual
GTP/GPRS                       : Enabled        perpetual
AnyConnect Premium Peers          : 4              perpetual
AnyConnect Essentials             : Disabled       perpetual
Other VPN Peers                   : 750            perpetual
Total VPN Peers                   : 750            perpetual
Shared License                    : Disabled        perpetual
AnyConnect for Mobile             : Disabled       perpetual
AnyConnect for Cisco VPN Phone    : Disabled       perpetual
Advanced Endpoint Assessment      : Disabled       perpetual
UC Phone Proxy Sessions        : 4              perpetual
Total UC Proxy Sessions        : 4              perpetual
Botnet Traffic Filter          : Enabled        33 days
Intercompany Media Engine      : Disabled       perpetual

This platform has an ASA 5520 VPN Plus license.

Running Permanent Activation Key: 0xyadayad1 0xyadayad1 0xyadayad1 0xyadayad1 0xyadayad1

Licensed features for this platform:
Maximum Physical Interfaces    : Unlimited      perpetual
Maximum VLANs                  : 150            perpetual
Inside Hosts                   : Unlimited      perpetual
Failover                       : Active/Active  perpetual
VPN-DES                        : Enabled        perpetual
VPN-3DES-AES                   : Disabled       perpetual
Security Contexts              : 2              perpetual
```

```
GTP/GPRS                        : Disabled       perpetual
AnyConnect Premium Peers          : 2              perpetual
AnyConnect Essentials             : Disabled       perpetual
Other VPN Peers                   : 750            perpetual
Total VPN Peers                   : 750            perpetual
Shared License                    : Disabled        perpetual
AnyConnect for Mobile             : Disabled       perpetual
AnyConnect for Cisco VPN Phone    : Disabled       perpetual
Advanced Endpoint Assessment      : Disabled       perpetual
UC Phone Proxy Sessions         : 2              perpetual
Total UC Proxy Sessions         : 2              perpetual
Botnet Traffic Filter           : Disabled       perpetual
Intercompany Media Engine       : Disabled       perpetual

The flash permanent activation key is the SAME as the running permanent key.
```

Example 5: Primary Unit Output for the ASA Services Module in a Failover Pair for show activation-key

The following is sample output from the **show activation-key** command for the primary failover unit that shows:

- The primary unit license (the combined permanent license and time-based licenses).
- The "Failover Cluster" license, which is the combined licenses from the primary and secondary units. This is the license that is actually running on the ASA. The values in this license that reflect the combination of the primary and secondary licenses are in bold.
- The primary unit installed time-based licenses (active and inactive).

```
ciscoasa# show activation-key

erial Number:  SAL144705BF
Running Permanent Activation Key: 0x4d1ed752 0xc8cfeb37 0xf4c38198 0x93c04c28 0x4a1c049a
Running Timebased Activation Key: 0xbc07bbd7 0xb15591e0 0xed68c013 0xd79374ff 0x44f87880

Licensed features for this platform:
Maximum Interfaces                : 1024           perpetual
Inside Hosts                      : Unlimited      perpetual
Failover                          : Active/Active  perpetual
DES                               : Enabled        perpetual
3DES-AES                          : Enabled        perpetual
Security Contexts                 : 25             perpetual
GTP/GPRS                          : Enabled        perpetual
Botnet Traffic Filter             : Enabled        330 days

This platform has an WS-SVC-ASA-SM1 No Payload Encryption license.


Failover cluster licensed features for this platform:
Maximum Interfaces                : 1024           perpetual
Inside Hosts                      : Unlimited      perpetual
Failover                          : Active/Active  perpetual
DES                               : Enabled        perpetual
3DES-AES                          : Enabled        perpetual
Security Contexts                 : 50 perpetual
GTP/GPRS                          : Enabled        perpetual
Botnet Traffic Filter             : Enabled        330 days
This platform has an WS-SVC-ASA-SM1 No Payload Encryption license.

The flash permanent activation key is the SAME as the running permanent key.
```

```
Active Timebased Activation Key:
0xbc07bbd7 0xb15591e0 0xed68c013 0xd79374ff 0x44f87880
Botnet Traffic Filter           : Enabled    330 days
```

Example 6: Secondary Unit Output for the ASA Services Module in a Failover Pair for show activation-key

The following is sample output from the **show activation-key** command for the secondary failover unit that shows:

- The secondary unit license (the combined permanent license and time-based licenses).
- The "Failover Cluster" license, which is the combined licenses from the primary and secondary units. This is the license that is actually running on the ASA. The values in this license that reflect the combination of the primary and secondary licenses are in bold.
- The secondary installed time-based licenses (active and inactive). This unit does not have any time-based licenses, so none display in this sample output.

```
ciscoasa# show activation-key detail

Serial Number:  SAD143502E3
Running Permanent Activation Key: 0xf404c46a 0xb8e5bd84 0x28c1b900 0x92eca09c 0x4e2a0683

Licensed features for this platform:
Maximum Interfaces                : 1024           perpetual
Inside Hosts                      : Unlimited      perpetual
Failover                          : Active/Active  perpetual
DES                               : Enabled        perpetual
3DES-AES                          : Enabled        perpetual
Security Contexts                 : 25             perpetual
GTP/GPRS                          : Disabled       perpetual
Botnet Traffic Filter             : Disabled       perpetual

This platform has an WS-SVC-ASA-SM1 No Payload Encryption license.


Failover cluster licensed features for this platform:
Maximum Interfaces                : 1024           perpetual
Inside Hosts                      : Unlimited      perpetual
Failover                          : Active/Active  perpetual
DES                               : Enabled        perpetual
3DES-AES                          : Enabled        perpetual
Security Contexts                 : 50 perpetual
GTP/GPRS                          : Enabled perpetual
Botnet Traffic Filter             : Enabled 330 days

This platform has an WS-SVC-ASA-SM1 No Payload Encryption license.

The flash permanent activation key is the SAME as the running permanent key.
```

Example 7: Output in a Cluster for show activation-key

```
ciscoasa# show activation-key
 Serial Number: JMX1504L2TD
 Running Permanent Activation Key: 0x4a3eea7b 0x54b9f61a 0x4143a90c 0xe5849088 0x4412d4a9

 Licensed features for this platform:
```

```
Maximum Physical Interfaces : Unlimited perpetual
Maximum VLANs : 100 perpetual
Inside Hosts : Unlimited perpetual
Failover : Active/Active perpetual
Encryption-DES : Enabled perpetual
Encryption-3DES-AES : Enabled perpetual
Security Contexts : 2 perpetual
GTP/GPRS : Disabled perpetual
AnyConnect Premium Peers : 2 perpetual
AnyConnect Essentials : Disabled perpetual
Other VPN Peers : 250 perpetual
Total VPN Peers : 250 perpetual
Shared License : Disabled perpetual
AnyConnect for Mobile : Disabled perpetual
AnyConnect for Cisco VPN Phone : Disabled perpetual
Advanced Endpoint Assessment : Disabled perpetual
UC Phone Proxy Sessions : 2 perpetual
Total UC Proxy Sessions : 2 perpetual
Botnet Traffic Filter : Disabled perpetual
Intercompany Media Engine : Disabled perpetual
Cluster : Enabled perpetual

This platform has an ASA 5585-X base license.

Failover cluster licensed features for this platform:
Maximum Physical Interfaces : Unlimited perpetual
Maximum VLANs : 100 perpetual
Inside Hosts : Unlimited perpetual
Failover : Active/Active perpetual
Encryption-DES : Enabled perpetual
Encryption-3DES-AES : Enabled perpetual
Security Contexts : 4 perpetual
GTP/GPRS : Disabled perpetual
AnyConnect Premium Peers : 4 perpetual
AnyConnect Essentials : Disabled perpetual
Other VPN Peers : 250 perpetual
Total VPN Peers : 250 perpetual
Shared License : Disabled perpetual
AnyConnect for Mobile : Disabled perpetual
AnyConnect for Cisco VPN Phone : Disabled perpetual
Advanced Endpoint Assessment : Disabled perpetual
UC Phone Proxy Sessions : 4 perpetual
Total UC Proxy Sessions : 4 perpetual
Botnet Traffic Filter : Disabled perpetual
Intercompany Media Engine : Disabled perpetual
Cluster : Enabled perpetual

This platform has an ASA 5585-X base license.

The flash permanent activation key is the SAME as the running permanent key.
```

Monitoring the Shared License

To monitor the shared license, enter one of the following commands.

- **show shared license** [**detail** | **client** [*hostname*] | **backup**]

  Shows shared license statistics. Optional keywords are available only for the licensing server: the **detail** keyword shows statistics per participant. To limit the display to one participant, use the **client** keyword. The **backup** keyword shows information about the backup server.

  To clear the shared license statistics, enter the **clear shared license** command.

The following is sample output from the **show shared license** command on the license participant:

```
ciscoasa>  show shared license
Primary License Server : 10.3.32.20
  Version              : 1
  Status               : Inactive

Shared license utilization:
  SSLVPN:
    Total for network :     5000
    Available         :     5000
    Utilized          :        0
  This device:
    Platform limit    :      250
    Current usage     :        0
    High usage        :        0
  Messages Tx/Rx/Error:
    Registration    : 0 / 0 / 0
    Get             : 0 / 0 / 0
    Release         : 0 / 0 / 0
    Transfer        : 0 / 0 / 0
```

The following is sample output from the **show shared license detail** command on the license server:

```
ciscoasa>  show shared license detail
Backup License Server Info:

Device ID           : ABCD
Address             : 10.1.1.2
Registered          : NO
HA peer ID          : EFGH
Registered          : NO
  Messages Tx/Rx/Error:
    Hello           : 0 / 0 / 0
    Sync            : 0 / 0 / 0
    Update          : 0 / 0 / 0

Shared license utilization:
  SSLVPN:
    Total for network :      500
    Available         :      500
    Utilized          :        0
  This device:
    Platform limit    :      250
    Current usage     :        0
    High usage        :        0
  Messages Tx/Rx/Error:
    Registration    : 0 / 0 / 0
    Get             : 0 / 0 / 0
    Release         : 0 / 0 / 0
    Transfer        : 0 / 0 / 0


Client Info:

  Hostname          : 5540-A
  Device ID         : XXXXXXXXXXX
  SSLVPN:
    Current usage   : 0
    High            : 0
  Messages Tx/Rx/Error:
    Registration    : 1 / 1 / 0
    Get             : 0 / 0 / 0
```

```
    Release         : 0 / 0 / 0
    Transfer        : 0 / 0 / 0
...
```

- **show activation-key**

  Shows the licenses installed on the ASA. The **show version** command also shows license information.

- **show vpn-sessiondb**

  Shows license information about VPN sessions.

History for PAK Licenses

Feature Name	Platform Releases	Description
Increased Connections and VLANs	7.0(5)	Increased the following limits: • ASA5510 Base license connections from 32000 to 5000; VLANs from 0 to 10. • ASA5510 Security Plus license connections from 64000 to 130000; VLANs from 10 to 25. • ASA5520 connections from 130000 to 280000; VLANs from 25 to 100. • ASA5540 connections from 280000 to 400000; VLANs from 100 to 200.
SSL VPN Licenses	7.1(1)	SSL VPN licenses were introduced.
Increased SSL VPN Licenses	7.2(1)	A 5000-user SSL VPN license was introduced for the ASA 5550 and above.
Increased interfaces for the Base license on the ASA 5510	7.2(2)	For the Base license on the ASA 5510, the maximum number of interfaces was increased from 3 plus a management interface to unlimited interfaces.

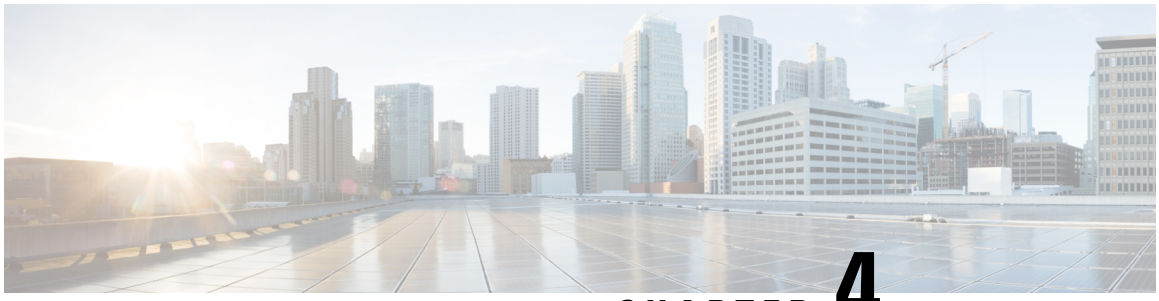
Feature Name	Platform Releases	Description
Increased VLANs	7.2(2)	The maximum number of VLANs for the Security Plus license on the ASA 5505 was increased from 5 (3 fully functional; 1 failover; one restricted to a backup interface) to 20 fully functional interfaces. In addition, the number of trunk ports was increased from 1 to 8. Now there are 20 fully functional interfaces, you do not need to use the backup interface command to cripple a backup ISP interface; you can use a fully functional interface for it. The backup interface command is still useful for an Easy VPN configuration. VLAN limits were also increased for the ASA 5510 (from 10 to 50 for the Base license, and from 25 to 100 for the Security Plus license), the ASA 5520 (from 100 to 150), the ASA 5550 (from 200 to 250).
Gigabit Ethernet Support for the ASA 5510 Security Plus License	7.2(3)	The ASA 5510 now supports Gigabit Ethernet (1000 Mbps) for the Ethernet 0/0 and 0/1 ports with the Security Plus license. In the Base license, they continue to be used as Fast Ethernet (100 Mbps) ports. Ethernet 0/2, 0/3, and 0/4 remain as Fast Ethernet ports for both licenses. **Note** The interface names remain Ethernet 0/0 and Ethernet 0/1. Use the **speed** command to change the speed on the interface and use the **show interface** command to see what speed is currently configured for each interface.
Advanced Endpoint Assessment License	8.0(2)	The Advanced Endpoint Assessment license was introduced. As a condition for the completion of a Cisco AnyConnect or clientless SSL VPN connections, the remote computer scans for a greatly expanded collection of antivirus and antispyware applications, firewalls, operating systems, and associated updates. It also scans for any registry entries, filenames, and process names that you specify. It sends the scan results to the ASA. The ASA uses both the user login credentials and the computer scan results to assign a Dynamic Access Policy (DAP). With an Advanced Endpoint Assessment License, you can enhance Host Scan by configuring an attempt to update noncompliant computers to meet version requirements. Cisco can provide timely updates to the list of applications and versions that Host Scan supports in a package that is separate from Cisco Secure Desktop.
VPN Load Balancing for the ASA 5510	8.0(2)	VPN load balancing is now supported on the ASA 5510 Security Plus license.

Feature Name	Platform Releases	Description
AnyConnect for Mobile License	8.0(3)	The AnyConnect for Mobile license was introduced. It lets Windows mobile devices connect to the ASA using the AnyConnect Client.
Time-based Licenses	8.0(4)/8.1(2)	Support for time-based licenses was introduced.
Increased VLANs for the ASA 5580	8.1(2)	The number of VLANs supported on the ASA 5580 are increased from 100 to 250.
Unified Communications Proxy Sessions license	8.0(4)	The UC Proxy sessions license was introduced. Phone Proxy, Presence Federation Proxy, and Encrypted Voice Inspection applications use TLS proxy sessions for their connections. Each TLS proxy session is counted against the UC license limit. All of these applications are licensed under the UC Proxy umbrella, and can be mixed and matched. This feature is not available in Version 8.1.
Botnet Traffic Filter License	8.2(1)	The Botnet Traffic Filter license was introduced. The Botnet Traffic Filter protects against malware network activity by tracking connections to known bad domains and IP addresses.
AnyConnect Essentials License	8.2(1)	The AnyConnect Essentials License was introduced. This license enables AnyConnect VPN client access to the ASA. This license does not support browser-based SSL VPN access or Cisco Secure Desktop. For these features, activate an AnyConnect Premium license instead of the AnyConnect Essentials license. **Note** With the AnyConnect Essentials license, VPN users can use a Web browser to log in, and download and start (WebLaunch) the AnyConnect Client. The AnyConnect Client software offers the same set of client features, whether it is enabled by this license or an AnyConnect Premium license. The AnyConnect Essentials license cannot be active at the same time as the following licenses on a given ASA: AnyConnect Premium license (all types) or the Advanced Endpoint Assessment license. You can, however, run AnyConnect Essentials and AnyConnect Premium licenses on different ASAs in the same network. By default, the ASA uses the AnyConnect Essentials license, but you can disable it to use other licenses by using the **webvpn**, and then the **no anyconnect-essentials** command.
SSL VPN license changed to AnyConnect Premium SSL VPN Edition license	8.2(1)	The SSL VPN license name was changed to the AnyConnect Premium SSL VPN Edition license.

Feature Name	Platform Releases	Description
Shared Licenses for SSL VPN	8.2(1)	Shared licenses for SSL VPN were introduced. Multiple ASAs can share a pool of SSL VPN sessions on an as-needed basis.
Mobility Proxy application no longer requires Unified Communications Proxy license	8.2(2)	The Mobility Proxy no longer requires the UC Proxy license.
10 GE I/O license for the ASA 5585-X with SSP-20	8.2(3)	We introduced the 10 GE I/O license for the ASA 5585-X with SSP-20 to enable 10-Gigabit Ethernet speeds for the fiber ports. The SSP-60 supports 10-Gigabit Ethernet speeds by default. **Note** The ASA 5585-X is not supported in 8.3(x).
10 GE I/O license for the ASA 5585-X with SSP-10	8.2(4)	We introduced the 10 GE I/O license for the ASA 5585-X with SSP-10 to enable 10-Gigabit Ethernet speeds for the fiber ports. The SSP-40 supports 10-Gigabit Ethernet speeds by default. **Note** The ASA 5585-X is not supported in 8.3(x).
Non-identical failover licenses	8.3(1)	Failover licenses no longer need to be identical on each unit. The license used for both units is the combined license from the primary and secondary units. We modified the following commands: **show activation-key** and **show version**.
Stackable time-based licenses	8.3(1)	Time-based licenses are now stackable. In many cases, you might need to renew your time-based license and have a seamless transition from the old license to the new one. For features that are only available with a time-based license, it is especially important that the license not expire before you can apply the new license. The ASA allows you to *stack* time-based licenses so that you do not have to worry about the license expiring or about losing time on your licenses because you installed the new one early.
Intercompany Media Engine License	8.3(1)	The IME license was introduced.
Multiple time-based licenses active at the same time	8.3(1)	You can now install multiple time-based licenses, and have one license per feature active at a time. We modified the following commands: **show activation-key** and **show version**.
Discrete activation and deactivation of time-based licenses.	8.3(1)	You can now activate or deactivate time-based licenses using a command. We modified the following commands: **activation-key** [**activate** | **deactivate**].

Feature Name	Platform Releases	Description
AnyConnect Premium SSL VPN Edition license changed to AnyConnect Premium SSL VPN license	8.3(1)	The AnyConnect Premium SSL VPN Edition license name was changed to the AnyConnect Premium SSL VPN license.
No Payload Encryption image for export	8.3(2)	If you install the No Payload Encryption software on the ASA 5505 through 5550, then you disable Unified Communications, strong encryption VPN, and strong encryption management protocols. **Note** This special image is only supported in 8.3(x); for No Payload Encryption support in 8.4(1) and later, you need to purchase a special hardware version of the ASA.
Increased contexts for the ASA 5550, 5580, and 5585-X	8.4(1)	For the ASA 5550 and ASA 5585-X with SSP-10, the maximum contexts was increased from 50 to 100. For the ASA 5580 and 5585-X with SSP-20 and higher, the maximum was increased from 50 to 250.
Increased VLANs for the ASA 5580 and 5585-X	8.4(1)	For the ASA 5580 and 5585-X, the maximum VLANs was increased from 250 to 1024.
Increased connections for the ASA 5580 and 5585-X	8.4(1)	We increased the firewall connection limits: • ASA 5580-20—1,000,000 to 2,000,000. • ASA 5580-40—2,000,000 to 4,000,000. • ASA 5585-X with SSP-10: 750,000 to 1,000,000. • ASA 5585-X with SSP-20: 1,000,000 to 2,000,000. • ASA 5585-X with SSP-40: 2,000,000 to 4,000,000. • ASA 5585-X with SSP-60: 2,000,000 to 10,000,000.
AnyConnect Premium SSL VPN license changed to AnyConnect Premium license	8.4(1)	The AnyConnect Premium SSL VPN license name was changed to the AnyConnect Premium license. The license information display was changed from “SSL VPN Peers” to “AnyConnect Premium Peers.”
Increased AnyConnect VPN sessions for the ASA 5580	8.4(1)	The AnyConnect VPN session limit was increased from 5,000 to 10,000.
Increased Other VPN sessions for the ASA 5580	8.4(1)	The other VPN session limit was increased from 5,000 to 10,000.

Feature Name	Platform Releases	Description
IPsec remote access VPN using IKEv2	8.4(1)	IPsec remote access VPN using IKEv2 was added to the AnyConnect Essentials and AnyConnect Premium licenses. **Note** The following limitation exists in our support for IKEv2 on the ASA: We currently do not support duplicate security associations. IKEv2 site-to-site sessions were added to the Other VPN license (formerly IPsec VPN). The Other VPN license is included in the Base license.
No Payload Encryption hardware for export	8.4(1)	For models available with No Payload Encryption (for example, the ASA 5585-X), the ASA software disables Unified Communications and VPN features, making the ASA available for export to certain countries.
Dual SSPs for SSP-20 and SSP-40	8.4(2)	For SSP-40 and SSP-60, you can use two SSPs of the same level in the same chassis. Mixed-level SSPs are not supported (for example, an SSP-40 with an SSP-60 is not supported). Each SSP acts as an independent device, with separate configurations and management. You can use the two SSPs as a failover pair if desired. When using two SSPs in the chassis, VPN is not supported; note, however, that VPN has not been disabled.
IPS Module license for the ASA 5512-X through ASA 5555-X	8.6(1)	The IPS SSP software module on the ASA 5512-X, ASA 5515-X, ASA 5525-X, ASA 5545-X, and ASA 5555-X requires the IPS module license.
Clustering license for the ASA 5580 and ASA 5585-X.	9.0(1)	A clustering license was added for the ASA 5580 and ASA 5585-X.
Support for VPN on the ASASM	9.0(1)	The ASASM now supports all VPN features.
Unified communications support on the ASASM	9.0(1)	The ASASM now supports all Unified Communications features.
ASA 5585-X Dual SSP support for the SSP-10 and SSP-20 (in addition to the SSP-40 and SSP-60); VPN support for Dual SSPs	9.0(1)	The ASA 5585-X now supports dual SSPs using all SSP models (you can use two SSPs of the same level in the same chassis). VPN is now supported when using dual SSPs.
ASA 5500-X support for clustering	9.1(4)	The ASA 5512-X, ASA 5515-X, ASA 5525-X, ASA 5545-X, and ASA 5555-X now support 2-unit clusters. Clustering for 2 units is enabled by default in the base license; for the ASA 5512-X, you need the Security Plus license.
Support for 16 cluster members for the ASA 5585-X	9.2(1)	The ASA 5585-X now supports 16-unit clusters.
ASAv4 and ASAv30 Standard and Premium model licenses introduced	9.2(1)	The ASAv was introduced with a simple licensing scheme: ASAv4 and ASAv30 permanent licenses in Standard or Premium levels. No add-on licenses are available.



CHAPTER 4

Licenses: Smart Software Licensing

Smart Software Licensing lets you purchase and manage a pool of licenses centrally. Unlike product authorization key (PAK) licenses, smart licenses are not tied to a specific serial number. You can easily deploy or retire ASAs without having to manage each unit's license key. Smart Software Licensing also lets you see your license usage and needs at a glance.

Note Smart Software Licensing is not supported on the ISA 3000. They use PAK licenses. See About PAK Licenses, on page 49.

For more information about Smart Licensing features and behaviors per platform, see Smart Enabled Product Families.

- About Smart Software Licensing, on page 87
- Prerequisites for Smart Software Licensing, on page 108
- Guidelines for Smart Software Licensing, on page 109
- Defaults for Smart Software Licensing, on page 109
- ASAv: Configure Smart Software Licensing, on page 109
- Firepower 4100/9300: Configure Smart Software Licensing, on page 127
- Other Models: Configure Smart Software Licensing, on page 129
- Licenses Per Model, on page 142
- Monitoring Smart Software Licensing, on page 154
- Smart Software Manager Communication, on page 158
- History for Smart Software Licensing, on page 160

About Smart Software Licensing

Cisco Smart Licensing is a flexible licensing model that provides you with an easier, faster, and more consistent way to purchase and manage software across the Cisco portfolio and across your organization. And it's secure—you control what users can access. With Smart Licensing you get:

- **Easy Activation:** Smart Licensing establishes a pool of software licenses that can be used across the entire organization—no more PAKs (Product Activation Keys).
- **Unified Management:** Cisco License Central provides a complete view into all of your Cisco products and services in an easy-to-use portal, so you always know what you have and what you are using.

- **License Flexibility:** Your license is not node-locked to your hardware, so you can easily use and transfer licenses as needed.

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 the Licensing Hub.

Smart Software Licensing for the ASA on the Firepower 4100/9300 Chassis

For the ASA on the Firepower 4100/9300 chassis, Smart Software Licensing configuration is split between the Firepower 4100/9300 chassis supervisor and the ASA.

- Firepower 4100/9300 chassis—Configure all Smart Software Licensing infrastructure on the chassis, including parameters for communicating with the Smart Software Manager. The Firepower 4100/9300 chassis itself does not require any licenses to operate. See the FXOS configuration guide for licensing procedures.

Note Inter-chassis clustering requires that you enable the same Smart Licensing method on each chassis in the cluster.

- ASA Application—Configure all license entitlements in the ASA.

Smart Software Manager and Accounts

When you purchase 1 or more licenses for the device, you manage them in the Cisco Smart Software Manager:

https://software.cisco.com/#module/SmartLicensing

The Smart Software Manager lets you create a master account for your organization.

Note If you do not yet have an account, click the link to set up a new account. The Smart Software Manager lets you create a master account for your organization.

By default, your licenses are assigned to the *Default Virtual Account* under your master account. As the account administrator, you can optionally create additional virtual accounts; for example, you can create accounts for regions, departments, or subsidiaries. Multiple virtual accounts let you more easily manage large numbers of licenses and devices.

Offline Management

If your devices do not have internet access, and cannot register with the Smart Software Manager, you must configure offline licensing.

Permanent License Reservation

If your devices cannot access the internet for security reasons, you can optionally request permanent licenses for each ASA. Permanent licenses do not require periodic access to the Smart Software Manager. As with

PAK licenses, you can purchase a license and install the license key for the ASA. Unlike a PAK license, you can obtain and manage the licenses with the Smart Software Manager. You can easily switch between regular smart licensing mode and permanent license reservation mode.

Note ASA does not support Specific License Reservation (SLR). In SLR, specific feature entitlements are enabled permanently. ASA supports only PLR, where all the features are enabled permanently.

ASA Virtual Permanent License Reservation

Note Permanent license reservation is supported only on VMware and KVM.

You can obtain a model-specific license that enables all of the following features:

- Maximum throughput for your model
- Standard tier
- Strong Encryption (3DES and AES) license, if you have enabled it in your Smart Licensing account
- AnyConnect Client capabilities enabled for the platform

Note Use of AnyConnect Client features is contingent on your purchase of the AnyConnect Client license that enables the right to use AnyConnect Client (see AnyConnect Plus, AnyConnect Apex, and AnyConnect VPN Only Licenses, on page 94).

Permanent License Reservation Mode Based on Memory and vCPU

You can configure a model-specific license based on the RAM and vCPUs assigned to your ASAv. For example, an ASAv with 8 GB RAM and 4 vCPUs always uses an ASAv30 license with 2G throughput.

The vCPU and memory-to-license relationships are as follows:

- 2 GB, 1 vCPU—ASAv10 (1G)
- 8 GB, 4 vCPUs —ASAv30 (2G)
- 16 GB, 8 vCPUs—ASAv50 (10G)
- 32 GB, 16 vCPUs—ASAv100 (20G)

Later, if you want to change the model level of a unit, you will have to return the current license and request a new license at the correct model level. To change the model of an already deployed ASAv, from the hypervisor you can change the vCPUs and DRAM settings to match the new model requirements. See the *ASAv Virtual Getting Started Guide* for these values.

If you stop using a license, you must return the license by generating a return code on ASAv and then enter that code into the Smart Software Manager. You must follow the return process correctly to ensure that you do not pay for unused licenses.

For more information about configuring permanent license reservation, see ASAv: Configure Permanent License Reservation, on page 122.

Firepower 1010 Permanent License Reservation

You can obtain a license that enables all features:

- Standard tier
- Security Plus
- Strong Encryption (3DES/AES) license if your account qualifies
- AnyConnect Client capabilities enabled to the platform maximum.

  Use of AnyConnect Client features is contingent on your purchase of the AnyConnect Client license that enables the right to use AnyConnect Client (see AnyConnect Plus, AnyConnect Apex, and AnyConnect VPN Only Licenses, on page 94).

Note You also need to request the entitlements in the ASA configuration so that the ASA allows their use.

If you stop using a license, you must return the license by generating a return code on the ASA, and then entering that code into the Smart Software Manager. Make sure you follow the return process correctly so you do not pay for unused licenses.

Firepower 1100 Permanent License Reservation

You can obtain a license that enables all features:

- Standard tier
- Maximum Security Contexts
- Strong Encryption (3DES/AES) license if your account qualifies
- AnyConnect Client capabilities enabled to the platform maximum.

  Use of AnyConnect Client features is contingent on your purchase of the AnyConnect Client license that enables the right to use AnyConnect Client (see AnyConnect Plus, AnyConnect Apex, and AnyConnect VPN Only Licenses, on page 94).

Note You also need to request the entitlements in the ASA configuration so that the ASA allows their use.

If you stop using a license, you must return the license by generating a return code on the ASA, and then entering that code into the Smart Software Manager. Make sure you follow the return process correctly so you do not pay for unused licenses.

Firepower 2100 Permanent License Reservation

You can obtain a license that enables all features:

- Standard tier
- Maximum Security Contexts
- Strong Encryption (3DES/AES) license if your account qualifies
- AnyConnect Client capabilities enabled to the platform maximum.

  Use of AnyConnect Client features is contingent on your purchase of the AnyConnect Client license that enables the right to use AnyConnect Client (see AnyConnect Plus, AnyConnect Apex, and AnyConnect VPN Only Licenses, on page 94).

Note You also need to request the entitlements in the ASA configuration so that the ASA allows their use.

If you stop using a license, you must return the license by generating a return code on the ASA, and then entering that code into the Smart Software Manager. Make sure you follow the return process correctly so you do not pay for unused licenses.

Secure Firewall 3100 Permanent License Reservation

You can obtain a license that enables all features:

- Standard tier
- Maximum Security Contexts
- Strong Encryption (3DES/AES) license if your account qualifies
- AnyConnect Client capabilities enabled to the platform maximum.

  Use of AnyConnect Client features is contingent on your purchase of the AnyConnect Client license that enables the right to use AnyConnect Client (see AnyConnect Plus, AnyConnect Apex, and AnyConnect VPN Only Licenses, on page 94).

Note You also need to request the entitlements in the ASA configuration so that the ASA allows their use.

If you stop using a license, you must return the license by generating a return code on the ASA, and then entering that code into the Smart Software Manager. Make sure you follow the return process correctly so you do not pay for unused licenses.

Firepower 4100/9300 Chassis Permanent License Reservation

You can obtain a license that enables all features:

- Standard tier.
- Maximum Security Contexts
- Carrier license

- Strong Encryption (3DES/AES) license if your account qualifies
- AnyConnect Client capabilities enabled to the platform maximum.

  Use of AnyConnect Client features is contingent on your purchase of the AnyConnect Client license that enables the right to use AnyConnect Client (see AnyConnect Plus, AnyConnect Apex, and AnyConnect VPN Only Licenses, on page 94).



Note The license is managed on the Firepower 4100/9300 chassis, but you also need to request the entitlements in the ASA configuration so that the ASA allows their use. To manage the license, see the FXOS configuration guide.

If you stop using a license, you must return the license by generating a return code on the Firepower 4100/9300 chassis, and then entering that code into the Smart Software Manager. Make sure you follow the return process correctly so you don't pay for unused licenses.

Smart Software Manager On-Prem

If your devices cannot access the internet for security reasons, you can optionally install a local Smart Software Manager On-Prem (formerly known as "Smart Software Satellite Server") server as a virtual machine (VM). The Smart Software Manager On-Prem provides a subset of Smart Software Manager functionality, and allows you to provide essential licensing services for all your local devices. Only the Smart Software Manager On-Prem needs to connect periodically to the main Smart Software Manager to sync your license usage. You can sync on a schedule or you can sync manually.

You can perform the following functions on the Smart Software Manager On-Prem:

- Activate or register a license
- View your company's licenses
- Transfer licenses between company entities

For more information, see Cisco Smart Software Manager On-Prem Data Sheet.

Licenses and Devices Managed per Virtual Account

Licenses and devices are managed per virtual account: only that virtual account's devices can use the licenses assigned to the account. If you need additional licenses, you can transfer an unused license from another virtual account. You can also transfer devices between virtual accounts.

For the ASA on the Firepower 4100/9300 chassis—Only the chassis registers as a device, while the ASA applications in the chassis request their own licenses. For example, for a Firepower 9300 chassis with 3 security modules, the chassis counts as one device, but the modules use 3 separate licenses.

Evaluation License

ASAv

The ASAv does not support an evaluation mode. Before the ASAv registers with the Smart Software Manager, it operates in a severely rate-limited state.

Firepower 1000

Before the Firepower 1000 registers with the Smart Software Manager, it operates for 90 days (total usage) in evaluation mode. Only default entitlements are enabled. When this period ends, the Firepower 1000 becomes out-of-compliance.



Note You cannot receive an evaluation license for Strong Encryption (3DES/AES); you must register with the Smart Software Manager to receive the export-compliance token that enables the Strong Encryption (3DES/AES) license.

Firepower 2100

Before the Firepower 2100 registers with the Smart Software Manager, it operates for 90 days (total usage) in evaluation mode. Only default entitlements are enabled. When this period ends, the Firepower 2100 becomes out-of-compliance.



Note You cannot receive an evaluation license for Strong Encryption (3DES/AES); you must register with the Smart Software Manager to receive the export-compliance token that enables the Strong Encryption (3DES/AES) license.

Secure Firewall 3100

Before the Secure Firewall 3100 registers with the Smart Software Manager, it operates for 90 days (total usage) in evaluation mode. Only default entitlements are enabled. When this period ends, the Secure Firewall 3100 becomes out-of-compliance.



Note You cannot receive an evaluation license for Strong Encryption (3DES/AES); you must register with the Smart Software Manager to receive the export-compliance token that enables the Strong Encryption (3DES/AES) license.

Firepower 4100/9300 Chassis

The Firepower 4100/9300 chassis supports two types of evaluation license:

- Chassis-level evaluation mode—Before the Firepower 4100/9300 chassis registers with the Smart Software Manager, it operates for 90 days (total usage) in evaluation mode. The ASA cannot request specific entitlements in this mode; only default entitlements are enabled. When this period ends, the Firepower 4100/9300 chassis becomes out-of-compliance.
- Entitlement-based evaluation mode—After the Firepower 4100/9300 chassis registers with the Smart Software Manager, you can obtain time-based evaluation licenses that can be assigned to the ASA. In the ASA, you request entitlements as usual. When the time-based license expires, you need to either renew the time-based license or obtain a permanent license.

Note You cannot receive an evaluation license for Strong Encryption (3DES/AES); you must register with the Smart Software Manager and obtain a permanent license to receive the export-compliance token that enables the Strong Encryption (3DES/AES) license.

About Licenses by Type

The following sections include additional information about licenses by type.

AnyConnect Plus, AnyConnect Apex, and AnyConnect VPN Only Licenses

AnyConnect Client licenses are not applied directly to the ASA. However, you need to purchase licenses and add them to your Smart Account to guarantee the right to use the ASA as the AnyConnect Client headend.

- For the AnyConnect Plus and AnyConnect Apex licenses, add up the number of peers you intend to use across all the ASAs in your Smart Account and purchase license(s) for that many peers.
- For the AnyConnect VPN Only, purchase one license per ASA. Unlike the other licenses that provide a pool of peers that can be shared by multiple ASAs, the AnyConnect VPN Only license is per headend.

For more information, see:

- Cisco AnyConnect Client Ordering Guide
- AnyConnect Client Licensing Frequently Asked Questions (FAQ)

Other VPN Peers

Other VPN peers include the following VPN types:

- IPsec remote access VPN using IKEv1
- IPsec site-to-site VPN using IKEv1
- IPsec site-to-site VPN using IKEv2

This license is included in the Base license.

Total VPN Peers Combined, All Types

- The Total VPN Peers is the maximum VPN peers allowed of both AnyConnect Client and Other VPN peers combined. For example, if the total is 1000, you can allow 500 AnyConnect Client and 500 Other VPN peers simultaneously; or 700 AnyConnect Client and 300 Other VPN; or use all 1000 for AnyConnect Client. If you exceed the total VPN peers, you can overload the ASA, so be sure to size your network appropriately.

Encryption License

Strong Encryption: ASAv

Strong Encryption (3DES/AES) is available for management connections before you connect to the Smart Software Manager or Smart Software Manager On-Prem server, so you can launch ASDM and connect to the Smart Software Manager. For through-the-box traffic that requires strong encryption (such as VPN), throughput is severely limited until you connect to the Smart Software Manager and obtain the Strong Encryption license.

When you request the registration token for the ASAv from your Smart Software Licensing account, check the **Allow export-controlled functionality on the products registered with this token** check box so that the Strong Encryption (3DES/AES) license is applied (your account must be qualified for its use). If the ASAv becomes out-of-compliance later, as long as the export compliance token was successfully applied, the ASAv will retain the license and not revert to the rate-limited state. The license is removed if you re-register the ASAv, and export compliance is disabled, or if you restore the ASAv to factory default settings.

If you initially register the ASAv without strong encryption and later add strong encryption, then you must reload the ASAv for the new license to take effect.

For permanent license reservation licenses, the Strong Encryption (3DES/AES) license is enabled if your account qualifies for its use.

If your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account.

Strong Encryption: Firepower 1000, Firepower 2100 in Appliance Mode, Secure Firewall 3100

The ASA includes 3DES capability by default for management access only, so you can connect to the Smart Software Manager and also use ASDM immediately. You can also use SSH and SCP if you later configure SSH access on the ASA. Other features that require strong encryption (such as VPN) must have Strong Encryption enabled, which requires you to first register to the Smart Software Manager.



Note If you attempt to configure any features that can use strong encryption before you register—even if you only configure weak encryption—then your HTTPS connection will be dropped on that interface, and you cannot reconnect. The exception to this rule is if you are connected to a management-only interface, such as Management 1/1. SSH is not affected. If you lose your HTTPS connection, you can connect to the console port to reconfigure the ASA, connect to a management-only interface, or connect to an interface not configured for a strong encryption feature.

When you request the registration token for the ASA from your Smart Software Licensing account, check the **Allow export-controlled functionality on the products registered with this token** check box so that the Strong Encryption (3DES/AES) license is applied (your account must be qualified for its use). If the ASA becomes out-of-compliance later, as long as the export compliance token was successfully applied, the ASA will continue to allow through the box traffic. Even if you re-register the ASA, and export compliance is disabled, the license remains enabled. The license is removed if you restore the ASA to factory default settings.

If you initially register the ASA without strong encryption and later add strong encryption, then you must reload the ASA for the new license to take effect.

For permanent license reservation licenses, the Strong Encryption (3DES/AES) license is enabled if your account qualifies for its use.

If your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account.



Note Reloading a device with an expired licensing registration for over a year, will disable the export-controlled functionality. This can lead to critical issues such as VPN downtime or failover split-brain scenarios when IPsec encryption is enabled.

Strong Encryption: Firepower 2100 in Platform Mode

Strong Encryption (3DES/AES) is available for management connections before you connect to the Smart Software Manager or Smart Software Manager On-Prem server so you can launch ASDM. Note that ASDM access is only available on management-only interfaces with the default encryption. Through the box traffic that requires strong encryption (such as VPN) is not allowed until you connect and obtain the Strong Encryption license.

When you request the registration token for the ASA from your Smart Software Licensing account, check the **Allow export-controlled functionality on the products registered with this token** check box so that the Strong Encryption (3DES/AES) license is applied (your account must be qualified for its use). If the ASA becomes out-of-compliance later, as long as the export compliance token was successfully applied, the ASA will continue to allow through the box traffic. Even if you re-register the ASA, and export compliance is disabled, the license remains enabled. The license is removed if you restore the ASA to factory default settings.

If you initially register the ASA without strong encryption and later add strong encryption, then you must reload the ASA for the new license to take effect.

For permanent license reservation licenses, the Strong Encryption (3DES/AES) license is enabled if your account qualifies for its use.

If your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account.

Strong Encryption: Firepower 4100/9300 Chassis

When the ASA is deployed as a logical device, you can launch ASDM immediately. Through the box traffic that requires strong encryption (such as VPN) is not allowed until you connect and obtain the Strong Encryption license.

When you request the registration token for the chassis from your Smart Software Licensing account, check the **Allow export-controlled functionality on the products registered with this token** check box so that the Strong Encryption (3DES/AES) license is applied (your account must be qualified for its use).

If the ASA becomes out-of-compliance later, as long as the export compliance token was successfully applied, the ASA will continue to allow through the box traffic. The license is removed if you re-register the chassis, and export compliance is disabled, or if you restore the chassis to factory default settings.

If you initially register the chassis without strong encryption and later add strong encryption, then you must reload the ASA application for the new license to take effect.

For permanent license reservation licenses, the Strong Encryption (3DES/AES) license is enabled if your account qualifies for its use.

If your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account.

DES: All Models

If you have Strong Encryption enabled, you cannot use DES.

Carrier License

The Carrier license enables the following inspection features:

- Diameter—Diameter is an Authentication, Authorization, and Accounting (AAA) protocol used in next-generation mobile and fixed telecom networks such as EPS (Evolved Packet System) for LTE (Long Term Evolution) and IMS (IP Multimedia Subsystem). It replaces RADIUS and TACACS in these networks.
- GTP/GPRS—GPRS Tunneling Protocol is used in GSM, UMTS and LTE networks for general packet radio service (GPRS) traffic. GTP provides a tunnel control and management protocol to provide GPRS network access for a mobile station by creating, modifying, and deleting tunnels. GTP also uses a tunneling mechanism for carrying user data packets.
- M3UA—MTP3 User Adaptation (M3UA) is a client/server protocol that provides a gateway to the SS7 network for IP-based applications that interface with the SS7 Message Transfer Part 3 (MTP3) layer. M3UA makes it possible to run the SS7 User Parts (such as ISUP) over an IP network. M3UA is defined in RFC 4666.
- SCTP—SCTP (Stream Control Transmission Protocol) is described in RFC 4960. The protocol supports the telephony signaling protocol SS7 over IP and is also a transport protocol for several interfaces in the 4G LTE mobile network architecture.

Total TLS Proxy Sessions

Each TLS proxy session for Encrypted Voice Inspection is counted against the TLS license limit.

Other applications that use TLS proxy sessions do not count toward the TLS limit, for example, Mobility Advantage Proxy (which does not require a license).

Some applications might use multiple sessions for a connection. For example, if you configure a phone with a primary and backup Cisco Unified Communications Manager, there are 2 TLS proxy connections.

You independently set the TLS proxy limit using the **tls-proxy maximum-sessions** command or in ASDM, using the **Configuration > Firewall > Unified Communications > TLS Proxy** pane. To view the limits of your model, enter the **tls-proxy maximum-sessions ?** command. When you apply a TLS proxy license that is higher than the default TLS proxy limit, the ASA automatically sets the TLS proxy limit to match the license. The TLS proxy limit takes precedence over the license limit; if you set the TLS proxy limit to be less than the license, then you cannot use all of the sessions in your license.

Note For license part numbers ending in “K8” (for example, licenses under 250 users), TLS proxy sessions are limited to 1000. For license part numbers ending in “K9” (for example, licenses 250 users or larger), the TLS proxy limit depends on the configuration, up to the model limit. K8 and K9 refer to whether the license is restricted for export: K8 is unrestricted, and K9 is restricted.

If you clear the configuration (using the **clear configure all** command, for example), then the TLS proxy limit is set to the default for your model; if this default is lower than the license limit, then you see an error message to use the **tls-proxy maximum-sessions** command to raise the limit again (in ASDM, use the **TLS Proxy** pane). If you use failover and enter the **write standby** command or in ASDM, use **File > Save Running Configuration to Standby Unit** on the primary unit to force a configuration synchronization, the **clear configure all** command is generated on the secondary unit automatically, so you may see the warning message on the secondary unit. Because the configuration synchronization restores the TLS proxy limit set on the primary unit, you can ignore the warning.

You might also use SRTP encryption sessions for your connections:

- For K8 licenses, SRTP sessions are limited to 250.
- For K9 licenses, there is no limit.

Note Only calls that require encryption/decryption for media are counted toward the SRTP limit; if passthrough is set for the call, even if both legs are SRTP, they do not count toward the limit.

VLANs, Maximum

For an interface to count against the VLAN limit, you must assign a VLAN to it. For example:

```
interface gigabitethernet 0/0.100
vlan 100
```

Botnet Traffic Filter License

Requires a Strong Encryption (3DES/AES) License to download the dynamic database.

Failover or ASA Cluster Licenses

Failover Licenses for the ASAv

The standby unit requires the same model license as the primary unit.

Failover Licenses for the Firepower 1010

Smart Software Manager Regular and On-Prem

Both Firepower 1010 units must be registered with the Smart Software Manager or Smart Software Manager On-Prem server. Both units require you to enable the Standard license and the Security Plus license *before* you can configure failover.

Typically, you do not also need to enable the Strong Encryption (3DES/AES) feature license in the ASA, because both units should have obtained the Strong Encryption token when you registered the units. When using the registration token, both units must have the same encryption level.

If your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. In this case, enable it on the active unit after you enable failover. The configuration is replicated to the standby unit, but the standby unit does not use the configuration; it remains in a cached state. Only the active unit requests the license from the server. The license is aggregated into a single failover license that is shared by the failover pair, and this aggregated license is also cached on the standby unit to be used if it becomes the active unit in the future. After a failover, the new active unit continues to use the aggregated license. It uses the cached license configuration to re-request the entitlement from the server. When the old active unit rejoins the pair as a standby unit, it releases the license entitlement. Before the standby unit releases the entitlement, the new active unit's license might be in a non-compliant state if there are no available licenses in the account. The failover pair can use the aggregated license for 30 days, but if it is still non-compliant after the grace period, and you are not using the Strong Encryption token, then you will not be able to make configuration changes to features requiring the Strong Encryption (3DES/AES) feature license; operation is otherwise unaffected. The new active unit sends an entitlement authorization renewal request every 35 seconds until the license is compliant. If you disband the failover pair, then the active unit releases the entitlements, and both units retain the licensing configuration in a cached state. To re-activate licensing, you need to clear the configuration on each unit, and re-configure it.

Permanent License Reservation

For permanent license reservation, you must purchase separate licenses for each chassis and enable the licenses *before* you configure failover.

Failover Licenses for the Firepower 1100

Smart Software Manager Regular and On-Prem

Only the active unit requests licenses from the server. Licenses are aggregated into a single failover license that is shared by the failover pair. There is no extra cost for secondary units.

After you enable failover for Active/Standby failover, you can only configure smart licensing on the active unit. For Active/Active failover, you can only configure smart licensing on the unit with failover group 1 as active. The configuration is replicated to the standby unit, but the standby unit does not use the configuration; it remains in a cached state. The aggregated license is also cached on the standby unit to be used if it becomes the active unit in the future.

Note Each ASA must have the same encryption license when forming a failover pair. When you register an ASA to the smart licensing server, the Strong Encryption license is automatically enabled for qualified customers when you apply the registration token. Because of this requirement, you have two choices for licensing when you use the Strong Encryption token with failover:

- Before you enable failover, register both units to the smart licensing server. In this case, both units will have strong encryption. Then, after you enable failover, continue configuring license entitlements on the active unit. If you enable encryption for the failover link, it will use AES/3DES (strong encryption).
- Before you register the active unit to the smart licensing server, enable failover. In this case, both units will not yet have strong encryption. Then configure license entitlements and register the active unit to the smart licensing server; both units will get strong encryption from the aggregated license. Note that if you enabled encryption on the failover link, it will use DES (weak encryption) because the failover link was established before the units gained strong encryption. You must reload *both* units to use AES/3DES on the link. If you only reload one unit, then that unit will try to use AES/3DES while the original unit uses DES, which will result in both units becoming active (split brain).

Each add-on license type is managed as follows:

- Standard—Although only the active unit requests this license from the server, the standby unit has the Standard license enabled by default; it does not need to register with the server to use it.
- Context—Only the active unit requests this license. However, the Standard license includes 2 contexts by default and is present on both units. The value from each unit's Standard license plus the value of the Context license on the active unit are combined up to the platform limit. For example:
  - Active/Standby: The Standard license includes 2 contexts; for two Firepower 1120 units, these licenses add up to 4 contexts. You configure a 3-Context license on the active unit in an Active/Standby pair. Therefore, the aggregated failover license includes 7 contexts. However, because the platform limit for one unit is 5, the combined license allows a maximum of 5 contexts only. In this case, you might only configure the active Context license to be 1 context.
  - Active/Active: The Standard license includes 2 contexts; for two Firepower 1140 units, these licenses add up to 4 contexts. You configure a 4-Context license on the primary unit in an Active/Active pair. Therefore, the aggregated failover license includes 8 contexts. One unit can use 5 contexts and the other unit can use 3 contexts, for example; but during a failure, one unit will use all 8. Because the platform limit for one unit is 10, the combined license allows a maximum of 10 contexts; the 8 contexts are within the limit.
- Strong Encryption (3DES/AES)—If your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. Only the active unit requests this license, and both units can use it due to license aggregation.

After a failover, the new active unit continues to use the aggregated license. It uses the cached license configuration to re-request the entitlement from the server. When the old active unit rejoins the pair as a standby unit, it releases the license entitlement. Before the standby unit releases the entitlement, the new active unit's license might be in a non-compliant state if there are no available licenses in the account. The failover pair can use the aggregated license for 30 days, but if it is still non-compliant after the grace period, you will not be able to make configuration changes to features requiring special licenses (i.e. add an extra context); operation is otherwise unaffected. The new active unit sends an entitlement authorization renewal request

every 35 seconds until the license is compliant. If you disband the failover pair, then the active unit releases the entitlements, and both units retain the licensing configuration in a cached state. To re-activate licensing, you need to clear the configuration on each unit, and re-configure it.

Permanent License Reservation

For permanent license reservation, you must purchase separate licenses for each chassis and enable the licenses *before* you configure failover.

Failover Licenses for the Firepower 2100

Smart Software Manager Regular and On-Prem

Only the active unit requests licenses from the server. Licenses are aggregated into a single failover license that is shared by the failover pair. There is no extra cost for secondary units.

After you enable failover for Active/Standby failover, you can only configure smart licensing on the active unit. For Active/Active failover, you can only configure smart licensing on the unit with failover group 1 as active. The configuration is replicated to the standby unit, but the standby unit does not use the configuration; it remains in a cached state. The aggregated license is also cached on the standby unit to be used if it becomes the active unit in the future.

Note Each ASA must have the same encryption license when forming a failover pair. When you register an ASA to the smart licensing server, the Strong Encryption license is automatically enabled for qualified customers when you apply the registration token. Because of this requirement, you have two choices for licensing when you use the Strong Encryption token with failover:

- Before you enable failover, register both units to the smart licensing server. In this case, both units will have strong encryption. Then, after you enable failover, continue configuring license entitlements on the active unit. If you enable encryption for the failover link, it will use AES/3DES (strong encryption).
- Before you register the active unit to the smart licensing server, enable failover. In this case, both units will not yet have strong encryption. Then configure license entitlements and register the active unit to the smart licensing server; both units will get strong encryption from the aggregated license. Note that if you enabled encryption on the failover link, it will use DES (weak encryption) because the failover link was established before the units gained strong encryption. You must reload *both* units to use AES/3DES on the link. If you only reload one unit, then that unit will try to use AES/3DES while the original unit uses DES, which will result in both units becoming active (split brain).

Each add-on license type is managed as follows:

- Standard—Although only the active unit requests this license from the server, the standby unit has the Standard license enabled by default; it does not need to register with the server to use it.
- Context—Only the active unit requests this license. However, the Standard license includes 2 contexts by default and is present on both units. The value from each unit's Standard license plus the value of the Context license on the active unit are combined up to the platform limit. For example:
  - Active/Standby: The Standard license includes 2 contexts; for two Firepower 2130 units, these licenses add up to 4 contexts. You configure a 30-Context license on the active unit in an Active/Standby pair. Therefore, the aggregated failover license includes 34 contexts. However, because the platform limit for one unit is 30, the combined license allows a maximum of 30 contexts only. In this case, you might only configure the active Context license to be 25 contexts.

- Active/Active: The Standard license includes 2 contexts; for two Firepower 2130 units, these licenses add up to 4 contexts. You configure a 10-Context license on the primary unit in an Active/Active pair. Therefore, the aggregated failover license includes 14 contexts. One unit can use 9 contexts and the other unit can use 5 contexts, for example; but during a failure, one unit will use all 14. Because the platform limit for one unit is 30, the combined license allows a maximum of 30 contexts; the 14 contexts are within the limit.

- Strong Encryption (3DES/AES)—If your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. Only the active unit requests this license, and both units can use it due to license aggregation.

After a failover, the new active unit continues to use the aggregated license. It uses the cached license configuration to re-request the entitlement from the server. When the old active unit rejoins the pair as a standby unit, it releases the license entitlement. Before the standby unit releases the entitlement, the new active unit's license might be in a non-compliant state if there are no available licenses in the account. The failover pair can use the aggregated license for 30 days, but if it is still non-compliant after the grace period, you will not be able to make configuration changes to features requiring special licenses (i.e. add an extra context); operation is otherwise unaffected. The new active unit sends an entitlement authorization renewal request every 35 seconds until the license is compliant. If you disband the failover pair, then the active unit releases the entitlements, and both units retain the licensing configuration in a cached state. To re-activate licensing, you need to clear the configuration on each unit, and re-configure it.

Permanent License Reservation

For permanent license reservation, you must purchase separate licenses for each chassis and enable the licenses *before* you configure failover.

Failover Licenses for the Secure Firewall 3100

Smart Software Manager Regular and On-Prem

Each unit requires the Standard license (enabled by default) and the same encryption license. We recommend licensing each unit with the licensing server *before* you enable failover to avoid licensing mismatch issues, and when using the Strong Encryption license, issues with failover link encryption.

The failover feature itself does not require any licenses. There is no extra cost for the Context license on data units.

The Strong Encryption license is automatically enabled for qualified customers when you apply the registration token. For the optional Strong Encryption (3DES/AES) feature license enabled in the ASA configuration, see below.

In the ASA license configuration, the Standard license is always enabled by default on both units. After you enable failover for Active/Standby failover, you can only configure smart licensing on the active unit. For Active/Active failover, you can only configure smart licensing on the unit with failover group 1 as active. The configuration is replicated to the standby unit, but the standby unit does not use the configuration; it remains in a cached state. The aggregated license is also cached on the standby unit to be used if it becomes the active unit in the future.

Each add-on license type is managed as follows:

- Standard—Each unit requests a Standard license from the server.

- Context—Only the active unit requests this license. However, the Standard license includes 2 contexts by default and is present on both units. The value from each unit's Standard license plus the value of the Context license on the active unit are combined up to the platform limit. For example:
  - Active/Standby: The Standard license includes 2 contexts; for two Secure Firewall 3130 units, these licenses add up to 4 contexts. You configure a 100-Context license on the active unit in an Active/Standby pair. Therefore, the aggregated failover license includes 104 contexts. However, because the platform limit for one unit is 100, the combined license allows a maximum of 100 contexts only. In this case, you might only configure the active Context license to be 95 contexts.
  - Active/Active: The Standard license includes 2 contexts; for two Secure Firewall 3130 units, these licenses add up to 4 contexts. You configure a 10-Context license on the primary unit in an Active/Active pair. Therefore, the aggregated failover license includes 14 contexts. One unit can use 9 contexts and the other unit can use 5 contexts, for example; but during a failure, one unit will use all 14. Because the platform limit for one unit is 100, the combined license allows a maximum of 100 contexts; the 14 contexts are within the limit.
- Strong Encryption (3DES/AES)—If your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. Only the active unit requests this license, and both units can use it due to license aggregation.

After a failover, the new active unit continues to use the aggregated license. It uses the cached license configuration to re-request the entitlement from the server. When the old active unit rejoins the pair as a standby unit, it releases the license entitlement. Before the standby unit releases the entitlement, the new active unit's license might be in a non-compliant state if there are no available licenses in the account. The failover pair can use the aggregated license for 30 days, but if it is still non-compliant after the grace period, you will not be able to make configuration changes to features requiring special licenses (i.e. add an extra context); operation is otherwise unaffected. The new active unit sends an entitlement authorization renewal request every 35 seconds until the license is compliant. If you disband the failover pair, then the active unit releases the entitlements, and both units retain the licensing configuration in a cached state. To re-activate licensing, you need to clear the configuration on each unit, and re-configure it.

Permanent License Reservation

For permanent license reservation, you must purchase separate licenses for each chassis and enable the licenses *before* you configure failover.

Failover Licenses for the Firepower 4100/9300

Smart Software Manager Regular and On-Prem

Both Firepower 4100/9300 must be registered with the Smart Software Manager or Smart Software Manager On-Prem server before you configure failover. There is no extra cost for secondary units.

The Strong Encryption license is automatically enabled for qualified customers when you apply the registration token. When using the token, each chassis must have the same encryption license. For the optional Strong Encryption (3DES/AES) feature license enabled in the ASA configuration, see below.

After you enable failover, for the ASA license configuration for Active/Standby failover, you can only configure smart licensing on the active unit. For Active/Active failover, you can only configure smart licensing on the unit with failover group 1 as active. The configuration is replicated to the standby unit, but the standby unit does not use the configuration; it remains in a cached state. Only the active unit requests the licenses from the server. The licenses are aggregated into a single failover license that is shared by the failover pair, and

this aggregated license is also cached on the standby unit to be used if it becomes the active unit in the future. Each license type is managed as follows:

- Standard—Although only the active unit requests this license from the server, the standby unit has the Standard license enabled by default; it does not need to register with the server to use it.
- Context—Only the active unit requests this license. However, the Standard license includes 10 contexts by default and is present on both units. The value from each unit's Standard license plus the value of the Context license on the active unit are combined up to the platform limit. For example:
  - Active/Standby: The Standard license includes 10 contexts; for 2 units, these licenses add up to 20 contexts. You configure a 250-Context license on the active unit in an Active/Standby pair. Therefore, the aggregated failover license includes 270 contexts. However, because the platform limit for one unit is 250, the combined license allows a maximum of 250 contexts only. In this case, you should only configure the active Context license to be 230 contexts.
  - Active/Active: The Standard license includes 10 contexts; for 2 units, these licenses add up to 20 contexts. You configure a 10-Context license on the primary unit in an Active/Active pair. Therefore, the aggregated failover license includes 30 contexts. One unit can use 17 contexts and the other unit can use 13 contexts, for example; but during a failure, one unit will use all 30. Because the platform limit for one unit is 250, the combined license allows a maximum of 250 contexts; the 30 contexts are within the limit.
- Carrier—Only the active requests this license, and both units can use it due to license aggregation.
- Strong Encryption (3DES)—If your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. Only the active unit requests this license, and both units can use it due to license aggregation.

After a failover, the new active unit continues to use the aggregated license. It uses the cached license configuration to re-request the entitlement from the server. When the old active unit rejoins the pair as a standby unit, it releases the license entitlement. Before the standby unit releases the entitlement, the new active unit's license might be in a non-compliant state if there are no available licenses in the account. The failover pair can use the aggregated license for 30 days, but if it is still non-compliant after the grace period, you will not be able to make configuration changes to features requiring special licenses; operation is otherwise unaffected. The new active unit sends an entitlement authorization renewal request every 35 seconds until the license is compliant. If you disband the failover pair, then the active unit releases the entitlements, and both units retain the licensing configuration in a cached state. To re-activate licensing, you need to clear the configuration on each unit, and re-configure it.

Permanent License Reservation

For permanent license reservation, you must purchase separate licenses for each chassis and enable the licenses *before* you configure failover.

ASA Cluster Licenses for the Secure Firewall 3100

Smart Software Manager Regular and On-Prem

Each unit requires the Standard license (enabled by default) and the same encryption license. We recommend licensing each unit with the licensing server *before* you enable clustering to avoid licensing mismatch issues, and when using the Strong Encryption license, issues with cluster control link encryption.

The clustering feature itself does not require any licenses. There is no extra cost for the Context license on data units.

The Strong Encryption license is automatically enabled for qualified customers when you apply the registration token. For the optional Strong Encryption (3DES/AES) feature license enabled in the ASA configuration, see below.

In the ASA license configuration, the Standard license is always enabled by default on all units. You can only configure smart licensing on the control unit. The configuration is replicated to the data units, but for some licenses, they do not use the configuration; it remains in a cached state, and only the control unit requests the license. The licenses are aggregated into a single cluster license that is shared by the cluster units, and this aggregated license is also cached on the data units to be used if one of them becomes the control unit in the future. Each license type is managed as follows:

- Standard—Each unit requests a Standard license from the server.
- Context—Only the control unit requests the Context license from the server. The Standard license includes 2 contexts by default and is present on all cluster members. The value from each unit's Standard license plus the value of the Context license on the control unit are combined up to the platform limit in an aggregated cluster license. For example:
  - You have 6 Secure Firewall 3100s in the cluster. The Standard license includes 2 contexts; for 6 units, these licenses add up to 12 contexts. You configure an additional 20-Context license on the control unit. Therefore, the aggregated cluster license includes 32 contexts. Because the platform limit for one chassis is 100, the combined license allows a maximum of 100 contexts; the 32 contexts are within the limit. Therefore, you can configure up to 32 contexts on the control unit; each data unit will also have 32 contexts through configuration replication.
  - You have 3 Secure Firewall 3100s in the cluster. The Standard license includes 2 contexts; for 3 units, these licenses add up to 6 contexts. You configure an additional 100-Context license on the control unit. Therefore, the aggregated cluster license includes 106 contexts. Because the platform limit for one unit is 100, the combined license allows a maximum of 100 contexts; the 106 contexts are over the limit. Therefore, you can only configure up to 100 contexts on the control unit; each data unit will also have 100 contexts through configuration replication. In this case, you should only configure the control unit Context license to be 94 contexts.
- Strong Encryption (3DES)—If your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. Only the control unit requests this license, and all units can use it due to license aggregation.

If a new control unit is elected, the new control unit continues to use the aggregated license. It also uses the cached license configuration to re-request the control unit license. When the old control unit rejoins the cluster as a data unit, it releases the control unit license entitlement. Before the data unit releases the license, the control unit's license might be in a non-compliant state if there are no available licenses in the account. The retained license is valid for 30 days, but if it is still non-compliant after the grace period, you will not be able to make configuration changes to features requiring special licenses; operation is otherwise unaffected. The new active unit sends an entitlement authorization renewal request every 35 seconds until the license is compliant. You should refrain from making configuration changes until the license requests are completely processed. If a unit leaves the cluster, the cached control configuration is removed, while the per-unit entitlements are retained. In particular, you would need to re-request the Context license on non-cluster units.

Permanent License Reservation

For permanent license reservation, you must purchase separate licenses for each chassis and enable the licenses *before* you configure clustering.

ASA Cluster Licenses for the ASAv

Smart Software Manager Regular and On-Prem

Each unit requires the same Throughput license and the same encryption license. We recommend licensing each unit with the licensing server *before* you enable clustering to avoid licensing mismatch issues, and when using the Strong Encryption license, issues with cluster control link encryption.

The clustering feature itself does not require any licenses.

The Strong Encryption license is automatically enabled for qualified customers when you apply the registration token. For the optional Strong Encryption (3DES/AES) feature license enabled in the ASA configuration, see below.

In the ASA license configuration, you can only configure smart licensing on the control unit. The configuration is replicated to the data units, but for some licenses, they do not use the configuration; it remains in a cached state, and only the control unit requests the license. The licenses are aggregated into a single cluster license that is shared by the cluster units, and this aggregated license is also cached on the data units to be used if one of them becomes the control unit in the future. Each license type is managed as follows:

- Standard—Only the control unit requests the Standard license from the server, and all units can use it due to license aggregation.
- Throughput—Each unit requests its own Throughput license from the server.
- Strong Encryption (3DES)—If your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. Only the control unit requests this license, and all units can use it due to license aggregation.

Permanent License Reservation

For permanent license reservation, you must purchase separate licenses for each unit and enable the licenses *before* you configure clustering.

ASA Cluster Licenses for the Firepower 4100/9300

Smart Software Manager Regular and On-Prem

The clustering feature itself does not require any licenses. To use Strong Encryption and other optional licenses, each Firepower 4100/9300 chassis must be registered with the License Authority or Smart Software Manager Regular and On-Prem server. There is no extra cost for data units.

The Strong Encryption license is automatically enabled for qualified customers when you apply the registration token. When using the token, each chassis must have the same encryption license. For the optional Strong Encryption (3DES/AES) feature license enabled in the ASA configuration, see below.

In the ASA license configuration, you can only configure smart licensing on the control unit. The configuration is replicated to the data units, but for some licenses, they do not use the configuration; it remains in a cached state, and only the control unit requests the license. The licenses are aggregated into a single cluster license that is shared by the cluster units, and this aggregated license is also cached on the data units to be used if one of them becomes the control unit in the future. Each license type is managed as follows:

- Standard—Only the control unit requests the Standard license from the server, and both units can use it due to license aggregation.
- Context—Only the control unit requests the Context license from the server. The Standard license includes 10 contexts by default and is present on all cluster members. The value from each unit's Standard license plus the value of the Context license on the control unit are combined up to the platform limit in an aggregated cluster license. For example:
  - You have 6 Firepower 9300 modules in the cluster. The Standard license includes 10 contexts; for 6 units, these licenses add up to 60 contexts. You configure an additional 20-Context license on the control unit. Therefore, the aggregated cluster license includes 80 contexts. Because the platform limit for one module is 250, the combined license allows a maximum of 250 contexts; the 80 contexts are within the limit. Therefore, you can configure up to 80 contexts on the control unit; each data unit will also have 80 contexts through configuration replication.
  - You have 3 Firepower 4112 units in the cluster. The Standard license includes 10 contexts; for 3 units, these licenses add up to 30 contexts. You configure an additional 250-Context license on the control unit. Therefore, the aggregated cluster license includes 280 contexts. Because the platform limit for one unit is 250, the combined license allows a maximum of 250 contexts; the 280 contexts are over the limit. Therefore, you can only configure up to 250 contexts on the control unit; each data unit will also have 250 contexts through configuration replication. In this case, you should only configure the control unit Context license to be 220 contexts.
- Carrier—Required for Distributed S2S VPN. This license is a per-unit entitlement, and each unit requests its own license from the server.
- Strong Encryption (3DES)—For pre-2.3.0 Cisco Smart Software Manager On-Prem deployment; or if your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. This license is a per-unit entitlement, and each unit requests its own license from the server.

If a new control unit is elected, the new control unit continues to use the aggregated license. It also uses the cached license configuration to re-request the control unit license. When the old control unit rejoins the cluster as a data unit, it releases the control unit license entitlement. Before the data unit releases the license, the control unit's license might be in a non-compliant state if there are no available licenses in the account. The retained license is valid for 30 days, but if it is still non-compliant after the grace period, you will not be able to make configuration changes to features requiring special licenses; operation is otherwise unaffected. The new active unit sends an entitlement authorization renewal request every 12 hours until the license is compliant. You should refrain from making configuration changes until the license requests are completely processed. If a unit leaves the cluster, the cached control configuration is removed, while the per-unit entitlements are retained. In particular, you would need to re-request the Context license on non-cluster units.

Permanent License Reservation

For permanent license reservation, you must purchase separate licenses for each chassis and enable the licenses *before* you configure clustering.

Prerequisites for Smart Software Licensing

Smart Software Manager Regular and On-Prem Prerequisites

Firepower 4100/9300

Configure the Smart Software Licensing infrastructure on the Firepower 4100/9300 chassis before you configure the ASA licensing entitlements.

All Other Models

- Ensure internet access, or HTTP proxy access, or Smart Software Manager On-Prem server access from the device.
- Configure a DNS server so the device can resolve the name of the Smart Software Manager.
- Set the clock for the device. On the Firepower 2100 in Platform mode, you set the clock in FXOS.
- Create an account on the Cisco Smart Software Manager:

  https://software.cisco.com/#module/SmartLicensing

  If you do not yet have an account, click the link to set up a new account. The Smart Software Manager lets you create an account for your organization.

Permanent License Reservation Prerequisites

- Create a master account on the Cisco Smart Software Manager:

  https://software.cisco.com/#module/SmartLicensing

  If you do not yet have an account, click the link to set up a new account. The Smart Software Manager lets you create a master account for your organization. Even though the ASA does need internet connectivity to the Smart Licensing server for permanent license reservation, the Smart Software Manager is used to manage your permanent licenses.
- Obtain support for permanent license reservation from the licensing team. You must provide a justification for using permanent license reservation. If your account is not approved, then you cannot purchase and apply permanent licenses.
- Purchase special permanent licenses. If you do not have the correct license in your account, then when you try to reserve a license on the ASA, you will see an error message similar to: "The licenses cannot be reserved because the Virtual Account does not contain a sufficient surplus of the following perpetual licenses: 1 - Firepower 4100 ASA PERM UNIV(perpetual)."
- The permanent license includes all available features, including the Strong Encryption (3DES/AES) license if your account qualifies. The AnyConnect Client capabilities are also enabled to the platform maximum, contingent on your purchase of an AnyConnect Client license that enables the right to use AnyConnect Client (see AnyConnect Plus, AnyConnect Apex, and AnyConnect VPN Only Licenses, on page 94).
- ASAv: Permanent license reservation is not supported for the Azure hypervisor.

Guidelines for Smart Software Licensing

- Only Smart Software Licensing is supported. For older software on the ASAv, if you upgrade an existing PAK-licensed ASAv, then the previously installed activation key will be ignored, but retained on the device. If you downgrade the ASAv, the activation key will be reinstated.
- For permanent license reservation, you must return the license before you decommission the device. If you do not officially return the license, the license remains in a used state and cannot be reused for a new device.
- Because the Cisco Transport Gateway uses a certificate with a non-compliant country code, you cannot use HTTPS when using the ASA in conjunction with that product. You must use HTTP with Cisco Transport Gateway.

Defaults for Smart Software Licensing

Smart Call Home Profile

The default configuration for all models except the Firepower 4100/9300 (which enables Smart Software License communication at the chassis-level) includes a Smart Call Home profile called "License" that specifies the URL for the Smart Software Manager.

```
call-home
  profile License
   destination address http https://tools.cisco.com/its/service/oddce/services/DDCEService
```

ASAv

- When you deploy the ASAv, you set the feature tier and throughput level. Only the Standard level is available at this time. For permanent license reservation, you do not need to set these parameters. When you enable permanent license reservation, these commands are removed from the configuration.

```
license smart
  feature tier standard
  throughput level {100M | 1G | 2G | 10G | 20G}
```

ASAv: Configure Smart Software Licensing

This section describes how to configure Smart Software Licensing for ASAv.

ASAv: Configure Regular Smart Software Licensing

When you deploy ASAv, you can pre-configure the device and include a registration token so it registers with the Smart Software Manager and enables Smart Software Licensing. If you need to change your HTTP proxy

server, license entitlement, or register the ASAv (for example, if you did not include the ID token in the Day0 configuration), perform this task.

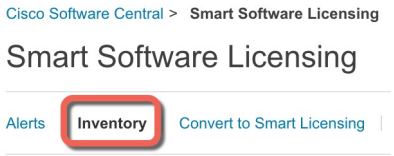
Note You may have pre-configured the HTTP proxy and license entitlements when you deployed your ASAv. You may also have included the registration token with your Day0 configuration when you deployed the ASAv; if so, you do not need to re-register using this procedure.

Procedure

Step 1 (Cisco Smart Software Manager), request and copy a registration token for the virtual account to which you want to add this device.

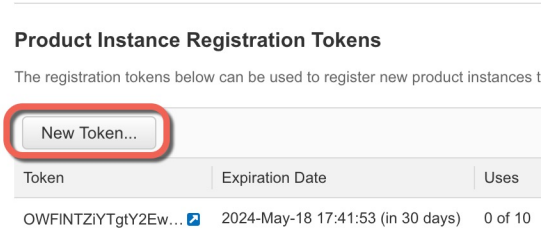
a) Click **Inventory**.

Figure 7: Inventory



b) On the **General** tab, click **New Token**.

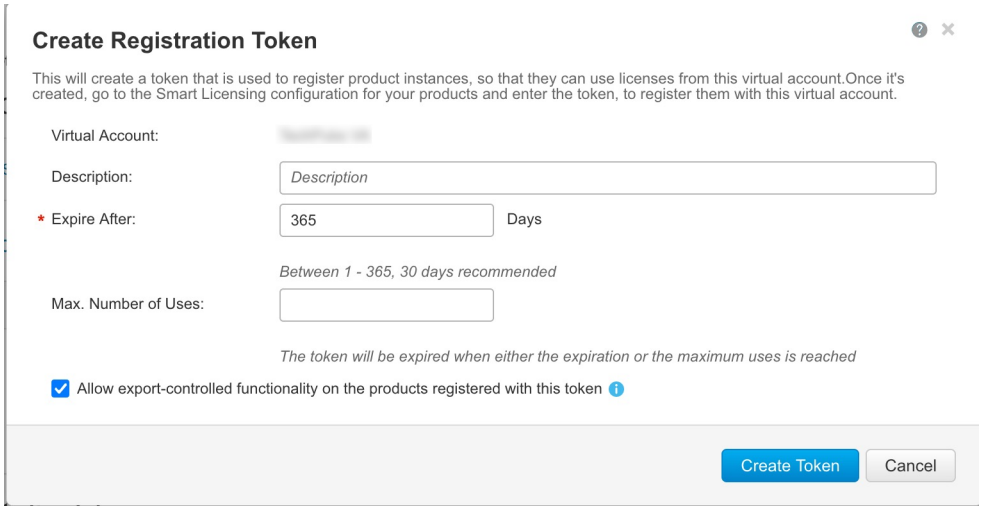
Figure 8: New Token



c) On the **Create Registration Token** dialog box enter the following settings, and then click **Create Token**:

- **Description**
- **Expire After**—Cisco recommends 30 days.
- **Max. Number of Uses**
- **Allow export-controlled functionaility on the products registered with this token**—Enables the export-compliance flag.

Figure 9: Create Registration Token



The token is added to your inventory.

d) Click the arrow icon to the right of the token to open the **Token** dialog box so you can copy the token ID to your clipboard. Keep this token ready for later in the procedure when you need to register the ASA.

Figure 10: View Token

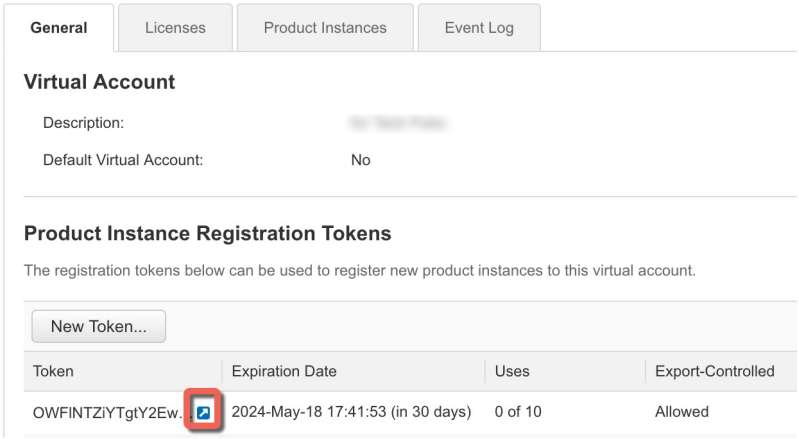
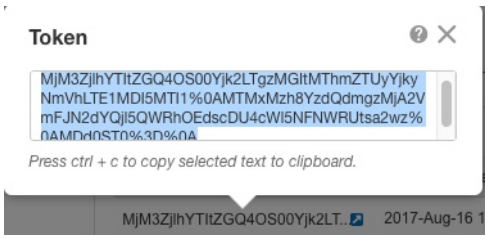


Figure 11: Copy Token



Step 2 (Optional) On ASAv, specify the HTTP Proxy URL for Call Home:

call-home

http-proxy *ip_address* **port** *port*

If your network uses an HTTP proxy for internet access, you must configure the proxy address for Smart Software Licensing. This proxy is also used for Smart Transport and Smart Call Home in general.

Note
HTTP proxy with authentication is not supported.

Example:

```
ciscoasa(config)# call-home
ciscoasa(cfg-call-home)# http-proxy 10.1.1.1 port 443
```

Step 3 Configure the license entitlements:

a) Enter license smart configuration mode:

license smart

Example:

```
ciscoasa(config)# license smart
ciscoasa(config-smart-lic)#
```

b) Set the feature tier:

feature tier standard

Only the standard tier is available, however, you need to enable it in the configuration.

c) Set the throughput level to determine the license requested from the Smart Software Manager:

throughput level {**100M** | **1G** | **2G** | **10G** | **20G**}

See the following throughput/license relationship:

- 100M—ASAv5
- 1G—ASAv10
- 2G—ASAv30
- 10G—ASAv50
- 20G—ASAv100

Example:

```
ciscoasa(config-smart-lic)# throughput level 2G
```

d) Exit license smart mode to apply your changes:

exit

Your changes do not take effect until you exit the license smart configuration mode, either by explicitly exiting the mode (**exit** or **end**) or by entering any command that takes you to a different mode.

Example:

```
ciscoasa(config-smart-lic)# exit
ciscoasa(config)#
```

Step 4 Register the ASAv with the Smart Software Manager.

license smart register idtoken *id_token* [**force**]

The ASAv attempts to register with the Smart Software Manager and request authorization for the configured license entitlements.

When you register the ASAv, the Smart Software Manager issues an ID certificate for communication between the ASAv and the Smart Software Manager. It also assigns the ASAv to the appropriate virtual account. Normally, this procedure is a one-time instance. However, you might need to later re-register the ASAv if the ID certificate expires because of a communication problem, for example.

Use the **force** keyword to register the ASAv that is already registered, but that might be out of sync with the Smart Software Manager. For example, use **force** if the ASAv was accidentally removed from the Smart Software Manager.

Example:

```
ciscoasa# license smart register idtoken YjE3Njc5MzYtMGQzMi00OTA4
LWJhODItNzBhMGQ5NGRlYjUxLTE0MTQ5NDAy%0AODQzNzl8NXk2bzV3SDE0ZkgwQk
dYRmZ1NTNCNGlvRnBHUFpjcm02WTB4TU4w%0Ac2NnMD0%3D%0A
```

Step 5 Check the license status.

show license status or **show running-config license**

If communication with the Smart Software Manager fails, check that you have a DNS server configured as well as correct routing to reach the server.

Example:

```
asav1# show license status

Smart Licensing is ENABLED

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
    Callhome hostname privacy: DISABLED
    Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

Registration:
  Status: REGISTERING - REGISTRATION IN PROGRESS
  Export-Controlled Functionality: NOT ALLOWED
  Initial Registration: FAILED on Aug 31 2023 19:38:45 UTC
    Failure reason: Communication message send error
  Next Registration Attempt: Aug 31 2023 19:56:56 UTC

License Authorization:
  Status: EVAL EXPIRED on Feb 25 2023 16:39:25 UTC
```

```
Export Authorization Key:
  Features Authorized:
    <none>

Miscellaneus:
  Custom Id: <empty>
```

ASAv: Configure Smart Software Manager On-Prem Licensing

This procedure applies for the ASAv using a Smart Software Manager On-Prem.

Before you begin

- Download the Smart Software Manager On-Prem OVA file from Cisco.com and install and configure it on a VMwareESXi server. For more information, see the *Cisco Smart Software Manager On-Prem Data Sheet*.
- Download the crypto CA trustpool before you place the device in an air-gapped network. This trustpool is normally downloaded automatically, but may be out of date in an air-gapped network:

  crypto ca trustpool import url http://www.cisco.com/security/pki/trs/ios_core.p7b

Procedure

Step 1 Request a registration token on the Smart Software Manager On-Prem.

Step 2 (Optional) On the ASAv, specify the HTTP Proxy URL for Call Home:

call-home

http-proxy *ip_address* **port** *port*

Note
HTTP proxy with authentication is not supported.

Example:

```
ciscoasa(config)# call-home
ciscoasa(cfg-call-home)# http-proxy 10.1.1.1 port 443
```

Step 3 Change the license server URL to go to the Smart Software Manager On-Prem.

call-home

profile License

destination address http https://*on-prem_ip_address***/Transportgateway/services/DeviceRequestHandler**

Example:

```
ciscoasa(config)# call-home
```

```
ciscoasa(cfg-call-home)# profile License
ciscoasa(cfg-call-home-profile) destination address http
https://10.1.5.5/Transportgateway/services/DeviceRequestHandler
```

Step 4 Configure the license entitlements.

a) Enter license smart configuration mode:

license smart

Example:

```
ciscoasa(config)# license smart
ciscoasa(config-smart-lic)#
```

b) Set the feature tier:

feature tier standard

Only the standard tier is available, however, you need to enable it in the configuration.

c) Set the throughput level to determine the license requested from the Smart Software Manager:

throughput level {**100M** | **1G** | **2G** | **10G** | **20G**}

See the following throughput/license relationship:

- 100M—ASAv5
- 1G—ASAv10
- 2G—ASAv30
- 10G—ASAv50
- 20G—ASAv100

Example:

```
ciscoasa(config-smart-lic)# throughput level 2G
```

d) Exit license smart mode to apply your changes:

exit

Your changes do not take effect until you exit the license smart configuration mode, either by explicitly exiting the mode (**exit** or **end**) or by entering any command that takes you to a different mode.

Example:

```
ciscoasa(config-smart-lic)# exit
ciscoasa(config)#
```

Step 5 Register the ASA using the token you requested in Step 1:

license smart register idtoken *id_token*

The ASA registers with the Smart Software Manager On-Prem and requests authorization for the configured license entitlements. The Smart Software Manager On-Prem also applies the Strong Encryption (3DES/AES) license if your account allows.

When you register the ASAv, the Smart Software Manager On-Prem issues an ID certificate for communication between the ASAv and the Smart Software Manager. It also assigns the ASAv to the appropriate virtual account. Normally, this procedure is a one-time instance. However, you might need to later re-register the ASAv if the ID certificate expires because of a communication problem, for example.

Example:

```
ciscoasa# license smart register idtoken YjE3Njc5MzYtMGQzMi00OTA4
LWJhODItNzBhMGQ5NGRlYjUxLTE0MTQ5NDAy%0AODQzNzl8NXk2bzV3SDE0ZkgwQk
dYRmZ1NTNCNGlvRnBHUFpjcm02WTB4TU4w%0Ac2NnMD0%3D%0A
```

Step 6 Check the license status.

show license status

If communication with the Smart Software Manager fails, check that you have a DNS server configured as well as correct routing to reach the server.

Example:

```
asav1# show license status

Smart Licensing is ENABLED

Utility:
  Status: DISABLED

Data Privacy:
  Sending Hostname: yes
    Callhome hostname privacy: DISABLED
    Smart Licensing hostname privacy: DISABLED
  Version privacy: DISABLED

Transport:
  Type: Callhome

Registration:
  Status: REGISTERING - REGISTRATION IN PROGRESS
  Export-Controlled Functionality: NOT ALLOWED
  Initial Registration: FAILED on Aug 31 2023 19:38:45 UTC
    Failure reason: Communication message send error
  Next Registration Attempt: Aug 31 2023 19:56:56 UTC

License Authorization:
  Status: EVAL EXPIRED on Feb 25 2023 16:39:25 UTC

Export Authorization Key:
  Features Authorized:
    <none>

Miscellaneus:
  Custom Id: <empty>
```

ASAv: Configure Utility (MSLA) Smart Software Licensing

Utility Licensing for a Managed Service License Agreement (MSLA) lets you pay for the amount of time a license is in use rather than paying a one time charge for a license subscription or a perpetual license. In Utility Licensing mode, the ASAv keeps track of license usage in units of time (15-minute intervals). The ASAv sends license usage reports (known as RUM reports) to the Smart Software Manager every four hours. The usage reports are then forwarded to a billing server. With Utility Licensing, Smart Call Home is not used as the transport for licensing messages. Instead the messages are sent directly via HTTP/HTTPS using *Smart Transport.*

Before you begin

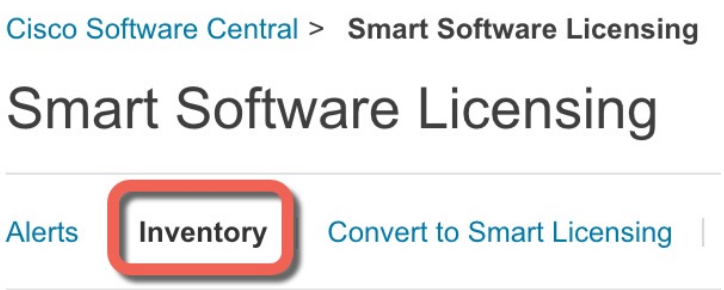
Download the Smart Software Manager On-Prem OVA file from Cisco.com and install and configure it on a VMware ESXi server. For more information, see the *Cisco Smart Software Manager On-Prem Data Sheet*.

Procedure

Step 1 In the Smart Software Manager On-Prem, request and copy a registration token for the virtual account to which you want to add this device.

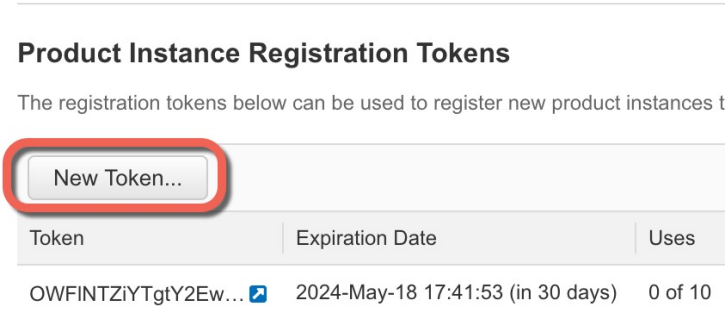
a) Click **Inventory**.

Figure 12: Inventory



b) On the **General** tab, click **New Token**.

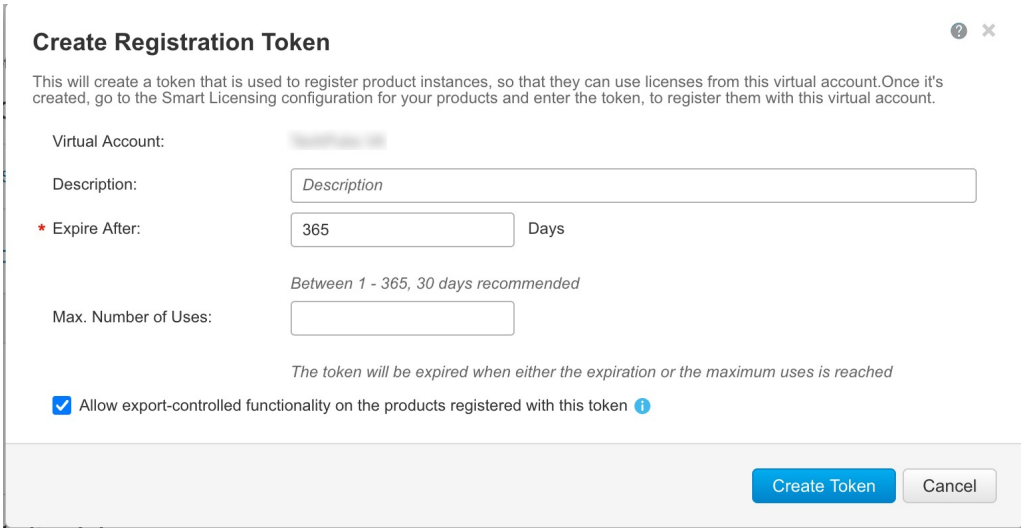
Figure 13: New Token



c) On the **Create Registration Token** dialog box enter the following settings, and then click **Create Token**:

- **Description**
- **Expire After**—Cisco recommends 30 days.
- **Allow export-controlled functionaility on the products registered with this token**—Enables the export-compliance flag.

Figure 14: Create Registration Token



The token is added to your inventory.

d) Click the arrow icon to the right of the token to open the **Token** dialog box so you can copy the token ID to your clipboard. Keep this token ready for later in the procedure when you need to register the ASA.

Figure 15: View Token

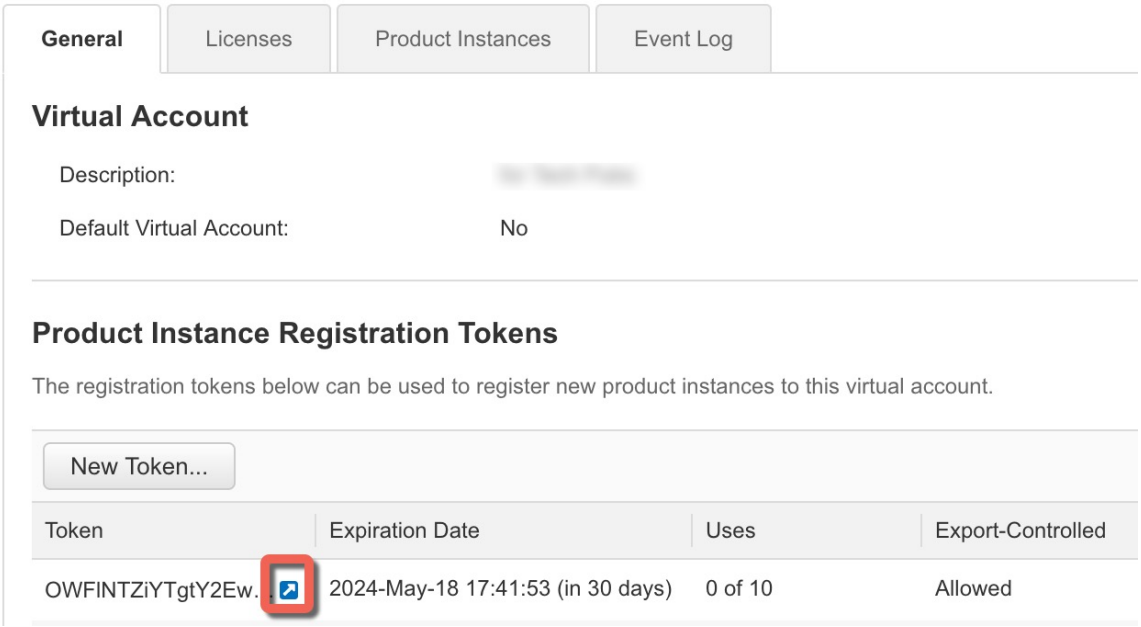
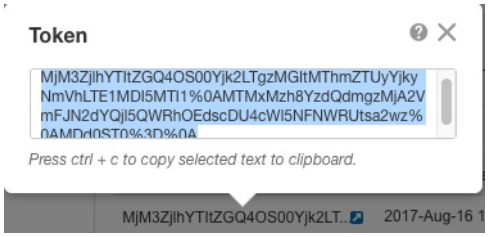


Figure 16: Copy Token



Step 2 On ASAv, configure Smart Licensing parameters.

a) Enter license smart configuration mode.

license smart

Example:

```
ciscoasa(config)# license smart
ciscoasa(config-smart-lic)#
```

b) Enable Smart Transport.

transport type smart

Utility Licensing requires Smart Transport instead of Smart Call Home.

Example:

```
ciscoasa(config-smart-lic)# transport type smart
```

c) (Optional) Specify the URL of the Smart Software Manager Regular or On-Prem. Optionally, you can specify an alternate destination for the license usage reports.

transport url {*transport_url* | **utility** *utility_url* | **default** }

If you do not set a URL, it uses the **default** for both Smart Transport and license usage reports, which is **https://smartreceiver.cisco.com/licservice/license**.

Example:

```
ciscoasa(config-smart-lic)# transport url http://server99.cisco.com/SmartTransport
ciscoasa(config-smart-lic)# transport url utility
http://server-utility.cisco.com/SmartTransport
```

d) (Optional) If your network uses an HTTP proxy for internet access, configure the proxy address.

transport proxy *proxy-url* **port** *proxy-port-number*

Note
HTTP proxy with authentication is not supported.

Example:

```
ciscoasa(config-smart-lic)# transport proxy 10.1.1.1 port 443
```

e) Suppress the licensing device's hostname or Smart Agent version number in the licensing messages.

privacy {**all** | **hostname** | **version**}

Example:

```
ciscoasa(config-smart-lic)# privacy all
```

f) Set the feature tier:

feature tier standard

Only the standard tier is available, however, you need to enable it in the configuration.

g) Set the throughput level to determine the license requested from the Smart Software Manager:

throughput level {**100M** | **1G** | **2G** | **10G** | **20G**}

See the following throughput/license relationship:

- 100M—ASAv5
- 1G—ASAv10
- 2G—ASAv30
- 10G—ASAv50
- 20G—ASAv100

Example:

```
ciscoasa(config-smart-lic)# throughput level 2G
```

h) (Optional) Enable strong encryption.

feature strong-encryption

This license is not required if you receive the strong encryption token from the Smart Software Manager. However, if your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. Only the active unit requests this license, and both units can use it due to license aggregation.

Example:

```
ciscoasa(config-smart-lic)# feature strong-encryption
```

i) Exit license smart mode to apply your changes:

exit

Your changes do not take effect until you exit the license smart configuration mode, either by explicitly exiting the mode (**exit** or **end**) or by entering any command that takes you to a different mode.

Example:

```
ciscoasa(config-smart-lic)# exit
ciscoasa(config)#
```

Step 3 Configure Utility Licensing.

a) Enter utility configuration mode.

utility

Example:

```
ciscoasa(config-smart-lic)# utility
ciscoasa(config-smart-lic-util)#
```

b) Create a unique customer identifier. This identifier is included in Utility Licensing usage report messages.

custom-id *custom-identifier*

Example:

```
ciscoasa(config-smart-lic-util)# custom-id MyCustomID
```

c) Create a unique customer profile. This information is included in Utility Licensing usage reports.

customer-info {**city** | **country** | **id** | **name** | **postalcode** | **state** |**street**} *value*

Example:

```
ciscoasa(config-smart-lic-util)# customer-info city MyCity
ciscoasa(config-smart-lic-util)# customer-info country MyCountry
ciscoasa(config-smart-lic-util)# customer-info id MyID
ciscoasa(config-smart-lic-util)# customer-info name MyName
ciscoasa(config-smart-lic-util)# customer-info postalcode MyPostalCode
ciscoasa(config-smart-lic-util)# customer-info state MyState
ciscoasa(config-smart-lic-util)# customer-info street MyStreet
```

d) Enable Utility Licensing.

mode standard

Example:

```
ciscoasa(config-smart-lic-util)# mode standard
```

Step 4 Register the ASA using the token you requested in Step 1:

license smart register idtoken *id_token*

Example:

```
ciscoasa# license smart register idtoken YjE3Njc5MzYtMGQzMi00OTA4
LWJhODItNzBhMGQ5NGRlYjUxLTE0MTQ5NDAy%0AODQzNzl8NXk2bzV3SDE0ZkgwQk
dYRmZ1NTNCNGlvRnBHUFpjcm02WTB4TU4w%0Ac2NnMD0%3D%0A
```

Step 5 Check the license status.

show license status

If communication with the Smart Software Manager fails, check that you have a DNS server configured as well as correct routing to reach the server.

Example:

```
asav1# show license status

Smart Licensing is ENABLED

Utility:
  Status: ENABLED
  Utility report:
    Last success: May 14 2018 21:37:25 UTC
    Last attempt: SUCCEEDED on May 14 2018 21:37:24 UTC
    Next attempt: May 15 2018 01:37:24 UTC

  Customer Information:
    Id: MyID
    Name: MyName
    Street: MyStreet
    City: MyCity
    State: MyState
    Country: MyCountry
    Postal Code: MyPostalCode

Data Privacy:
  Sending Hostname: no
    Callhome hostname privacy: DISABLED
    Smart Licensing hostname privacy: ENABLED
  Version privacy: ENABLED

Transport:
  Type: Smart
  Registration URL: http://server99.cisco.com/SmartTransport
  Utility URL: http://server-utility.cisco.com/SmartTransport

Registration:
  Status: REGISTERED
  Export-Controlled Functionality: Allowed
  Initial Registration: SUCCEEDED on May 14 2018 21:37:20 UTC
  Last Renewal Attempt: None
  Next Renewal Attempt: Sep 13 2018 13:34:40 UTC
  Registration Expires: May 14 2019 21:29:20 UTC

License Authorization:
  Status: AUTHORIZED on May 14 2018 21:37:22 UTC
  Last Communication Attempt: NOT STARTED
    Failure reason: Device in Thirdparty Utility Mode
  Next Communication Attempt: None
  Communication Deadline: Aug 12 2018 21:37:24 UTC
```

ASAv: Configure Permanent License Reservation

You can assign a permanent license to the ASAv. This section also describes how to return a license if you retire the ASAv or change model tiers and need a new license.

Procedure

Step 1 Install the ASAv Permanent License, on page 123

Step 2 (Optional) Return the ASAv Permanent License, on page 125

Install the ASAv Permanent License

For an ASAv that does not have internet access, you can request a permanent license from the Smart Software Manager. For more information about ASAv permanent license reservation, see ASA Virtual Permanent License Reservation, on page 89.



Note

- For permanent license reservation, you must return the license before you decommission the ASAv. If you do not officially return the license, the license remains in a used state and cannot be reused for a new ASAv. See (Optional) Return the ASAv Permanent License, on page 125.
- If you clear your configuration after you install the permanent license, for example, by using the **write erase** command, you only have to re-enable permanent license reservation using the **license smart reservation** command without any arguments, you do not need to complete the rest of this procedure.

Before you begin

- Purchase permanent licenses so that they are available in the Smart Software Manager. Not all accounts are approved for permanent license reservation. Make sure that you have approval from Cisco for this feature before you attempt to configure it.
- Request a permanent license after the ASAv starts up; you cannot install a permanent license as part of the Day 0 configuration.

Procedure

Step 1 From the ASAv CLI, enable permanent license reservation:

license smart reservation

Example:

```
ciscoasa (config)# license smart reservation
ciscoasa (config)#
```

To use regular smart licensing, use the **no** form of this command.

The following commands are removed:

```
license smart
  feature tier standard
```

```
throughput level {100M | 1G | 2G | 10G | 20G}
```

Step 2 Request the license code to enter in the Smart Software Manager:

license smart reservation request universal

Example:

```
ciscoasa# license smart reservation request universal
Enter this request code in the Cisco Smart Software Manager portal:
ABP:ASAv,S:9AU5ET6UQHD{A8ug5/1jRDaSp3w8uGlfeQ{53C13E
ciscoasa#
```

If you change the memory or vCPUs, you will have to return the current license and request a new license at the new model level. The license requested must match the installed memory and vCPUs. For more information about the vCPU and memory-to-license matrix for ASAv permanent license reservation, see ASA Virtual Permanent License Reservation, on page 89.

To change the model of an already deployed ASAv, from the hypervisor you can change the vCPUs and DRAM settings to match the new model requirements; see the ASAv quick start guide for these values. To view your current model, use the **show vm** command.

If you re-enter this command, then the same code is displayed, even after a reload. If you have not yet entered this code into the Smart Software Manager and want to cancel the request, enter:

license smart reservation cancel

If you disable permanent license reservation, then any pending requests are canceled. If you already entered the code into the Smart Software Manager, then you must complete this procedure to apply the license to the ASAv, after which point you can return the license if desired. See (Optional) Return the ASAv Permanent License, on page 125.

Step 3 Go to the Smart Software Manager Inventory screen, and click the **Licenses** tab:

https://software.cisco.com/#SmartLicensing-Inventory

The **Licenses** tab displays all existing licenses related to your account, both regular and permanent.

Step 4 Click **License Reservation**, and enter the ASAv code in the **Reservation Request Code** field.

Step 5 Click **Reserve License**.

The Smart Software Manager generates an authorization code. You can download the code or copy it to the clipboard. At this point, the license is now in use according to the Smart Software Manager.

If you do not see the **License Reservation** button, then your account is not authorized for permanent license reservation. In this case, you should disable permanent license reservation and re-enter the regular smart license commands.

Step 6 From the ASAv CLI, run the following command and enter the authorization code:

license smart reservation install *code*

Example:

```
ciscoasa# license smart reservation install AAu3431rGRS00Ig5HQl2vpzg{MEYCIQCBw$
INFO: ASAv platform license state is Licensed.
ciscoasa#
```

Your ASAv is now fully licensed.

Step 7 Run this command to save the running configuration to the startup configuration to avoid any loss of configuration during reboot.

write memory

Step 8 From the ASAv, run this command to display and confirm that the ASA has the permanent license.

show license features

(Optional) Return the ASAv Permanent License

If you no longer need a permanent license, for example, you are retiring the ASAv or changing its model level, because of which it needs a new license, you must officially return the license to the Smart Software Manager using this procedure. If you do not follow all the steps, the license stays in a used state and cannot be freed up for use elsewhere.

Procedure

Step 1 From the ASAv CLI, generate a return code:

license smart reservation return

Example:

```
ciscoasa# license smart reservation return
Enter this return code in the Cisco Smart Software Manager portal:
Au3431rGRS00Ig5HQl2vpcg{uXiTRfVrp7M/zDpirLwYCaq8oSv60yZJuFDVBS2QliQ=
```

The ASAv immediately becomes unlicensed and moves to the Evaluation state. If you need to view this code again, re-enter this command. Note that if you request a new permanent license (**license smart reservation request universal**) or change the ASAv model level (by powering down and changing the vCPUs/RAM), then you cannot re-display this code. Be sure to capture the code to complete the return.

Step 2 View the ASAv universal device identifier (UDI) so you can find this ASAv instance in the Smart Software Manager:

show license udi

Example:

```
ciscoasa# show license udi
UDI: PID:ASAv,SN:9AHV3KJBEKE
ciscoasa#
```

Step 3 Go to the Smart Software Manager Inventory screen, and click the **Product Instances** tab:

https://software.cisco.com/#SmartLicensing-Inventory

The **Product Instances** tab displays all licensed products by the UDI.

Step 4 Find the ASAv you want to unlicense, choose **Actions** > **Remove**, and enter the ASAv return code in the **Reservation Return Code** field.

The permanent license is returned to the available pool.

Step 5 Click **Remove Product Instance**.

Step 6 To disable permanent license reservation, do the following:

a) Run the following command to disable permanent license reservation:

no license smart reservation

b) Run the following command to confirm that the license is deregistered:

show license features

The command output shows `No active entitlement`.

Step 7 Run this command to save the running configuration to the startup configuration to avoid any loss of configuration during reboot.

write memory

(Optional) Deregister the ASAv (Regular and On-Prem)

Deregistering the ASAv removes the ASAv from your account. All license entitlements and certificates on the ASAv are removed. You might want to deregister to free up a license for a new ASAv. Alternatively, you can remove the ASAv from the Smart Software Manager.



Note If you deregister the ASAv, then it will revert to a severely rate-limited state after you reload the ASAv.

Procedure

Deregister the ASAv:

license smart deregister

The ASAv then reloads.

(Optional) Renew the ASAv ID Certificate or License Entitlement (Regular and On-Prem)

By default, the ID certificate is automatically renewed every 6 months, and the license entitlement is renewed every 30 days. You might want to manually renew the registration for either of these items if you have a limited window for Internet access, or if you make any licensing changes in the Smart Software Manager, for example.

Procedure

Step 1 Renew the ID certificate:

license smart renew id

Step 2 Renew the license entitlement:

license smart renew auth

Firepower 4100/9300: Configure Smart Software Licensing

This procedure applies for a chassis using the Smart Software Manager, Smart Software Manager On-Prem, or for Permanent License Reservation; see the FXOS configuration guide to configure your method as a prerequisite.

For Permanent License Reservation, the license enables all features: Standard tier with maximum Security Contexts and the Carrier license. However, for the ASA to know to use these features, you need to enable them on the ASA.

Before you begin

For an ASA cluster, you need to access the control unit for configuration. Check the Firewall Chassis Manager to see which unit is the control unit. You can also check from the ASA CLI, as shown in this procedure.

Procedure

Step 1 Connect to the Firepower 4100/9300 chassis CLI (console or SSH), and then session to the ASA:

connect module *slot* **console**
connect asa

Example:

```
Firepower> connect module 1 console
Firepower-module1> connect asa

asa>
```

The next time you connect to the ASA console, you go directly to the ASA; you do not need to enter **connect asa** again.

For an ASA cluster, you only need to access the control unit for license configuration and other configuration. Typically, the control unit is in slot 1, so you should connect to that module first.

Step 2 At the ASA CLI, enter global configuration mode. By default, the enable password is blank unless you set it when you deployed the logical device, but you are prompted to change the password the first time you enter the **enable** command.

enable
configure terminal

Example:

```
asa> enable
Password:
The enable password is not set. Please set it now.
Enter Password: ******
Repeat Password: ******
asa# configure terminal
asa(config)#
```

Step 3 If required, for an ASA cluster confirm that this unit is the control unit:

show cluster info

Example:

```
asa(config)# show cluster info
Cluster stbu: On
  This is "unit-1-1" in state SLAVE
    ID : 0
    Version : 9.5(2)
    Serial No.: P3000000025
    CCL IP : 127.2.1.1
    CCL MAC : 000b.fcf8.c192
    Last join : 17:08:59 UTC Sep 26 2015
    Last leave: N/A
Other members in the cluster:
  Unit "unit-1-2" in state SLAVE
    ID : 1
    Version : 9.5(2)
    Serial No.: P3000000001
    CCL IP : 127.2.1.2
    CCL MAC : 000b.fcf8.c162
    Last join : 19:13:11 UTC Sep 23 2015
    Last leave: N/A
  Unit "unit-1-3" in state MASTER
    ID : 2
    Version : 9.5(2)
    Serial No.: JAB0815R0JY
    CCL IP : 127.2.1.3
    CCL MAC : 000f.f775.541e
    Last join : 19:13:20 UTC Sep 23 2015
    Last leave: N/A
```

If a different unit is the control unit, exit the connection and connect to the correct unit. See below for information about exiting the connection.

Step 4 Enter license smart configuration mode:

license smart

Example:

```
ciscoasa(config)# license smart
ciscoasa(config-smart-lic)#
```

Step 5 Set the feature tier:

feature tier standard

Only the standard tier is available. A tier license is a prerequisite for adding other feature licenses. You must have sufficient tier licenses in your account. Otherwise, you cannot configure any other feature licenses or any features that require licenses.

Step 6 Request one or more of the following features:

- Carrier (GTP/GPRS, Diameter, and SCTP inspection)

  feature carrier

- Security Contexts

  feature context *<1-248>*

  For Permanent License Reservation, you can specify the maximum contexts (248).

- Strong Encryption (3DES/AES)

  feature strong-encryption

  This license is not required if you receive the strong encryption token from the Smart Software Manager. However, if your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. Only the active unit requests this license, and both units can use it due to license aggregation.

Example:

```
ciscoasa(config-smart-lic)# feature carrier
ciscoasa(config-smart-lic)# feature context 50
```

Step 7 To exit the ASA console, enter **~** at the prompt to exit to the Telnet application. Enter **quit** to exit back to the supervisor CLI.

Other Models: Configure Smart Software Licensing

This section describes how to configure Smart Software Licensing for the following models:

- Firepower 1010
- Firepower 1100
- Firepower 2100
- Secure Firewall 3100

Choose one of the following methods:

- Other Models: Configure Regular Smart Software Licensing, on page 130

  You can also (Optional) Other Models: Deregister the ASA (Regular and On-Prem), on page 141 or (Optional) Other Models: Renew the ASA ID Certificate or License Entitlement (Regular and On-Prem), on page 141.

- Other Models: Configure Smart Software Manager On-Prem Licensing, on page 134

  You can also (Optional) Other Models: Deregister the ASA (Regular and On-Prem), on page 141 or (Optional) Other Models: Renew the ASA ID Certificate or License Entitlement (Regular and On-Prem), on page 141.

- Other Models: Configure Permanent License Reservation, on page 137

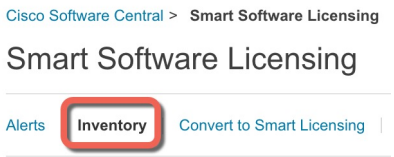
Other Models: Configure Regular Smart Software Licensing

This procedure applies for an ASA using the Smart Software Manager.

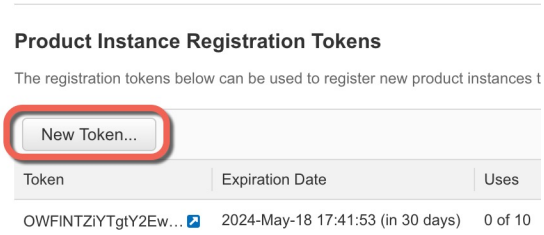
Procedure

Step 1 In the Cisco Smart Software Manager, request and copy a registration token for the virtual account to which you want to add this device.

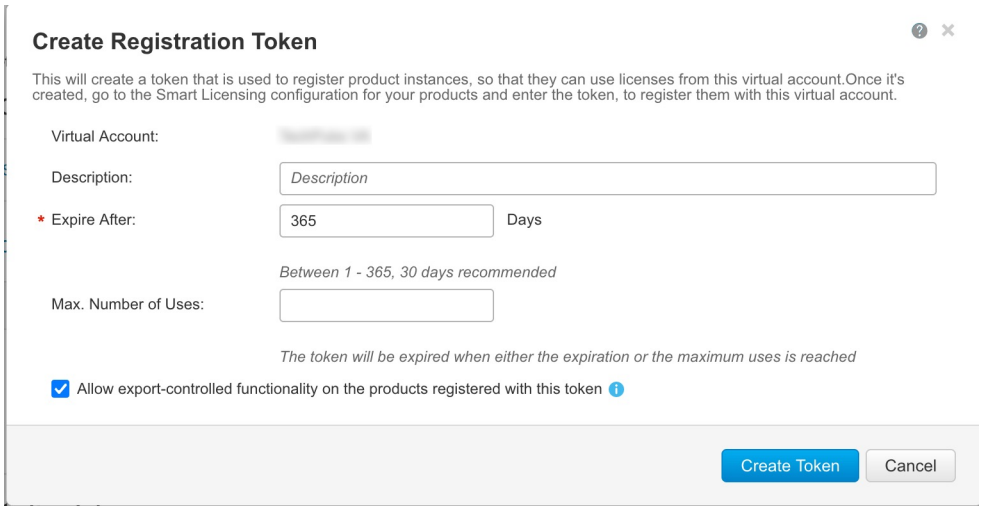
a. Click **Inventory**.



b. On the **General** tab, click **New Token**.



c. On the **Create Registration Token** dialog box enter the following settings, and then click **Create Token**:



- **Description**
- **Expire After**—Cisco recommends 30 days.
- **Max. Number of Uses**
- **Allow export-controlled functionaility on the products registered with this token**—Enables the export-compliance flag.

The token is added to your inventory.

d. Click the arrow icon to the right of the token to open the **Token** dialog box so you can copy the token ID to your clipboard. Keep this token ready for later in the procedure when you need to register the ASA.

Figure 17: View token

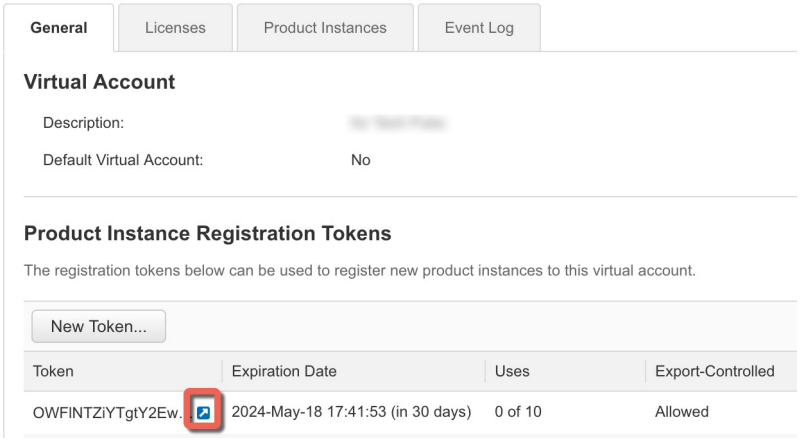
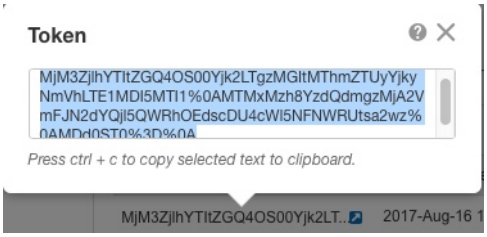


Figure 18: Copy token



Step 2 (Optional) On the ASA, specify the HTTP Proxy URL for Call Home:

call-home

http-proxy *ip_address* **port** *port*

If your network uses an HTTP proxy for internet access, you must configure the proxy address for Smart Software Licensing. This proxy is also used for Smart Transport and Smart Call Home in general.

Note
HTTP proxy with authentication is not supported.

Example:

```
ciscoasa(config)# call-home
ciscoasa(cfg-call-home)# http-proxy 10.1.1.1 port 443
```

Step 3 Request license entitlements on the ASA.

a) Enter license smart configuration mode:

license smart

Example:

```
ciscoasa(config)# license smart
ciscoasa(config-smart-lic)#
```

b) (Firepower 1000/2100) Set the feature tier:

feature tier standard

Only the standard tier is available, however, you need to enable it in the configuration; a tier license is a prerequisite for adding other feature licenses. For Secure Firewall models, the Standard license is always enabled and cannot be disabled.

c) (Optional) (Supported models) Request the security context license.

feature context *number*

By default, the ASA supports 2 contexts, so you should request the number of contexts you want minus the default contexts. The maximum number of contexts depends on your model:

- Firepower 1120—5 contexts
- Firepower 1140—10 contexts
- Firepower 1150—25 contexts

- Firepower 2110—25 contexts
- Firepower 2120—25 contexts
- Firepower 2130—30 contexts
- Firepower 2140—40 contexts
- Secure Firewall 3100—100 contexts

For example, to use the maximum of 25 contexts on the Firepower 1150, enter 23 for the number of contexts; this value is added to the default of 2.

Example:

```
ciscoasa(config-smart-lic)# feature context 18
```

d) (Optional) (Firepower 1010) Request the Security Plus license to enable failover.

feature security-plus

Example:

```
ciscoasa(config-smart-lic)# feature security-plus
```

e) (Optional) Enable strong encryption. **Do not** enter this command if you receive the strong encryption token.

feature strong-encryption

This license is not required if you receive the strong encryption token from the Smart Software Manager. However, if your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. Only the active unit requests this license, and both units can use it due to license aggregation.

Note
If you enter this command, your device will consume the strong encryption license, whether it's needed or not. So unless you have that license in your account, your account will go out of compliance. Since this license is not needed when using the strong encryption token, you may accidentally put your account out of compliance without any benefit.

Example:

```
ciscoasa(config-smart-lic)# feature strong-encryption
```

Step 4 Register the ASA using the token you copied in Step 1:

license smart register idtoken *id_token*

Example:

```
ciscoasa# license smart register idtoken YjE3Njc5MzYtMGQzMi00OTA4
LWJhODItNzBhMGQ5NGRlYjUxLTE0MTQ5NDAy%0AODQzNzl8NXk2bzV3SDE0ZkgwQk
dYRmZ1NTNCNGlvRnBHUFpjcm02WTB4TU4w%0Ac2NnMD0%3D%0A
```

The ASA registers with the Smart Software Manager and requests authorization for the configured license entitlements. The Smart Software Manager also applies the Strong Encryption (3DES/AES) license if your account allows. Use the **show license summary** command to check the license status and usage.

Example:

```
ciscoasa# show license summary

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: Biz1
  Virtual Account: IT
  Export-Controlled Functionality: Allowed
  Last Renewal Attempt: None
  Next Renewal Attempt: Mar 19 20:26:29 2018 UTC

License Authorization:
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Oct 23 01:41:26 2017 UTC

License Usage:
  License                 Entitlement tag               Count Status
  -----------------------------------------------------------------------------
  regid.2014-08.com.ci... (FP1010-ASA-Std)                     1 AUTHORIZED
```

Other Models: Configure Smart Software Manager On-Prem Licensing

This procedure applies for an ASA using a Smart Software Manager On-Prem.

Before you begin

- Download the Smart Software Manager On-Prem OVA file from Cisco.com and install and configure it on a VMwareESXi server. For more information, see the *Cisco Smart Software Manager On-Prem Data Sheet*.
- Download the crypto CA trustpool before you place the device in an air-gapped network. This trustpool is normally downloaded automatically, but may be out of date in an air-gapped network:

  crypto ca trustpool import url http://www.cisco.com/security/pki/trs/ios_core.p7b

Procedure

Step 1 Request a registration token on the Smart Software Manager On-Prem server.

Step 2 (Optional) On the ASA, specify the HTTP Proxy URL for Call Home:

call-home

http-proxy *ip_address* **port** *port*

If your network uses an HTTP proxy for internet access, you must configure the proxy address for Smart Software Licensing. This proxy is also used for Smart Transport and Smart Call Home in general.

Note
HTTP proxy with authentication is not supported.

Example:

```
ciscoasa(config)# call-home
ciscoasa(cfg-call-home)# http-proxy 10.1.1.1 port 443
```

Step 3 Change the license server URL to go to the Smart Software Manager On-Prem.

call-home

profile License

destination address http https://*on-prem_ip_address***/Transportgateway/services/DeviceRequestHandler**

Example:

```
ciscoasa(config)# call-home
ciscoasa(cfg-call-home)# profile License
ciscoasa(cfg-call-home-profile) destination address http
https://10.1.5.5/Transportgateway/services/DeviceRequestHandler
```

Step 4 Request license entitlements on the ASA.

a) Enter license smart configuration mode:

license smart

Example:

```
ciscoasa(config)# license smart
ciscoasa(config-smart-lic)#
```

b) (Firepower 1000/2100) Set the feature tier:

feature tier standard

Only the standard tier is available, however, you need to enable it in the configuration; a tier license is a prerequisite for adding other feature licenses. For Secure Firewall models, the Standard license is always enabled and cannot be disabled.

c) (Optional) (Supported models) Request the security context license.

feature context *number*

By default, the ASA supports 2 contexts, so you should request the number of contexts you want minus the default contexts. The maximum number of contexts depends on your model:

- Firepower 1120—5 contexts
- Firepower 1140—10 contexts
- Firepower 1150—25 contexts
- Firepower 2110—25 contexts

- Firepower 2120—25 contexts
- Firepower 2130—30 contexts
- Firepower 2140—40 contexts
- Secure Firewall 3100—100 contexts

For example, to use the maximum of 25 contexts on the Firepower 1150, enter 23 for the number of contexts; this value is added to the default of 2.

Example:

```
ciscoasa(config-smart-lic)# feature context 18
```

d) (Optional) (Firepower 1010) Request the Security Plus license to enable failover.

feature security-plus

Example:

```
ciscoasa(config-smart-lic)# feature security-plus
```

e) (Optional) Enable strong encryption. **Do not** enter this command if you receive the strong encryption token.

feature strong-encryption

This license is not required if you receive the strong encryption token from the Smart Software Manager. However, if your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. Only the active unit requests this license, and both units can use it due to license aggregation.

Note
If you enter this command, your device will consume the strong encryption license, whether it's needed or not. So unless you have that license in your account, your account will go out of compliance. Since this license is not needed when using the strong encryption token, you may accidentally put your account out of compliance without any benefit.

Example:

```
ciscoasa(config-smart-lic)# feature strong-encryption
```

Step 5 Register the ASA using the token you requested in Step 1:

license smart register idtoken *id_token*

Example:

```
ciscoasa# license smart register idtoken YjE3Njc5MzYtMGQzMi00OTA4
LWJhODItNzBhMGQ5NGRlYjUxLTE0MTQ5NDAy%0AODQzNzl8NXk2bzV3SDE0ZkgwQk
dYRmZ1NTNCNGlvRnBHUFpjcm02WTB4TU4w%0Ac2NnMD0%3D%0A
```

The ASA registers with the Smart Software Manager On-Prem server and requests authorization for the configured license entitlements. The Smart Software Manager On-Prem server also applies the Strong

Encryption (3DES/AES) license if your account allows. Use the **show license summary** command to check the license status and usage.

Example:

```
ciscoasa# show license summary

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: Biz1
  Virtual Account: IT
  Export-Controlled Functionality: Allowed
  Last Renewal Attempt: None
  Next Renewal Attempt: Mar 19 20:26:29 2018 UTC

License Authorization:
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Oct 23 01:41:26 2017 UTC

License Usage:
  License                 Entitlement tag               Count Status
  -----------------------------------------------------------------------------
  regid.2014-08.com.ci... (FP1010-ASA-Std)                    1 AUTHORIZED
```

Other Models: Configure Permanent License Reservation

You can assign a permanent license. This section also describes how to return a license if you retire the ASA.

Procedure

Step 1 Install the 1000/2100/3100 Permanent License, on page 137.

Step 2 (Optional) Other Models: Return the Permanent License, on page 140.

Install the 1000/2100/3100 Permanent License

For ASAs that do not have internet access, you can request a permanent license from the Smart Software Manager. The permanent license enables all features: Standard license with maximum Security Contexts.

Note For permanent license reservation, you must return the license before you decommission the ASA. If you do not officially return the license, the license remains in a used state and cannot be reused for a new ASA. See (Optional) Other Models: Return the Permanent License, on page 140.

Before you begin

Purchase permanent licenses so they are available in the Smart Software Manager. Not all accounts are approved for permanent license reservation. Make sure you have approval from Cisco for this feature before you attempt to configure it.

Procedure

Step 1 At the ASA CLI, enable permanent license reservation:

license smart reservation

Example:

```
ciscoasa (config)# license smart reservation
ciscoasa (config)#
```

Step 2 Request the license code to enter in the Smart Software Manager:

license smart reservation request universal

Example:

```
ciscoasa# license smart reservation request universal
Enter this request code in the Cisco Smart Software Manager portal:
BB-ZFPR-2140:JAD200802RR-AzKmHcc71-2A
ciscoasa#
```

If you re-enter this command, then the same code is displayed, even after a reload. If you have not yet entered this code into the Smart Software Manager and want to cancel the request, enter:

license smart reservation cancel

If you disable permanent license reservation, then any pending requests are canceled. If you already entered the code into the Smart Software Manager, then you must complete this procedure to apply the license to the ASA, after which point you can return the license if desired. See (Optional) Other Models: Return the Permanent License, on page 140.

Step 3 Go to the Smart Software Manager Inventory screen, and click the **Licenses** tab:

https://software.cisco.com/#SmartLicensing-Inventory

The **Licenses** tab displays all existing licenses related to your account, both regular and permanent.

Step 4 Click **License Reservation**, and type the ASA code into the box. Click **Reserve License**.

The Smart Software Manager generates an authorization code. You can download the code or copy it to the clipboard. At this point, the license is now in use according to the Smart Software Manager.

If you do not see the **License Reservation** button, then your account is not authorized for permanent license reservation. In this case, you should disable permanent license reservation and re-enter the regular smart license commands.

Step 5 On the ASA, enter the authorization code:

license smart reservation install *code*

Example:

```
ciscoasa# license smart reservation install AAu3431rGRS00Ig5HQl2vpzg{MEYCIQCBw$
ciscoasa#
```

Step 6 Request license entitlements on the ASA.

Note
Although the permanent license allows the full use of all of the licenses, you still need to turn on the entitlements in the ASA configuration so that the ASA knows it can use them.

a) Enter license smart configuration mode:

license smart

Example:

```
ciscoasa(config)# license smart
ciscoasa(config-smart-lic)#
```

b) (Firepower 1000/2100) Set the feature tier:

feature tier standard

Only the standard tier is available, however, you need to enable it in the configuration; a tier license is a prerequisite for adding other feature licenses. For Secure Firewall models, the Standard license is always enabled and cannot be disabled.

c) (Optional) (Supported models) Request the security context license.

feature context *number*

By default, the ASA supports 2 contexts, so you should request the number of contexts you want minus the default contexts. The maximum number of contexts depends on your model:

- Firepower 1120—5 contexts
- Firepower 1140—10 contexts
- Firepower 1150—25 contexts
- Firepower 2110—25 contexts
- Firepower 2120—25 contexts
- Firepower 2130—30 contexts
- Firepower 2140—40 contexts
- Secure Firewall 3100—100 contexts

For example, to use the maximum of 25 contexts on the Firepower 1150, enter 23 for the number of contexts; this value is added to the default of 2.

Example:

```
ciscoasa(config-smart-lic)# feature context 18
```

d) (Optional) (Firepower 1010) Request the Security Plus license to enable failover.

feature security-plus

Example:

```
ciscoasa(config-smart-lic)# feature security-plus
```

e) (Optional) Enable strong encryption. **Do not** enter this command if you receive the strong encryption token.

feature strong-encryption

This license is not required if you receive the strong encryption token from the Smart Software Manager. However, if your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. Only the active unit requests this license, and both units can use it due to license aggregation.

Note
If you enter this command, your device will consume the strong encryption license, whether it's needed or not. So unless you have that license in your account, your account will go out of compliance. Since this license is not needed when using the strong encryption token, you may accidentally put your account out of compliance without any benefit.

Example:

```
ciscoasa(config-smart-lic)# feature strong-encryption
```

(Optional) Other Models: Return the Permanent License

If you no longer need a permanent license (for example, you are retiring an ASA), you must officially return the license to the Smart Software Manager using this procedure. If you do not follow all steps, then the license stays in a used state and cannot easily be freed up for use elsewhere.

Procedure

Step 1 On the ASA, generate a return code:

license smart reservation return

Example:

```
ciscoasa# license smart reservation return
Enter this return code in the Cisco Smart Software Manager portal:
Au3431rGRS00Ig5HQl2vpcg{uXiTRfVrp7M/zDpirLwYCaq8oSv60yZJuFDVBS2QliQ=
```

The ASA immediately becomes unlicensed and moves to the Evaluation state. If you need to view this code again, re-enter this command. Note that if you request a new permanent license (**license smart reservation request universal**), then you cannot re-display this code. Be sure to capture the code to complete the return.

If the evaluation period has expired, then the ASA moves into an expired state. For more information about out-of-compliance states, see Out-of-Compliance State, on page 159.

Step 2 View the ASA universal device identifier (UDI) so you can find this ASA instance in the Smart Software Manager:

show license udi

Example:

```
ciscoasa# show license udi
UDI: PID:FPR-2140,SN:JAD200802RR
ciscoasa#
```

Step 3 Go to the Smart Software Manager Inventory screen, and click the **Product Instances** tab:

https://software.cisco.com/#SmartLicensing-Inventory

The **Product Instances** tab displays all licensed products by the UDI.

Step 4 Find the ASA you want to unlicense, choose **Actions** > **Remove**, and type the ASA return code into the box. Click **Remove Product Instance**.

The permanent license is returned to the available pool.

(Optional) Other Models: Deregister the ASA (Regular and On-Prem)

Deregistering the ASA removes the ASA from your account. All license entitlements and certificates on the ASA are removed. You might want to deregister to free up a license for a new ASA. Alternatively, you can remove the ASA from the Smart Software Manager. This procedure applies to the following device models:

- Firepower 1000/2100
- /3100

Procedure

Deregister the ASA:

license smart deregister

(Optional) Other Models: Renew the ASA ID Certificate or License Entitlement (Regular and On-Prem)

By default, the ID certificate is automatically renewed every 6 months, and the license entitlement is renewed every 30 days. You might want to manually renew the registration for either of these items if you have a

limited window for internet access, or if you make any licensing changes in the Smart Software Manager, for example.

Procedure

Step 1 Renew the ID certificate:

license smart renew id

Step 2 Renew the license entitlement:

license smart renew auth

Licenses Per Model

This section lists the license entitlements available for the ASAv and Firepower 4100/9300 chassis ASA security module.

ASAv

When you set the throughput level in the ASA configuration using the **throughput level** command, it determines the license requested from the Smart Software Manager. See the following throughput level/license relationship:

- 100M—ASAv5
- 1G—ASAv10
- 2G—ASAv30
- 10G—ASAv50
- 20G—ASAv100

The throughput level also determines the maximum AnyConnect Client and TLS proxy sessions. However, a lower ASAv memory profile will cap your actual number of sessions, so to determine your sessions, you need to check both the throughput level and the memory installed.

The memory of your ASAv determines the maximum concurrent firewall connections and VLANs, and is not determined by the throughput level.

The following table shows the licensed features for the ASAv series.

Licenses	Description
Entitlement Licenses	

Licenses	Description
Throughput Level	You set the throughput level in the ASA configuration using the **throughput level** command. This level determines the license you need. 100M: ASAv5 1G: ASAv10 2G: ASAv30 10G: ASAv50 20G: ASAv100
Firewall Licenses	
Botnet Traffic Filter	Enabled
Firewall Connections, Concurrent	Firewall connections are determined by the ASAv memory. 2 GB to 7.9 GB: 100,000 8 GB to 15.9 GB: 500,000 16 GB to 31.9 GB: 2,000,000 32 GB to 64 GB: 4,000,000
Carrier	Enabled
Total TLS Proxy Sessions	TLS Proxy Sessions are determined by the throughput level and ASAv memory. 100M throughput + any memory: 500 1G throughput + any memory: 500 2G throughput: • 2 GB to 7.9 GB memory: 500 • 8 GB+ memory: 1000 10G throughput: • 2 GB to 7.9 GB memory: 500 • 8 GB to 15.9 GB memory: 1000 • 16 GB+ memory: 10,000 20G throughput: • 2 GB to 7.9 GB memory: 500 • 8 GB to 15.9 GB memory: 1000 • 16 GB to 31.9 GB memory: 10,000 • 32 GB+ memory: 20,000

Licenses	Description	
VPN Licenses		
AnyConnect Client peers	Unlicensed	AnyConnect Client peers are determined by the throughput level and ASAv memory. *Optional AnyConnect Plus or AnyConnect Apex license, Maximums:* *100M throughput + any memory: 50* *1G throughput + any memory: 250* *2G throughput:* • *2 GB to 7.9 GB memory: 250* • *8 GB+ memory: 750* *10G throughput:* • *2 GB to 7.9 GB memory: 250* • *8 GB to 15.9 GB memory: 750* • *16 GB+ memory: 10,000* *20G throughput:* • *2 GB to 7.9 GB memory: 250* • *8 GB to 15.9 GB memory: 750* • *16 GB to 31.9 GB: 10,000* • *32 GB+ memory: 20,000*

Licenses	Description
Other VPN Peers	**Note** Other VPN peers are determined by the throughput level and ASAv memory. 100M throughput + any memory: 50 1G throughput + any memory: 250 2G throughput: • 2 GB to 7.9 GB memory: 250 • 8 GB+ memory: 750 10G throughput: • 2 GB to 7.9 GB memory: 250 • 8 GB to 15.9 GB memory: 750 • 16 GB+ memory: 10,000 20G throughput: • 2 GB to 7.9 GB memory: 250 • 8 GB to 15.9 GB memory: 750 • 16 GB to 31.9 GB: 10,000 • 32 GB+ memory: 20,000

Licenses	Description
Total VPN Peers, combined all types	**Note** Total VPN peers are determined by the throughput level and ASAv memory. 100M throughput + any memory: 50 1G throughput + any memory: 250 2G throughput: • 2 GB to 7.9 GB memory: 250 • 8 GB+ memory: 750 10G throughput: • 2 GB to 7.9 GB memory: 250 • 8 GB to 15.9 GB memory: 750 • 16 GB+ memory: 10,000 20G throughput: • 2 GB to 7.9 GB memory: 250 • 8 GB to 15.9 GB memory: 750 • 16 GB to 31.9 GB: 10,000 • 32 GB+ memory: 20,000
General Licenses	
Encryption	Base (DES) or Strong (3DES/AES), depending on the account's export compliance setting
Failover	Active/Standby
Security Contexts	No support
Clustering	Enabled
VLANs, Maximum	VLANs are determined by the ASAv memory. 2 GB to 7.9 GB—50 8 GB to 15.9 GB—200 16 GB to 31.9 GB—1024 32 GB to 64 GB—1024

Firepower 1010

The following table shows the licensed features for the Firepower 1010.

Licenses	Standard License	
Firewall Licenses		
Botnet Traffic Filter	No Support.	
Firewall Conns, Concurrent	100,000	
Carrier	No support. Although SCTP inspection maps are not supported, SCTP stateful inspection using ACLs is supported.	
Total TLS Proxy Sessions	4,000	
VPN Licenses		
AnyConnect Client peers	Unlicensed	*Optional AnyConnect Plus,AnyConnect Apex, or AnyConnect VPN Only license, Maximum: 75*
Other VPN Peers	75	
Total VPN Peers, combined all types	75	
General Licenses		
Encryption	Base (DES) or Strong (3DES/AES), depending on the account's export compliance setting	
Security Plus (failover, VPN Load Balancing)	Disabled	*Optional*
Security Contexts	No support.	
Clustering	No support.	
VLANs, Maximum	60	

Firepower 1100 Series

The following table shows the licensed features for the Firepower 1100 series.

Licenses	Standard License
Firewall Licenses	
Botnet Traffic Filter	No Support.
Firewall Conns, Concurrent	Firepower 1120: 200,000 Firepower 1140: 400,000 Firepower 1150: 600,000

Licenses	Standard License	
Carrier	No support. Although SCTP inspection maps are not supported, SCTP stateful inspection using ACLs is supported.	
Total TLS Proxy Sessions	Firepower 1120: 4,000 Firepower 1140: 8,000 Firepower 1150: 8,000	
VPN Licenses		
AnyConnect Client peers	Unlicensed	*Optional AnyConnect Plus, AnyConnect Apex, or AnyConnect VPN Only license, Maximum:* *Firepower 1120: 150* *Firepower 1140: 400* *Firepower 1150: 800*
Other VPN Peers	Firepower 1120: 150 Firepower 1140: 400 Firepower 1150: 800	
Total VPN Peers, combined all types	Firepower 1120: 150 Firepower 1140: 400 Firepower 1150: 800	
General Licenses		
Encryption	Base (DES) or Strong (3DES/AES), depending on the account's export compliance setting	
Security Contexts	2	*Optional License, Maximum:* *Firepower 1120: 5* *Firepower 1140: 10* *Firepower 1150: 25*
Clustering	No support.	
VLANs, Maximum	1024	

Firepower 2100 Series

The following table shows the licensed features for the Firepower 2100 series.

Licenses	Standard License
Firewall Licenses	

Licenses	Standard License	
Botnet Traffic Filter	No Support.	
Firewall Conns, Concurrent	Firepower 2110: 1,000,000 Firepower 2120: 1,500,000 Firepower 2130: 2,000,000 Firepower 2140: 3,000,000	
Carrier	No support. Although SCTP inspection maps are not supported, SCTP stateful inspection using ACLs is supported.	
Total TLS Proxy Sessions	Firepower 2110: 4,000 Firepower 2120: 8,000 Firepower 2130: 8,000 Firepower 2140: 10,000	
VPN Licenses		
AnyConnect Client peers	Unlicensed	*Optional AnyConnect Plus, AnyConnect Apex, or AnyConnect VPN Only license, Maximum:* *Firepower 2110: 1,500* *Firepower 2120: 3,500* *Firepower 2130: 7,500* *Firepower 2140: 10,000*
Other VPN Peers	Firepower 2110: 1,500 Firepower 2120: 3,500 Firepower 2130: 7,500 Firepower 2140: 10,000	
Total VPN Peers, combined all types	Firepower 2110: 1,500 Firepower 2120: 3,500 Firepower 2130: 7,500 Firepower 2140: 10,000	
General Licenses		
Encryption	Base (DES) or Strong (3DES/AES), depending on the account's export compliance setting	

Licenses	Standard License	
Security Contexts	2	*Optional License, Maximum:* *Firepower 2110: 25* *Firepower 2120: 25* *Firepower 2130: 30* *Firepower 2140: 40*
Clustering	No support.	
VLANs, Maximum	1024	

Secure Firewall 3100 Series

The following table shows the licensed features for the Secure Firewall 3100 series.

Licenses	Standard License	
Firewall Licenses		
Botnet Traffic Filter	No Support.	
Firewall Conns, Concurrent	Secure Firewall 3110: 2,000,000 Secure Firewall 3120: 4,000,000 Secure Firewall 3130: 6,000,000 Secure Firewall 3140: 10,000,000	
Carrier	No support. Although SCTP inspection maps are not supported, SCTP stateful inspection using ACLs is supported.	
Total TLS Proxy Sessions	Secure Firewall 3110: 10,000 Secure Firewall 3120: 15,000 Secure Firewall 3130: 15,000 Secure Firewall 3140: 15,000	
VPN Licenses		

Licenses	Standard License	
AnyConnect Client peers	Unlicensed	*Optional AnyConnect Plus, AnyConnect Apex, or AnyConnect VPN Only license, Maximum:* *Secure Firewall 3110: 3000* *Secure Firewall 3120: 7000* *Secure Firewall 3130: 15,000* *Secure Firewall 3140: 20,000*
Other VPN Peers	Secure Firewall 3110: 3000 Secure Firewall 3120: 7000 Secure Firewall 3130: 15,000 Secure Firewall 3140: 20,000	
Total VPN Peers, combined all types	Secure Firewall 3110: 3000 Secure Firewall 3120: 7000 Secure Firewall 3130: 15,000 Secure Firewall 3140: 20,000	
General Licenses		
Encryption	Base (DES) or Strong (3DES/AES), depending on the account's export compliance setting	
Security Contexts	2	*Optional License, Maximum: 100*
Clustering	Enabled	
VLANs, Maximum	1024	

Firepower 4100

The following table shows the licensed features for the Firepower 4100.

Licenses	Standard License
Firewall Licenses	
Botnet Traffic Filter	No Support.

Licenses	Standard License	
Firewall Conns, Concurrent	Firepower 4110: 10,000,000 Firepower 4112: 10,000,000 Firepower 4115: 15,000,000 Firepower 4120: 15,000,000 Firepower 4125: 25,000,000 Firepower 4140: 25,000,000 Firepower 4145: 40,000,000 Firepower 4150: 35,000,000	
Carrier	Disabled	*Optional License: Carrier*
Total TLS Proxy Sessions	Firepower 4110: 10,000 All others: 15,000	
VPN Licenses		
AnyConnect Client peers	Unlicensed	*Optional AnyConnect Plus, AnyConnect Apex, or AnyConnect VPN Only license:* *Firepower 4110: 10,000* *Firepower 4112: 10,000* *Firepower 4115: 15,000* *Firepower 4120: 15,000* *Firepower 4125: 20,000* *Firepower 4140: 20,000* *Firepower 4145: 20,000* *Firepower 4150: 20,000*
Other VPN Peers	Firepower 4110: 10,000 Firepower 4112: 10,000 Firepower 4115: 15,000 Firepower 4120: 15,000 Firepower 4125: 20,000 Firepower 4140: 20,000 Firepower 4145: 20,000 Firepower 4150: 20,000	

Licenses	Standard License	
Total VPN Peers, combined all types	Firepower 4110: 10,000 Firepower 4112: 10,000 Firepower 4115: 15,000 Firepower 4120: 15,000 Firepower 4125: 20,000 Firepower 4140: 20,000 Firepower 4145: 20,000 Firepower 4150: 20,000	
General Licenses		
Encryption	Base (DES) or Strong (3DES/AES), depending on the account's export compliance setting	
Security Contexts	10	*Optional License: Maximum of 250*
Clustering	Enabled	
VLANs, Maximum	1024	

Firepower 9300

The following table shows the licensed features for the Firepower 9300.

Licenses	Standard License	
Firewall Licenses		
Botnet Traffic Filter	No Support.	
Firewall Conns, Concurrent	Firepower 9300 SM-56: 60,000,000 Firepower 9300 SM-48: 60,000,000 Firepower 9300 SM-44: 60,000,000 Firepower 9300 SM-40: 55,000,000 Firepower 9300 SM-36: 60,000,000 Firepower 9300 SM-24: 55,000,000	
Carrier	Disabled	*Optional License: Carrier*
Total TLS Proxy Sessions	15,000	
VPN Licenses		

Licenses	Standard License	
AnyConnect Client peers	Unlicensed	*Optional AnyConnect Plus, AnyConnect Apex, or AnyConnect VPN Only license: 20,000 maximum*
Other VPN Peers	20,000	
Total VPN Peers, combined all types	20,000	
General Licenses		
Encryption	Base (DES) or Strong (3DES/AES), depending on the account's export compliance setting	
Security Contexts	10	*Optional License: Maximum of 250*
Clustering	Enabled	
VLANs, Maximum	1024	

Monitoring Smart Software Licensing

You can monitor the license features, status, and certificate, as well as enable debug messages.

Viewing Your Current License

See the following commands for viewing your license:

- **show license features**

  The following example shows the ASA with only a Standard license:

```
ciscoasa# show license features
Serial Number:  JAD254312UA
Export Compliant: YES

License mode: Smart Licensing

Licensed features for this platform:
Maximum Physical Interfaces       : Unlimited
Maximum VLANs                     : 512
Inside Hosts                      : Unlimited
Failover                          : Active/Active
Encryption-DES                    : Enabled
Encryption-3DES-AES               : Enabled
Security Contexts                 : 2
Carrier                           : Disabled
AnyConnect Premium Peers          : 150
AnyConnect Essentials             : Disabled
Other VPN Peers                   : 150
Total VPN Peers                   : 150
AnyConnect for Mobile             : Enabled
```

```
AnyConnect for Cisco VPN Phone    : Enabled
Advanced Endpoint Assessment      : Enabled
Shared License                    : Disabled
Total TLS Proxy Sessions          : 320
Cluster                           : Disabled
```

Viewing Smart License Status

See the following commands for viewing license status:

- **show license all**

  Displays the state of Smart Software Licensing, Smart Agent version, UDI information, Smart Agent state, global compliance status, the entitlements status, licensing certificate information, and scheduled Smart Agent tasks.

  The following example shows an ASAv license:

```
ciscoasa# show license all
Smart Licensing Status
======================

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: ASA
  Virtual Account: ASAv Internal Users
  Export-Controlled Functionality: Not Allowed
  Initial Registration: SUCCEEDED on Sep 21 20:26:29 2015 UTC
  Last Renewal Attempt: None
  Next Renewal Attempt: Mar 19 20:26:28 2016 UTC
  Registration Expires: Sep 20 20:23:25 2016 UTC

License Authorization:
  Status: AUTHORIZED on Sep 21 21:17:35 2015 UTC
  Last Communication Attempt: SUCCEEDED on Sep 21 21:17:35 2015 UTC
  Next Communication Attempt: Sep 24 00:44:10 2015 UTC
  Communication Deadline: Dec 20 21:14:33 2015 UTC

License Usage
==============

regid.2014-08.com.cisco.ASAv-STD-1G,1.0_4fd3bdbd-29ae-4cce-ad82-45ad3db1070c
(ASAv-STD-1G):
  Description: This entitlement tag was created via Alpha Extension application
  Count: 1
  Version: 1.0
  Status: AUTHORIZED

Product Information
===================
UDI: PID:ASAv,SN:9AHV3KJBEKE

Agent Version
=============
Smart Agent for Licensing: 1.6_reservation/36
```

- **show license status**

Shows the smart license status.

The following example shows the status for the ASAv using regular smart software licensing:

```
ciscoasa# show license status

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: ASA
  Virtual Account: ASAv Internal Users
  Export-Controlled Functionality: Not Allowed
  Initial Registration: SUCCEEDED on Sep 21 20:26:29 2015 UTC
  Last Renewal Attempt: None
  Next Renewal Attempt: Mar 19 20:26:28 2016 UTC
  Registration Expires: Sep 20 20:23:25 2016 UTC

License Authorization:
  Status: AUTHORIZED on Sep 23 01:41:26 2015 UTC
  Last Communication Attempt: SUCCEEDED on Sep 23 01:41:26 2015 UTC
  Next Communication Attempt: Oct 23 01:41:26 2015 UTC
  Communication Deadline: Dec 22 01:38:25 2015 UTC
```

The following example shows the status for the ASAv using permanent license reservation:

```
ciscoasa# show license status

Smart Licensing is ENABLED
License Reservation is ENABLED

Registration:
  Status: REGISTERED - UNIVERSAL LICENSE RESERVATION
  Export-Controlled Functionality: Allowed
  Initial Registration: SUCCEEDED on Jan 28 16:42:45 2016 UTC

License Authorization:
  Status: AUTHORIZED - RESERVED on Jan 28 16:42:45 2016 UTC

Licensing HA configuration error:
    No Reservation Ha config error
```

- **show license summary**

  Shows a summary of smart license status and usage.

  The following example shows the summary for the ASAv using regular smart software licensing:

```
ciscoasa# show license summary

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED
  Smart Account: ASA
  Virtual Account: ASAv Internal Users
  Export-Controlled Functionality: Not Allowed
  Last Renewal Attempt: None
  Next Renewal Attempt: Mar 19 20:26:29 2016 UTC

License Authorization:
```

```
  Status: AUTHORIZED
  Last Communication Attempt: SUCCEEDED
  Next Communication Attempt: Oct 23 01:41:26 2015 UTC

License Usage:
  License                 Entitlement tag               Count Status
  -----------------------------------------------------------------------------
  regid.2014-08.com.ci... (ASAv-STD-1G)                     1 AUTHORIZED
```

The following example shows the summary for the ASAv using permanent license reservation:

```
ciscoasa# show license summary

Smart Licensing is ENABLED

Registration:
  Status: REGISTERED - UNIVERSAL LICENSE RESERVATION
  Export-Controlled Functionality: Allowed

License Authorization:
  Status: AUTHORIZED - RESERVED
```

- **show license usage**

  Shows the smart license usage.

  The following example shows the usage for the ASAv:

```
ciscoasa# show license usage

License Authorization:
  Status: AUTHORIZED on Sep 23 01:41:26 2015 UTC

regid.2014-08.com.cisco.ASAv-STD-1G,1.0_4fd3bdbd-29ae-4cce-ad82-45ad3db1070c
(ASAv-STD-1G):
  Description: This entitlement tag was created via Alpha Extension application
  Count: 1
  Version: 1.0
  Status: AUTHORIZED
```

Viewing the UDI

See the following command to view the universal product identifier (UDI):

show license udi

The following example shows the UDI for the ASAv:

```
ciscoasa# show license udi
UDI: PID:ASAv,SN:9AHV3KJBEKE
ciscoasa#
```

Debugging Smart Software Licensing

See the following commands for debugging clustering:

- **debug license agent** {**error** | **trace** | **debug** | **all**}

  Turns on debugging from the Smart Agent.

- **debug license** *level*

  Turns on various levels of Smart Software Licensing Manager debugs.

Smart Software Manager Communication

This section describes how your device communicates with the Smart Software Manager.

Device Registration and Tokens

For each virtual account, you can create a registration token. This token is valid for 30 days by default. Enter this token ID plus entitlement levels when you deploy each device, or when you register an existing device. You can create a new token if an existing token is expired.



Note Firepower 4100/9300 chassis—Device registration is configured in the chassis, not on the ASA logical device.

At startup after deployment, or after you manually configure these parameters on an existing device, the device registers with the Smart Software Manager. When the device registers with the token, the Smart Software Manager issues an ID certificate for communication between the device and the Smart Software Manager. This certificate is valid for 1 year, although it will be renewed every 6 months.

Periodic Communication with the Smart Software Manager

The device communicates with the Smart Software Manager every 30 days. If you make changes in the Smart Software Manager, you can refresh the authorization on the device so the change takes place immediately. Or you can wait for the device to communicate as scheduled.

You can optionally configure an HTTP proxy.

ASAv

The ASAv must have internet access either directly or through an HTTP proxy at least every 90 days. Normal license communication occurs every 30 days, but with the grace period, your device will stay compliant for up to 90 days without calling home. After the grace period, you should contact the Smart Software Manager, or your ASAv will be out-of-compliance; operation is otherwise unaffected.

All Other Models

The ASA must have internet access either directly or through an HTTP proxy at least every 90 days. Normal license communication occurs every 30 days, but with the grace period, your device will operate for up to 90 days without calling home. After the grace period, you must contact the Smart Software Manager, or you will not be able to make configuration changes to features requiring special licenses; operation is otherwise unaffected.

Out-of-Compliance State

The device can become out of compliance in the following situations:

- Over-utilization—The device is using more licenses than are available.
- License expiration—A time-based license has expired.
- Lack of communication—The device cannot reach the licensing authority for re-authorization, causing the authorization to expire.

To verify whether your device is out of compliance, or approaching that status, you must compare the entitlements currently in use by your device against those in your Smart Account.

When the device is out of compliance, operations may be limited, depending on the model:

- ASAv—The system enters a restrictive state that limits concurrent AnyConnect sessions to two and throttles total throughput to 100 Kbps. This rate limiting ensures that the device continues to operate but with significantly reduced capacity until a valid license is installed. This behavior is specific to the ASA virtual model.
- All Other Models—You will not be able to make configuration changes to features requiring special licenses, but operation is otherwise unaffected. For example, existing contexts over the Standard license limit can continue to run, and you can modify their configuration, but you will not be able to add a *new* context. If you do not have sufficient Standard licenses when you first register, you cannot configure any licensed features, including strong encryption features.

Smart Call Home Infrastructure

By default, a Smart Call Home profile exists in the configuration that specifies the URL for the Smart Software Manager. You cannot remove this profile. Note that the only configurable option for the License profile is the destination address URL for the Smart Software Manager. Unless directed by Cisco TAC, you should not change the Smart Software Manager URL.



Note For the Firepower 4100/9300 chassis, Smart Call Home for licensing is configured in the Firepower 4100/9300 chassis supervisor, not on the ASA.

You cannot disable Smart Call Home for Smart Software Licensing. For example, even if you disable Smart Call Home using the **no service call-home** command, Smart Software Licensing is not disabled.

Other Smart Call Home functions are not turned on unless you specifically configure them.

Smart License Certificate Management

The ASA automatically creates a trustpoint containing the certificate of the CA that issued the Smart Transport or Smart Call Home server certificate. To avoid service interruption if the issuing hierarchy of the server certificate changes, configure the **auto-update** command to enable the automatic update of the trustpool bundle at periodic intervals.

The server certificate received from a Smart License Server must contain "ServAuth" in the Extended Key Usage field. This check will be done on non self-signed certificates only; self-signed certificates do not provide any value in this field.

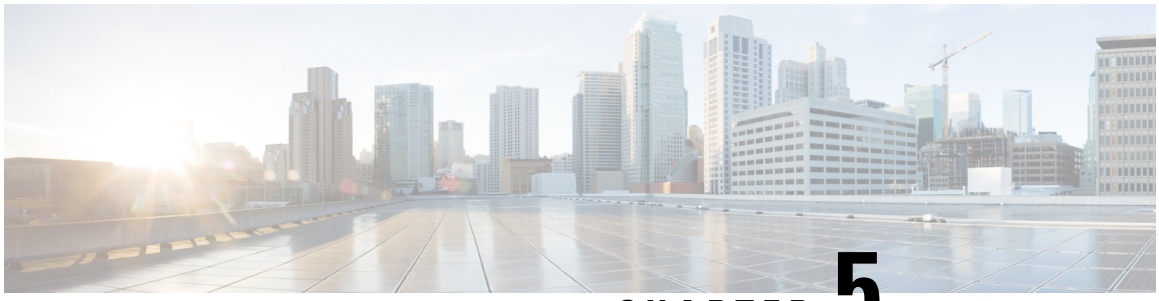
Note Reloading a device with an expired licensing registration for over a year, will disable the export-controlled functionality. This can lead to critical issues such as VPN downtime or failover split-brain scenarios when IPsec encryption is enabled.

History for Smart Software Licensing

Feature Name	Platform Releases	Description
ASAv100 permanent license reservation	9.14(1.30)	The ASAv100 now supports permanent license reservation using product ID L-ASAV100SR-K9=. **Note:** Not all accounts are approved for permanent license reservation.
ASAv MSLA Support	9.13(1)	The ASAv supports Cisco's Managed Service License Agreement (MSLA) program, which is a software licensing and consumption framework designed for Cisco customers and partners who offer managed software services to third parties. MSLA is a new form of Smart Licensing where the licensing Smart Agent keeps track of the usage of licensing entitlements in units of time. New/Modified commands: **license smart**, **mode**, **utility**, **custom-id**, **custom-info**, **privacy**, **transport type**, **transport url**, **transport proxy**
ASAv Flexible Licensing	9.13(1)	Flexible Licensing is a new form of Smart Licensing where any ASAv license now can be used on any supported ASAv vCPU/memory configuration. Session limits for AnyConnect Client and TLS proxy will be determined by the ASAv platform entitlement installed rather than a platform limit tied to a model type. New/Modified commands: **show version**, **show vm**, **show cpu**, **show license features**
Licensing changes for failover pairs on the Firepower 4100/9300 chassis	9.7(1)	Only the active unit requests the license entitlements. Previously, both units requested license entitlements. Supported with FXOS 2.1.1.
Permanent License Reservation for the ASAv Short String enhancement	9.6(2)	Due to an update to the Smart Agent (to 1.6.4), the request and authorization codes now use shorter strings. We did not modify any commands.
Satellite Server support for the ASAv	9.6(2)	If your devices cannot access the internet for security reasons, you can optionally install a local Smart Software Manager satellite server as a virtual machine (VM). We did not modify any commands.

Feature Name	Platform Releases	Description
Permanent License Reservation for the ASA on the Firepower 4100/9300 chassis	9.6(2)	For highly secure environments where communication with the Cisco Smart Software Manager is not allowed, you can request a permanent license for the ASA on the Firepower 9300 and Firepower 4100. All available license entitlements are included in the permanent license, including the Standard Tier, Strong Encryption (if qualified), Security Contexts, and Carrier licenses. Requires FXOS 2.0.1. All configuration is performed on the Firepower 4100/9300 chassis; no configuration is required on the ASA.
Permanent License Reservation for the ASAv	9.5(2.200) 9.6(2)	For highly secure environments where communication with the Cisco Smart Software Manager is not allowed, you can request a permanent license for the ASAv. In 9.6(2), we also added support for this feature for the ASAv on Amazon Web Services. This feature is not supported for Microsoft Azure. We introduced the following commands: **license smart reservation, license smart reservation cancel, license smart reservation install, license smart reservation request universal, license smart reservation return**
Smart Agent Upgrade to v1.6	9.5(2.200) 9.6(2)	The smart agent was upgraded from Version 1.1 to Version 1.6. This upgrade supports permanent license reservation and also supports setting the Strong Encryption (3DES/AES) license entitlement according to the permission set in your license account. **Note** If you downgrade from Version 9.5(2.200), the ASAv does not retain the licensing registration state. You need to re-register with the **license smart register idtoken** *id_token* **force** command; obtain the ID token from the Smart Software Manager. We introduced the following commands: **show license status, show license summary, show license udi, show license usage** We modified the following commands: **show license all, show tech-support license** We deprecated the following commands: **show license cert, show license entitlement, show license pool, show license registration**

Feature Name	Platform Releases	Description
Strong Encryption (3DES) license automatically applied for the ASA on the Firepower 9300	9.5(2.1)	For regular Cisco Smart Software Manager users, the Strong Encryption license is automatically enabled for qualified customers when you apply the registration token on the Firepower 9300. **Note** If you are using the Smart Software Manager satellite deployment, to use ASDM and other strong encryption features, after you deploy the ASA you must enable the Strong Encryption (3DES) license using the ASA CLI. This feature requires FXOS 1.1.3. We removed the following command for non-satellite configurations: **feature strong-encryption**
Validation of the Smart Call Home/Smart Licensing certificate if the issuing hierarchy of the server certificate changes	9.5(2)	Smart licensing uses the Smart Call Home infrastructure. When the ASA first configures Smart Call Home anonymous reporting in the background, it automatically creates a trustpoint containing the certificate of the CA that issued the Smart Call Home server certificate. The ASA now supports validation of the certificate if the issuing hierarchy of the server certificate changes; you can enable the automatic update of the trustpool bundle at periodic intervals. We introduced the following command: **auto-import**
New Carrier license	9.5(2)	The new Carrier license replaces the existing GTP/GPRS license, and also includes support for SCTP and Diameter inspection. For the ASA on the Firepower 9300, the **feature mobile-sp** command will automatically migrate to the **feature carrier** command. We introduced or modified the following commands: **feature carrier, show activation-key, show license, show tech-support, show version**
Cisco Smart Software Licensing for the ASA on the Firepower 9300	9.4(1.150)	We introduced Smart Software Licensing for the ASA on the Firepower 9300. We introduced the following commands: **feature strong-encryption, feature mobile-sp, feature context**
Cisco Smart Software Licensing for the ASAv	9.3(2)	Smart Software Licensing lets you purchase and manage a pool of licenses. Unlike PAK licenses, smart licenses are not tied to a specific serial number. You can easily deploy or retire ASAv's without having to manage each unit’s license key. Smart Software Licensing also lets you see your license usage and needs at a glance. We introduced the following commands: **clear configure license, debug license agent, feature tier, http-proxy, license smart, license smart deregister, license smart register, license smart renew, show license, show running-config license, throughput level**



CHAPTER 5

Logical Devices for the Firepower 4100/9300

The Firepower 4100/9300 is a flexible security platform on which you can install one or more *logical devices*. This chapter describes basic interface configuration and how to add a standalone or High Availability logical device using the Firepower Chassis Manager. To add a clustered logical device, see ASA Cluster for the Firepower 4100/9300, on page 423. To use the FXOS CLI, see the FXOS CLI configuration guide. For more advanced FXOS procedures and troubleshooting, see the FXOS configuration guide.

- About Interfaces, on page 163
- About Logical Devices, on page 166
- Requirements and Prerequisites for Hardware and Software Combinations, on page 166
- Guidelines and Limitations for Logical Devices, on page 167
- Configure Interfaces, on page 168
- Configure Logical Devices, on page 174
- History for Logical Devices, on page 183

About Interfaces

The Firepower 4100/9300 chassis supports physical interfaces and EtherChannel (port-channel) interfaces. EtherChannel interfaces can include up to 16 member interfaces of the same type.

Chassis Management Interface

The chassis management interface is used for management of the FXOS Chassis by SSH or Firewall Chassis Manager. This interface is separate from the mgmt-type interface that you assign to the logical devices for application management.

To configure parameters for this interface, you must configure them from the CLI. To view information about this interface in the FXOS CLI, connect to local management and show the management port:

Firepower # **connect local-mgmt**

Firepower(local-mgmt) # **show mgmt-port**

Note that the chassis management interface remains up even if the physical cable or SFP module are unplugged, or if the **mgmt-port shut** command is performed, or if the logical device is offline.

Note The chassis management interface does not support jumbo frames.

Interface Types

Physical interfaces and EtherChannel (port-channel) interfaces can be one of the following types:

- Data—Use for regular data. Data interfaces cannot be shared between logical devices, and logical devices cannot communicate over the backplane to other logical devices. For traffic on Data interfaces, all traffic must exit the chassis on one interface and return on another interface to reach another logical device.
- Data-sharing—Use for regular data. Only supported with container instances, these data interfaces can be shared by one or more logical devices/container instances (Firewall Threat Defense-using-Firewall Management Center only).
- Mgmt—Use to manage application instances. These interfaces can be shared by one or more logical devices to access external hosts; logical devices cannot communicate over this interface with other logical devices that share the interface. You can only assign one management interface per logical device. Depending on your application and manager, you can later enable management from a data interface; but you must assign a Management interface to the logical device even if you don't intend to use it after you enable data management. For information about the separate chassis management interface, see Chassis Management Interface, on page 163.

Note Mgmt interface change will cause reboot of the logical device, for example one change mgmt from e1/1 to e1/2 will cause the logical device to reboot to apply the new management.

- Eventing—Use as a secondary management interface for Firewall Threat Defense-using-Firewall Management Center devices.

Note A virtual Ethernet interface is allocated when each application instance is installed. If the application does not use an eventing interface, then the virtual interface will be in an admin down state.

```
Firepower # show interface Vethernet775
Firepower # Vethernet775 is down (Administratively down)
Bound Interface is Ethernet1/10
Port description is server 1/1, VNIC ext-mgmt-nic5
```

- Cluster—Use as the cluster control link for a clustered logical device. By default, the cluster control link is automatically created on Port-channel 48. The Cluster type is only supported on EtherChannel interfaces.

See the following table for interface type support for the Firewall Threat Defense and ASA applications in standalone and cluster deployments.

Table 3: Interface Type Support

Application		Data	Data: Subinterface	Data-Sharing	Data-Sharing: Subinterface	Mgmt	Eventing	Cluster (EtherChannel only)	Cluster: Subinterface
Firewall Threat Defense	Standalone Native Instance	**Yes**	—	—	—	**Yes**	**Yes**	—	—
	Standalone Container Instance	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	—	—
	Cluster Native Instance	**Yes** (EtherChannel only for inter-chassis cluster)	—	—	—	**Yes**	**Yes**	**Yes**	—
	Cluster Container Instance	**Yes** (EtherChannel only for inter-chassis cluster)	—	—	—	**Yes**	**Yes**	**Yes**	**Yes**
ASA	Standalone Native Instance	**Yes**	—	—	—	**Yes**	—	**Yes**	—
	Cluster Native Instance	**Yes** (EtherChannel only for inter-chassis cluster)	—	—	—	**Yes**	—	**Yes**	—

FXOS Interfaces vs. Application Interfaces

The Firepower 4100/9300 manages the basic Ethernet settings of physical interfaces and EtherChannel (port-channel) interfaces. Within the application, you configure higher level settings. For example, you can only create EtherChannels in FXOS; but you can assign an IP address to the EtherChannel within the application.

The following sections describe the interaction between FXOS and the application for interfaces.

VLAN Subinterfaces

For all logical devices, you can create VLAN subinterfaces within the application.

Independent Interface States in the Chassis and in the Application

You can administratively enable and disable interfaces in both the chassis and in the application. For an interface to be operational, the interface must be enabled in both operating systems. Because the interface state is controlled independently, you may have a mismatch between the chassis and application.

About Logical Devices

A logical device lets you run one application instance (either ASA or Firepower Threat Defense) and also one optional decorator application (Radware DefensePro) to form a service chain.

When you add a logical device, you also define the application instance type and version, assign interfaces, and configure bootstrap settings that are pushed to the application configuration.



Note For the Firepower 9300, you can install different application types (ASA and Firepower Threat Defense) on separate modules in the chassis. You can also run different versions of an application instance type on separate modules.

Standalone and Clustered Logical Devices

You can add the following logical device types:

- Standalone—A standalone logical device operates as a standalone unit or as a unit in a High Availability pair.
- Cluster—A clustered logical device lets you group multiple units together, providing all the convenience of a single device (management, integration into a network) while achieving the increased throughput and redundancy of multiple devices. Multiple module devices, like the Firepower 9300, support intra-chassis clustering. For the Firepower 9300, all three modules must participate in the cluster, for both native and container instances.

Requirements and Prerequisites for Hardware and Software Combinations

The Firepower 4100/9300 supports multiple models, security modules, application types, and high availability and scalability features. See the following requirements for allowed combinations.

Firepower 9300 Requirements

The Firepower 9300 includes 3 security module slots and multiple types of security modules. See the following requirements:

- Security Module Types—You can install modules of different types in the Firepower 9300. For example, you can install the SM-48 as module 1, SM-40 as module 2, and SM-56 as module 3.
- Native and Container instances—When you install a container instance on a security module, that module can only support other container instances. A native instance uses all of the resources for a module, so

you can only install a single native instance on a module. You can use native instances on some modules, and container instances on the other module. For example, you can install a native instance on module 1 and module 2, but container instances on module 3.

- Clustering—All security modules in the cluster, whether it is intra-chassis or inter-chassis, must be the same type. You can have different quantities of installed security modules in each chassis, although all modules present in the chassis must belong to the cluster including any empty slots. For example, you can install 2 SM-40s in chassis 1, and 3 SM-40s in chassis 2. You cannot use clustering if you install 1 SM-48 and 2 SM-40s in the same chassis.
- High Availability—High Availability is only supported between same-type modules on the Firepower 9300. However, the two chassis can include mixed modules. For example, each chassis has an SM-40, SM-48, and SM-56. You can create High Availability pairs between the SM-40 modules, between the SM-48 modules, and between the SM-56 modules.
- ASA and Firewall Threat Defense application types—You can install different application types on separate modules in the chassis. For example, you can install ASA on module 1 and module 2, and Firewall Threat Defense on module 3.
- ASA or Firewall Threat Defense versions—You can run different versions of an application instance type on separate modules, or as separate container instances on the same module. For example, you can install the Firewall Threat Defense 6.3 on module 1, Firewall Threat Defense 6.4 on module 2, and Firewall Threat Defense 6.5 on module 3.

Firepower 4100 Requirements

The Firepower 4100 comes in multiple models. See the following requirements:

- Native and Container instances—When you install a container instance on a Firepower 4100, that device can only support other container instances. A native instance uses all of the resources for a device, so you can only install a single native instance on the device.
- Clustering—All chassis in the cluster must be the same model.
- High Availability—High Availability is only supported between same-type models.
- ASA and Firewall Threat Defense application types—The Firepower 4100 can only run a single application type.

Guidelines and Limitations for Logical Devices

See the following sections for guidelines and limitations.

Guidelines and Limitations for Interfaces

Default MAC Addresses

Default MAC address assignments depend on the type of interface.

- Physical interfaces—The physical interface uses the burned-in MAC address.

- EtherChannels—For an EtherChannel, all interfaces that are part of the channel group share the same MAC address. This feature makes the EtherChannel transparent to network applications and users, because they only see the one logical connection; they have no knowledge of the individual links. The port-channel interface uses a unique MAC address from a pool; interface membership does not affect the MAC address.

General Guidelines and Limitations

Firewall Mode

You can set the firewall mode to routed or transparent in the bootstrap configuration for the Firepower Threat Defense and ASA.

High Availability

- Configure high availability within the application configuration.
- You can use any data interfaces as the failover and state links. Data-sharing interfaces are not supported.

Context Mode

- Enable multiple context mode in the ASA after you deploy.

Requirements and Prerequisites for High Availability

- The two units in a High Availability Failover configuration must:
  - Be on a separate chassis; intra-chassis High Availability for the Firepower 9300 is not supported.
  - Be the same model.
  - Have the same interfaces assigned to the High Availability logical devices.
  - Have the same number and types of interfaces. All interfaces must be preconfigured in FXOS identically before you enable High Availability.
- High Availability is only supported between same-type modules on the Firepower 9300; but the two chassis can include mixed modules. For example, each chassis has an SM-56, SM-48, and SM-40. You can create High Availability pairs between the SM-56 modules, between the SM-48 modules, and between the SM-40 modules.
- For other High Availability system requirements, see Failover system requirements, on page 258.

Configure Interfaces

By default, physical interfaces are disabled. You can enable interfaces, add EtherChannels, and edit interface properties.

Note If you remove an interface in FXOS (for example, if you remove a network module, remove an EtherChannel, or reassign an interface to an EtherChannel), then the ASA configuration retains the original commands so that you can make any necessary adjustments; removing an interface from the configuration can have wide effects. You can manually remove the old interface configuration in the ASA OS.

Configure a Physical Interface

You can physically enable and disable interfaces, as well as set the interface speed and duplex. To use an interface, it must be physically enabled in FXOS and logically enabled in the application.

Note
- For QSFPH40G-CUxM, auto-negotiation is always enabled by default and you cannot disable it.
- If you replace an SFP on a port with a different SFP module, the speed, duplex, and auto-negotiation of the interface is not updated automatically. You must manually re-configure the interface.

Before you begin

- Interfaces that are already a member of an EtherChannel cannot be modified individually. Be sure to configure settings before you add it to the EtherChannel.

Procedure

Step 1 Enter interface mode.

scope eth-uplink

scope fabric a

Step 2 Enable the interface.

enter interface *interface_id*

enable

Example:

```
Firepower /eth-uplink/fabric # enter interface Ethernet1/8
Firepower /eth-uplink/fabric/interface # enable
```

Note
Interfaces that are already a member of a port-channel cannot be modified individually. If you use the **enter interface** or **scope interface** command on an interface that is a member of a port channel, you will receive an error stating that the object does not exist. You should edit interfaces using the **enter interface** command before you add them to a port-channel.

Step 3 (Optional) Set Debounce Time.

set debounce-time 5000 {Enter a value between 0-15000 milli-seconds}

Example:

```
Firepower /eth-uplink/fabric/interface # set debounce-time 5000
```

Example:

Note
Configuring Debounce Time is not supported on 1G interfaces.

Step 4 (Optional) Set the interface type.

set port-type {**data** | **mgmt** | **cluster**}

Example:

```
Firepower /eth-uplink/fabric/interface # set port-type mgmt
```

The **data** keyword is the default type. Do not choose the **cluster** keyword; by default, the cluster control link is automatically created on Port-channel 48.

Step 5 Enable or disable autonegotiation, if supported for your interface.

set auto-negotiation {**on** | **off**}

Example:

```
Firepower /eth-uplink/fabric/interface* # set auto-negotiation off
```

If a peer switch connecting to the port over a 50G cable does not support auto-negotiation, ensure to disable auto-negotiation on the switch and the platform interface as well. For example, N9K-C93400LD-H1 does not support auto-negotiation on a 50G cable. Hence, for the port to be connected you must disable the default auto-negotiation on the platform and the switch:

Example:

```
N5-50G-SW(config-if)# speed 50000
N5-50G-SW(config-if)# no negotiate auto
```

Step 6 Set the interface speed.

set admin-speed {**1gbps** | **10gbps** | **40gbps** | **100gbps**}

Example:

```
Firepower /eth-uplink/fabric/interface* # set admin-speed 1gbps
```

Step 7 Set the interface duplex mode.

set admin-duplex {**fullduplex** | **halfduplex**}

Example:

```
Firepower /eth-uplink/fabric/interface* # set admin-duplex halfduplex
```

Step 8 If you edited the default flow control policy, it is already applied to interfaces. If you created a new policy, apply it to the interface.

set flow-control-policy *name*

Example:

```
Firepower /eth-uplink/fabric/interface* # set flow-control-policy flow1
```

Step 9 Save the configuration.

commit-buffer

Example:

```
Firepower /eth-uplink/fabric/interface* # commit-buffer
Firepower /eth-uplink/fabric/interface #
```

Add an EtherChannel (Port Channel)

An EtherChannel (also known as a port channel) can include up to 16 member interfaces of the same media type and capacity, and must be set to the same speed and duplex. The media type can be either RJ-45 or SFP; SFPs of different types (copper and fiber) can be mixed. You cannot mix interface capacities (for example 1GB and 10GB interfaces) by setting the speed to be lower on the larger-capacity interface. The Link Aggregation Control Protocol (LACP) aggregates interfaces by exchanging the Link Aggregation Control Protocol Data Units (LACPDUs) between two network devices.

You can configure each physical Data interface in an EtherChannel to be:

- Active—Sends and receives LACP updates. An active EtherChannel can establish connectivity with either an active or a passive EtherChannel. You should use the active mode unless you need to minimize the amount of LACP traffic.
- On—The EtherChannel is always on, and LACP is not used. An “on” EtherChannel can only establish a connection with another “on” EtherChannel.



Note It may take up to three minutes for an EtherChannel to come up to an operational state if you change its mode from On to Active or from Active to On.

Non-data interfaces only support active mode.

LACP coordinates the automatic addition and deletion of links to the EtherChannel without user intervention. It also handles misconfigurations and checks that both ends of member interfaces are connected to the correct channel group. “On” mode cannot use standby interfaces in the channel group when an interface goes down, and the connectivity and configurations are not checked.

When the Firepower 4100/9300 creates an EtherChannel, the EtherChannel stays in a **Suspended** state for Active LACP mode or a **Down** state for On LACP mode until you assign it to a logical device, even if the physical link is up. The EtherChannel will be brought out of this **Suspended** state in the following situations:

- The EtherChannel is added as a data or management interface for a standalone logical device
- The EtherChannel is added as a management interface or cluster control link for a logical device that is part of a cluster
- The EtherChannel is added as a data interface for a logical device that is part of a cluster and at least one unit has joined the cluster

Note that the EtherChannel does not come up until you assign it to a logical device. If the EtherChannel is removed from the logical device or the logical device is deleted, the EtherChannel will revert to a **Suspended** or **Down** state.

Procedure

Step 1 Enter interface mode:

scope eth-uplink

scope fabric a

Step 2 Create the port-channel:

create port-channel *id*

enable

Step 3 Assign member interfaces:

create member-port *interface_id*

You can add up to 16 member interfaces of the same media type and capacity. The member interfaces must be set to the same speed and duplex, and must match the speed and duplex that you configured for this port channel. The media type can be either RJ-45 or SFP; SFPs of different types (copper and fiber) can be mixed. You cannot mix interface capacities (for example 1GB and 10GB interfaces) by setting the speed to be lower on the larger-capacity interface.

Example:

```
Firepower /eth-uplink/fabric/port-channel* # create member-port Ethernet1/1
Firepower /eth-uplink/fabric/port-channel/member-port* # exit
Firepower /eth-uplink/fabric/port-channel* # create member-port Ethernet1/2
Firepower /eth-uplink/fabric/port-channel/member-port* # exit
Firepower /eth-uplink/fabric/port-channel* # create member-port Ethernet1/3
Firepower /eth-uplink/fabric/port-channel/member-port* # exit
Firepower /eth-uplink/fabric/port-channel* # create member-port Ethernet1/4
Firepower /eth-uplink/fabric/port-channel/member-port* # exit
```

Step 4 (Optional) Set the interface type.

set port-type {**data** | **mgmt** | **cluster**}

Example:

```
Firepower /eth-uplink/fabric/port-channel # set port-type data
```

The **data** keyword is the default type. Do not choose the **cluster** keyword unless you want to use this port-channel as the cluster control link instead of the default.

Step 5 Set the required interface speed for members of the port-channel.

set speed {**10mbps** | **100mbps** | **1gbps** | **10gbps** | **40gbps** | **100gbps**}

If you add a member interface that is not at the specified speed, it will not successfully join the port channel. The default is **10gbps**.

Example:

```
Firepower /eth-uplink/fabric/port-channel* # set speed 1gbps
```

Step 6 (Optional) Set the required duplex for members of the port-channel.

set duplex {**fullduplex** | **halfduplex**}

If you add a member interface that is configured with the specified duplex, it will not successfully join the port channel. The default is **fullduplex**.

Example:

```
Firepower /eth-uplink/fabric/port-channel* # set duplex fullduplex
```

Step 7 Enable or disable autonegotiation, if supported for your interface.

set auto-negotiation {**on** | **off**}

Example:

```
Firepower /eth-uplink/fabric/interface* # set auto-negotiation off
```

Step 8 Set the LACP port-channel mode for data interfaces.

For non-Data interfaces, the mode is always active.

set port-channel-mode {**active** | **on**}

Example:

```
Firepower /eth-uplink/fabric/port-channel* # set port-channel-mode on
```

Step 9 If you edited the default flow control policy, it is already applied to interfaces. If you created a new policy, apply it to the interface.

set flow-control-policy *name*

Example:

```
Firepower /eth-uplink/fabric/interface* # set flow-control-policy flow1
```

Step 10 Commit the configuration:

commit-buffer

Configure Logical Devices

Add a standalone logical device or a High Availability pair on the Firepower 4100/9300 chassis.

For clustering, see #unique_219.

Add a Standalone ASA

Standalone logical devices work either alone or in a High Availability pair. On the Firepower 9300 with multiple security modules, you can deploy either a cluster or standalone devices. The cluster must use all modules, so you cannot mix and match a 2-module cluster plus a single standalone device, for example.

You can deploy a routed or transparent firewall mode ASA from the Firepower 4100/9300 chassis.

For multiple context mode, you must first deploy the logical device, and then enable multiple context mode in the ASA application.

Before you begin

- Download the application image you want to use for the logical device from Cisco.com, and then download that image to the Firepower 4100/9300.

Note For the Firepower 9300, you can install different application types (ASA and Firewall Threat Defense) on separate modules in the chassis. You can also run different versions of an application instance type on separate modules.

- Configure a management interface to use with the logical device. The management interface is required. Note that this management interface is not the same as the chassis management port that is used only for chassis management (in FXOS, you might see it displayed as MGMT, management0, or other similar names).
- Gather the following information:
  - Interface IDs for this device
  - Management interface IP address and network mask
  - Gateway IP address

Procedure

Step 1 Enter security services mode.

scope ssa

Example:

```
Firepower# scope ssa
Firepower /ssa #
```

Step 2 Set the application instance image version.

a) View available images. Note the Version number that you want to use.

show app

Example:

```
Firepower /ssa # show app
    Name       Version         Author     Supported Deploy Types CSP Type    Is Default
 App
    ---------- --------------- ---------- ---------------------- -----------
--------------
    asa        9.9.1           cisco      Native                 Application No
    asa        9.10.1          cisco      Native                 Application Yes
    ftd        6.2.3           cisco      Native                 Application Yes
```

b) Set the scope to the security module/engine slot.

scope slot *slot_id*

The *slot_id* is always 1 for the Firepower 4100, and 1, 2, or 3 for the Firepower 9300.

Example:

```
Firepower /ssa # scope slot 1
Firepower /ssa/slot #
```

c) Create the application instance.

enter app-instance asa *device_name*

The *device_name* can be between 1 and 64 characters. You will use this device name when you create the logical device for this instance.

Example:

```
Firepower /ssa/slot # enter app-instance asa ASA1
Firepower /ssa/slot/app-instance* #
```

d) Set the ASA image version.

set startup-version *version*

Example:

```
Firepower /ssa/slot/app-instance* # set startup-version 9.10.1
```

e) Exit to slot mode.

exit

Example:

```
Firepower /ssa/slot/app-instance* # exit
Firepower /ssa/slot* #
```

f) Exit to ssa mode.

exit

Example:

```
Firepower /ssa/slot* # exit
Firepower /ssa* #
```

Example:

```
Firepower /ssa # scope slot 1
Firepower /ssa/slot # enter app-instance asa ASA1
Firepower /ssa/slot/app-instance* # set startup-version 9.10.1
Firepower /ssa/slot/app-instance* # exit
Firepower /ssa/slot* # exit
Firepower /ssa* #
```

Step 3 Create the logical device.

enter logical-device *device_name* **asa** *slot_id* **standalone**

Use the same *device_name* as the application instance you added earlier.

Example:

```
Firepower /ssa # enter logical-device ASA1 asa 1 standalone
Firepower /ssa/logical-device* #
```

Step 4 Assign the management and data interfaces to the logical device. Repeat for each interface.

create external-port-link *name interface_id* **asa**

set description *description*

exit

- *name*—The name is used by the Firepower 4100/9300 chassis supervisor; it is not the interface name used in the ASA configuration.
- *description*—Use quotes (") around phrases with spaces.

The management interface is not the same as the chassis management port. You will later enable and configure the data interfaces on the ASA, including setting the IP addresses.

Example:

```
Firepower /ssa/logical-device* # create external-port-link inside Ethernet1/1 asa
Firepower /ssa/logical-device/external-port-link* # set description "inside link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create external-port-link management Ethernet1/7 asa
Firepower /ssa/logical-device/external-port-link* # set description "management link"
Firepower /ssa/logical-device/external-port-link* # exit
```

```
Firepower /ssa/logical-device* # create external-port-link outside Ethernet1/2 asa
Firepower /ssa/logical-device/external-port-link* # set description "external link"
Firepower /ssa/logical-device/external-port-link* # exit
```

Step 5 Configure the management bootstrap information.

a) Create the bootstrap object.

create mgmt-bootstrap asa

Example:

```
Firepower /ssa/logical-device* # create mgmt-bootstrap asa
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

b) Specify the firewall mode, routed or transparent.

create bootstrap-key FIREWALL_MODE

set value {**routed** | **transparent**}

exit

In routed mode, the device is considered to be a router hop in the network. Each interface that you want to route between is on a different subnet. A transparent firewall, on the other hand, is a Layer 2 firewall that acts like a “bump in the wire,” or a “stealth firewall,” and is not seen as a router hop to connected devices.

The firewall mode is only set at initial deployment. If you re-apply the bootstrap settings, this setting is not used.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key FIREWALL_MODE
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value routed
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

c) Specify the admin and enable password.

create bootstrap-key-secret PASSWORD

set value

Enter a value: *password*

Confirm the value: *password*

exit

Example:

The pre-configured ASA admin user and enable password is useful for password recovery; if you have FXOS access, you can reset the admin user password if you forget it.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key-secret PASSWORD
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
Enter a value: floppylampshade
Confirm the value: floppylampshade
```

```
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

d) Configure the IPv4 management interface settings.

create ipv4 *slot_id* **default**

set ip *ip_address* **mask** *network_mask*

set gateway *gateway_address*

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv4 1 default
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set ip 10.10.10.34 mask 255.255.255.0
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set gateway 10.10.10.1
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

e) Configure the IPv6 management interface settings.

create ipv6 *slot_id* **default**

set ip *ip_address* **prefix-length** *prefix*

set gateway *gateway_address*

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv6 1 default
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set ip 2001:0DB8:BA98::3210
prefix-length 64
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # set gateway 2001:0DB8:BA98::3211
Firepower /ssa/logical-device/mgmt-bootstrap/ipv6* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

f) Exit the management bootstrap mode.

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # exit
Firepower /ssa/logical-device* #
```

Step 6 Save the configuration.

commit-buffer

The chassis deploys the logical device by downloading the specified software version and pushing the bootstrap configuration and management interface settings to the application instance. Check the status of the deployment using the **show app-instance** command. The application instance is running and ready to use when the **Admin State** is **Enabled** and the **Oper State** is **Online**.

Example:

```
Firepower /ssa/logical-device* # commit-buffer
Firepower /ssa/logical-device # exit
Firepower /ssa # show app-instance
App Name   Identifier Slot ID    Admin State Oper State      Running Version Startup Version
 Deploy Type Profile Name Cluster State   Cluster Role
---------- ---------- ---------- ----------- --------------- -------------- ---------------
 ----------- ------------ --------------- ------------
asa        asa1       2          Disabled    Not Installed                  9.12.1
    Native                   Not Applicable  None
ftd        ftd1       1          Enabled     Online          6.4.0.49       6.4.0.49
    Container   Default-Small Not Applicable  None
```

Step 7 See the ASA configuration guide to start configuring your security policy.

Example

```
Firepower# scope ssa
Firepower /ssa # scope slot 1
Firepower /ssa/slot # enter app-instance asa MyDevice1
Firepower /ssa/slot/app-instance* # set startup-version 9.10.1
Firepower /ssa/slot/app-instance* # exit
Firepower /ssa/slot* # exit
Firepower /ssa* # create logical-device MyDevice1 asa 1 standalone
Firepower /ssa/logical-device* # create external-port-link inside Ethernet1/1 asa
Firepower /ssa/logical-device/external-port-link* # set description "inside link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create external-port-link management Ethernet1/7 asa
Firepower /ssa/logical-device/external-port-link* # set description "management link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create external-port-link outside Ethernet1/2 asa
Firepower /ssa/logical-device/external-port-link* # set description "external link"
Firepower /ssa/logical-device/external-port-link* # exit
Firepower /ssa/logical-device* # create mgmt-bootstrap asa
Firepower /ssa/logical-device/mgmt-bootstrap* # enter bootstrap-key FIREWALL_MODE
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value transparent
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key-secret PASSWORD
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
Enter a value: secretglassine
Confirm the value: secretglassine
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* # create ipv4 1 default
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set gateway 10.0.0.1
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # set ip 10.0.0.31 mask 255.255.255.0
Firepower /ssa/logical-device/mgmt-bootstrap/ipv4* # exit
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # commit-buffer
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key #
```

Add a High Availability Pair

Firewall Threat Defense or ASA High Availability (also known as failover) is configured within the application, not in FXOS. However, to prepare your chassis for high availability, see the following steps.

Before you begin

See Failover system requirements, on page 258.

Procedure

Step 1 Allocate the same interfaces to each logical device.

Step 2 Allocate 1 or 2 data interfaces for the failover and state link(s).

These interfaces exchange high availability traffic between the 2 chassis. We recommend that you use a 10 GB data interface for a combined failover and state link. If you have available interfaces, you can use separate failover and state links; the state link requires the most bandwidth. You cannot use the management-type interface for the failover or state link. We recommend that you use a switch between the chassis, with no other device on the same network segment as the failover interfaces.

Step 3 Enable High Availability on the logical devices. See Failover for High Availability, on page 257.

Step 4 If you need to make interface changes after you enable High Availability, perform the changes on the standby unit first, and then perform the changes on the active unit.

Note
For the ASA, if you remove an interface in FXOS (for example, if you remove a network module, remove an EtherChannel, or reassign an interface to an EtherChannel), then the ASA configuration retains the original commands so that you can make any necessary adjustments; removing an interface from the configuration can have wide effects. You can manually remove the old interface configuration in the ASA OS.

Change an Interface on an ASA Logical Device

You can allocate, unallocate, or replace a management interface on an ASA logical device. ASDM discovers the new interfaces automatically.

Adding a new interface, or deleting an unused interface has minimal impact on the ASA configuration. However, if you remove an allocated interface in FXOS (for example, if you remove a network module, remove an EtherChannel, or reassign an allocated interface to an EtherChannel), and the interface is used in your security policy, removal will impact the ASA configuration. In this case, the ASA configuration retains the original commands so that you can make any necessary adjustments. You can manually remove the old interface configuration in the ASA OS.

Note You can edit the membership of an allocated EtherChannel without impacting the logical device.

Before you begin

- Configure your interfaces and add any EtherChannels according to Configure a Physical Interface, on page 169 and Add an EtherChannel (Port Channel), on page 171.
- If you want to add an already-allocated interface to an EtherChannel (for example, all interfaces are allocated by default to a cluster), you need to unallocate the interface from the logical device first, then

add the interface to the EtherChannel. For a new EtherChannel, you can then allocate the EtherChannel to the device.

- For clustering or failover, make sure you add or remove the interface on all units. We recommend that you make the interface changes on the data/standby unit(s) first, and then on the control/active unit. New interfaces are added in an administratively down state, so they do not affect interface monitoring.

Procedure

Step 1 Enter security services mode:

Firepower# **scope ssa**

Step 2 Edit the logical device:

Firepower /ssa # **scope logical-device** *device_name*

Step 3 Unallocate an interface from the logical device:

Firepower /ssa/logical-device # **delete external-port-link** *name*

Enter the **show external-port-link** command to view interface names.

For a management interface, delete the current interface then commit your change using the **commit-buffer** command before you add the new management interface.

Step 4 Allocate a new interface to the logical device:

Firepower /ssa/logical-device* # **create external-port-link** *name interface_id* **asa**

Step 5 Commit the configuration:

commit-buffer

Commits the transaction to the system configuration.

Connect to the Console of the Application

Use the following procedure to connect to the console of the application.

Procedure

Step 1 Connect to the module CLI using a console connection or a Telnet connection.

connect module *slot_number* {**console** | **telnet**}

To connect to the security engine of a device that does not support multiple security modules, always use **1** as the *slot_number*.

The benefits of using a Telnet connection is that you can have multiple sessions to the module at the same time, and the connection speed is faster.

Example:

```
Firepower# connect module 1 console
Telnet escape character is '~'.
Trying 127.5.1.1...
Connected to 127.5.1.1.
Escape character is '~'.

CISCO Serial Over LAN:
Close Network Connection to Exit

Firepower-module1>
```

Step 2 Connect to the application console.

connect asa *name*

To view the instance names, enter the command without a name.

Example:

```
Firepower-module1> connect asa asa1
Connecting to asa(asa1) console... hit Ctrl + A + D  to return to bootCLI
[...]
asa>
```

Step 3 Exit the application console to the FXOS module CLI.

- ASA—Enter **Ctrl-a, d**

Step 4 Return to the supervisor level of the FXOS CLI.

Exit the console:

a) Enter **~**

You exit to the Telnet application.

b) To exit the Telnet application, enter:

telnet>**quit**

Exit the Telnet session:

a) Enter **Ctrl-], .**

Example

The following example connects to an ASA on security module 1 and then exits back to the supervisor level of the FXOS CLI.

```
Firepower# connect module 1 console
Telnet escape character is '~'.
Trying 127.5.1.1...
Connected to 127.5.1.1.
Escape character is '~'.
```

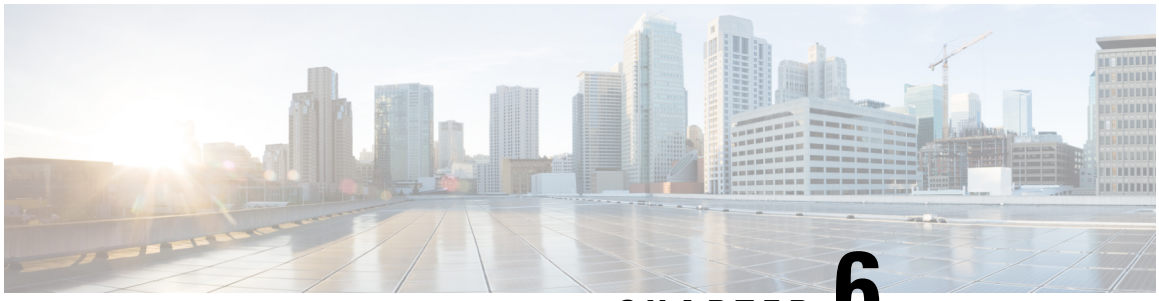
```
CISCO Serial Over LAN:
Close Network Connection to Exit

Firepower-module1>connect asa asa1
asa> ~
telnet> quit
Connection closed.
Firepower#
```

History for Logical Devices

Feature	Version	Details
ASA for the Firepower 4112	9.14(1)	We introduced the Firepower 4112. **Note** Requires FXOS 2.8.1.
Firepower 9300 SM-56 support	9.12.2	We introduced the SM-56 security module. **Note** Requires FXOS 2.6.1.157.
ASA for the Firepower 4115, 4125, and 4145	9.12(1)	We introduced the Firepower 4115, 4125, and 4145. **Note** Requires FXOS 2.6.1.
Firepower 9300 SM-40 and SM-48 support	9.12.1	We introduced the SM-40 and SM-48 security modules. **Note** Requires FXOS 2.6.1.
Support for ASA and Firepower Threat Defense on separate modules of the same Firepower 9300	9.12.1	You can now deploy ASA and Firepower Threat Defense logical devices on the same Firepower 930 **Note** Requires FXOS 2.6.1.
Cluster control link customizable IP Address for the Firepower 4100/9300	9.10.1	By default, the cluster control link uses the 127.2.0.0/16 network. You can now set the network whe you deploy the cluster in FXOS. The chassis auto-generates the cluster control link interface IP addres for each unit based on the chassis ID and slot ID: 127.2.*chassis_id*.*slot_id*. However, some networkin deployments do not allow 127.2.0.0/16 traffic to pass. Therefore, you can now set a custom /16 subn for the cluster control link in FXOS except for loopback (127.0.0.0/8) and multicast (224.0.0.0/4) addresses. **Note** Requires FXOS 2.4.1. New/Modified FXOS commands: **set cluster-control-link network**

Feature	Version	Details
Support for data EtherChannels in On mode	9.10.1	You can now set data and data-sharing EtherChannels to either Active LACP mode or to On mode. Other types of EtherChannels only support Active mode. **Note** Requires FXOS 2.4.1. New/Modified FXOS commands: **set port-channel-mode**
Inter-site clustering improvement for the ASA on the Firepower 4100/9300 chassis	9.7(1)	You can now configure the site ID for each Firepower 4100/9300 chassis when you deploy the ASA cluster. Previously, you had to configure the site ID within the ASA application; this new feature eases initial deployment. Note that you can no longer set the site ID within the ASA configuration. Also, for best compatibility with inter-site clustering, we recommend that you upgrade to ASA 9.7(1) and FXOS 2.1.1, which includes several improvements to stability and performance. We modified the following command: **site-id**
Support for the Firepower 4100 series	9.6(1)	With FXOS 1.1.4, the ASA supports inter-chassis clustering on the Firepower 4100 series. We did not modify any commands.
Inter-chassis clustering for 6 modules, and inter-site clustering for the Firepower 9300 ASA application	9.5(2.1)	With FXOS 1.1.3, you can now enable inter-chassis, and by extension inter-site clustering. You can include up to 6 modules in up to 6 chassis. We did not modify any commands.
Intra-chassis ASA Clustering for the Firepower 9300	9.4(1.150)	You can cluster up to 3 security modules within the Firepower 9300 chassis. All modules in the chassis must belong to the cluster. We introduced the following commands: **cluster replication delay, debug service-module, management-only individual, show cluster chassis**



CHAPTER 6

Transparent or Routed Firewall Mode

This chapter describes how to set the firewall mode to routed or transparent, as well as how the firewall works in each firewall mode.

You can set the firewall mode independently for each context in multiple context mode.

- Firewall mode, on page 185
- Default settings, on page 194
- Guidelines for firewall mode, on page 194
- Set the Firewall Mode, on page 195
- Examples for Firewall Mode, on page 196
- History for the Firewall Mode, on page 207

Firewall mode

A ASA device supports two firewall modes :

- Routed Firewall mode
- Transparent Firewall mode

Routed firewall mode

In routed mode, the ASA is considered to be a router hop in the network. Each interface that you want to route between is on a different subnet. You can share Layer 3 interfaces between contexts.

Integrated routing and bridging features

With Integrated Routing and Bridging, you can use a "bridge group" where you group together multiple interfaces on a network, and the ASA uses bridging techniques to pass traffic between the interfaces. Each bridge group includes a Bridge Virtual Interface (BVI) to which you assign an IP address on the network. The ASA routes between BVIs and regular routed interfaces. If you do not need multiple context mode or clustering or EtherChannel or VNI member interfaces, you might consider using routed mode instead of transparent mode. In routed mode, you can have one or more isolated bridge groups like in transparent mode, but also have normal routed interfaces as well for a mixed deployment.

Transparent firewall mode

Traditionally, a firewall is a routed hop and acts as a default gateway for hosts that connect to one of its screened subnets. A transparent firewall, on the other hand, is a Layer 2 firewall that acts like a “bump in the wire,” or a “stealth firewall,” and is not seen as a router hop to connected devices. However, like any other firewall, access control between interfaces is controlled, and all of the usual firewall checks are in place.

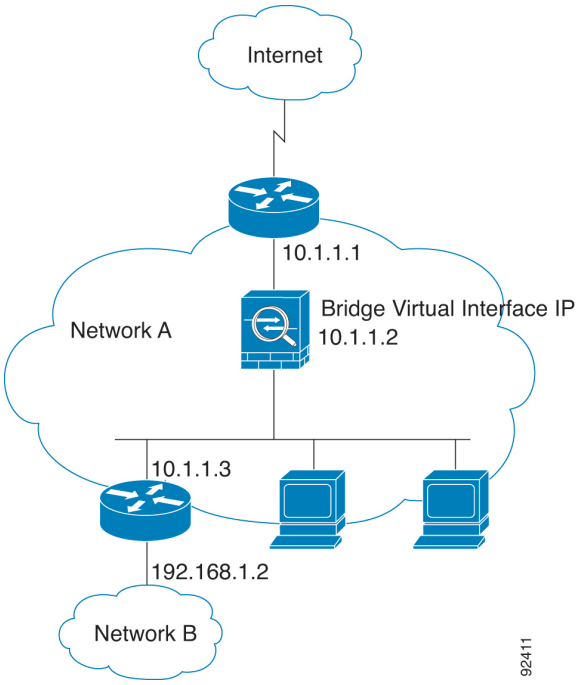
Layer 2 connectivity is achieved by using a "bridge group" where you group together the inside and outside interfaces for a network, and the ASA uses bridging techniques to pass traffic between the interfaces. Each bridge group includes a Bridge Virtual Interface (BVI) to which you assign an IP address on the network. You can have multiple bridge groups for multiple networks. In transparent mode, these bridge groups cannot communicate with each other.

Transparent firewall network usage

The ASA connects the same network between its interfaces. Because the firewall is not a routed hop, you can easily introduce a transparent firewall into an existing network.

The figure shows a typical transparent firewall network where the outside devices are on the same subnet as the inside devices. The inside router and hosts appear to be directly connected to the outside router.

Figure 19: Transparent firewall network



Management Interface

In addition to each Bridge Virtual Interface (BVI) IP address, you can add a separate Management *slot*/*port* interface that is not part of any bridge group, and that allows only management traffic to the ASA. For more information, see Management Interface, on page 594.

Passing traffic for routed-mode features

For features that are not directly supported on the transparent firewall, you can allow traffic to pass through so that upstream and downstream routers can support the functionality.

Supported traffic types

For example, by using an access rule, you can allow DHCP traffic (instead of the unsupported DHCP relay feature) or multicast traffic such as that created by IP/TV. You can also establish routing protocol adjacencies through a transparent firewall; you can allow OSPF, RIP, EIGRP, or BGP traffic through based on an access rule. Likewise, protocols like HSRP or VRRP can pass through the ASA.

Bridge groups

A bridge group is a group of interfaces that the ASA bridges instead of routes. Bridge groups are supported in both transparent and routed firewall mode. Like any other firewall interfaces, access control between interfaces is controlled, and all of the usual firewall checks are in place.

Bridge virtual interfaces

Each bridge group includes a Bridge Virtual Interface (BVI). The ASA uses the BVI IP address as the source address for packets originating from the bridge group. The BVI IP address must be on the same subnet as the bridge group member interfaces. The BVI does not support traffic on secondary networks; only traffic on the same network as the BVI IP address is supported.

BVI functionality by mode

In transparent mode: Only bridge group member interfaces are named and can be used with interface-based features.

In routed mode: The BVI acts as the gateway between the bridge group and other routed interfaces. To route between bridge groups/routed interfaces, you must name the BVI. For some interface-based features, you can use the BVI itself:

- Access rules—You can configure access rules for both bridge group member interfaces and for the BVI; for inbound rules, the member interface is checked first. For outbound rules, the BVI is checked first.
- DHCPv4 server—Only the BVI supports the DHCPv4 server configuration.
- Static routes—You can configure static routes for the BVI; you cannot configure static routes for the member interfaces.
- Syslog server and other traffic sourced from the ASA—When specifying a syslog server (or SNMP server, or other service where the traffic is sourced from the ASA), you can specify either the BVI or a member interface.

If you do not name the BVI in routed mode, then the ASA does not route bridge group traffic. This configuration replicates transparent firewall mode for the bridge group. If you do not need multiple context mode or clustering or EtherChannel or VNI member interfaces, you might consider using routed mode instead. In routed mode, you can have one or more isolated bridge groups like in transparent mode, but also have normal routed interfaces as well for a mixed deployment.

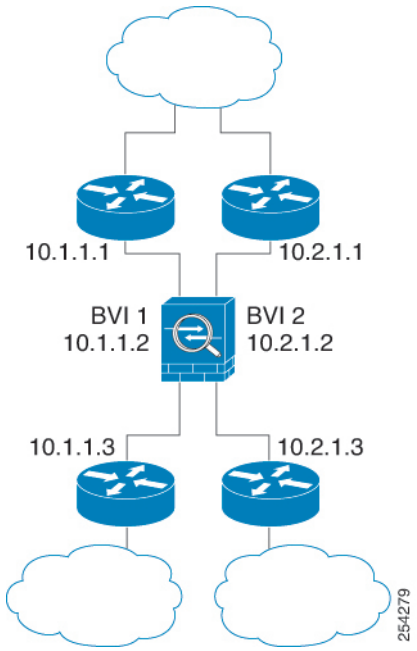
Bridge groups in transparent firewall mode

Bridge group traffic is isolated from other bridge groups; traffic is not routed to another bridge group within the ASA , and traffic must exit the ASA before it is routed by an external router back to another bridge group in the ASA. Although the bridging functions are separate for each bridge group, many other functions are shared between all bridge groups. For example, all bridge groups share a syslog server or AAA server configuration. For complete security policy separation, use security contexts with one bridge group in each context.

You can include multiple interfaces per bridge group. See Guidelines for firewall mode, on page 194 for the exact number of bridge groups and interfaces supported. If you use more than 2 interfaces per bridge group, you can control communication between multiple segments on the same network, and not just between inside and outside. For example, if you have three inside segments that you do not want to communicate with each other, you can put each segment on a separate interface, and only allow them to communicate with the outside interface. Or you can customize the access rules between interfaces to allow only as much access as desired.

This figure shows two networks connected to the ASA, which has two bridge groups.

Figure 20: Transparent firewall network with two bridge groups



Bridge groups in routed firewall mode

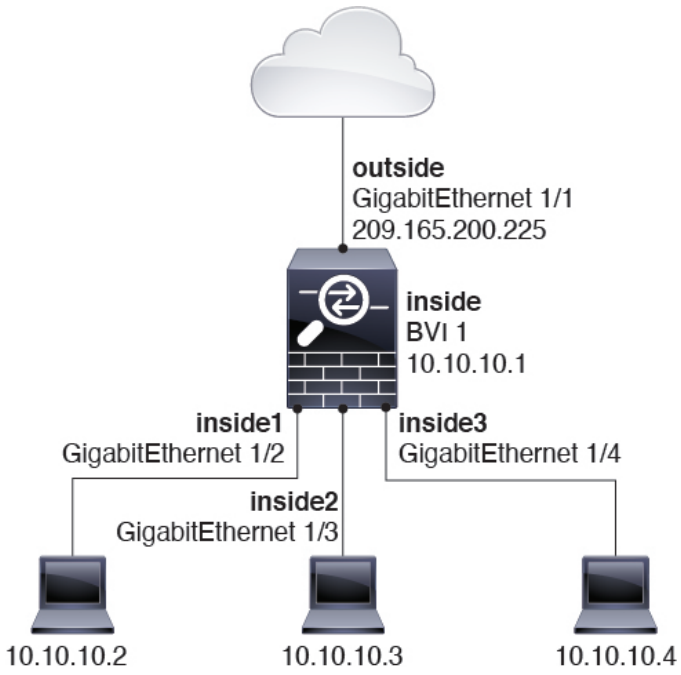
Bridge group traffic can be routed to other bridge groups or routed interfaces. You can choose to isolate bridge group traffic by not assigning a name to the BVI interface for the bridge group. If you name the BVI, then the BVI participates in routing like any other regular interface.

Bridge group usage and configuration

One use for a bridge group in routed mode is to use extra interfaces on the ASA instead of an external switch. For example, the default configuration for some devices include an outside interface as a regular interface, and then all other interfaces assigned to the inside bridge group. Because the purpose of this bridge group is to replace an external switch, you need to configure an access policy so all bridge group interfaces can freely

communicate. For example, as in the default configuration, set all the interfaces to the same security level, and then enable same-security interface communication; no access rule is required.

Figure 21: Routed firewall network with an inside bridge group and an outside routed interface



Passing Traffic Not Allowed in Routed Mode

In routed mode, some types of traffic cannot pass through the ASA even if you allow it in an access rule. The bridge group, however, can allow almost any traffic through using either an access rule (for IP traffic) or an EtherType rule (for non-IP traffic):

- IP traffic—In routed firewall mode, broadcast and multicast traffic is blocked even if you allow it in an access rule, including unsupported dynamic routing protocols and DHCP (unless you configure DHCP relay). Within a bridge group, you can allow this traffic with an access rule (using an extended ACL).
- Non-IP traffic—AppleTalk, IPX, BPDUs, and MPLS, for example, can be configured to go through using an EtherType rule.

Note The bridge group does not pass CDP packets packets, or any packets that do not have a valid EtherType greater than or equal to 0x600. An exception is made for BPDUs and IS-IS, which are supported.

Allow layer 3 traffic

This topic describes how different types of Layer 3 traffic are handled through bridge groups and the access rule requirements for each traffic type.

- Unicast IPv4 and IPv6 traffic is allowed through the bridge group automatically from a higher security interface to a lower security interface, without an access rule.

- For Layer 3 traffic traveling from a low to a high security interface, an access rule is required on the low security interface.
- ARPs are allowed through the bridge group in both directions without an access rule. ARP traffic can be controlled by ARP inspection.
- IPv6 neighbor discovery and router solicitation packets can be passed using access rules.
- Broadcast and multicast traffic can be passed using access rules.

Allowed MAC addresses

These destination MAC addresses are allowed through the bridge group if allowed by your access policy (For more information, refer to Allow layer 3 traffic, on page 189). Any MAC address not on this list is dropped.

- TRUE broadcast destination MAC address equal to FFFF.FFFF.FFFF
- IPv4 multicast MAC addresses from 0100.5E00.0000 to 0100.5EFE.FFFF
- IPv6 multicast MAC addresses from 3333.0000.0000 to 3333.FFFF.FFFF
- BPDU multicast address equal to 0100.0CCC.CCCD
- AppleTalk multicast MAC addresses from 0900.0700.0000 to 0900.07FF.FFFF

BPDU Handling

To prevent loops using the Spanning Tree Protocol, BPDUs are passed by default. To block BPDUs, you need to configure an EtherType rule to deny them. You can also block BPDUs on the external switches. For example, you can block BPDUs on the switch if members of the same bridge group are connected to switch ports in different VLANs. In this case, BPDUs from one VLAN will be visible in the other VLAN, which can cause Spanning Tree Root Bridge election process problems.

If you are using failover, you might want to block BPDUs to prevent the switch port from going into a blocking state when the topology changes. See Bridge Group Requirements for Failover, on page 268 for more information.

MAC address and route lookups

For traffic within a bridge group, the outgoing interface of a packet is determined by performing a destination MAC address lookup instead of a route lookup.

Route lookups, however, are necessary for these situations:

- Traffic originating on the ASA—Add a default/static route on the ASA for traffic destined for a remote network where a syslog server, for example, is located.
- SSL policy enabled in transparent firewall mode—
  - TLS 1.3 traffic passes through the ASA device.
  - ASA acts as a TCP proxy. It terminates the client-side TCP connection and initiates a separate server-side connection. The ASA TCP stack owns the session state.
  - For idle proxied sessions, ASA generates TCP ACK keepalive probes to preserve the connection state. These are not passthrough packets—the firewall originates them and sends them toward the client or server depending on the state of the TLS transit connection.

- Because the ASA originates these keepalives, it must perform an adjacency table lookup (next-hop or egress MAC resolution) even in transparent mode.
- If the adjacency lookup fails for one of these proxy-generated packets, the ASA logs syslog 110003.

  To prevent these syslog messages, configure routes on the ASA for the transit connection endpoints (client and server IP addresses). Proper routing enables successful adjacency lookups for keepalive packets.

  If the ASA successfully sends a TCP keepalive and receives a reply, it keeps the transit connection active. If the device cannot send the keepalives, or receives no reply, it removes the transit connection after approximately 240 seconds.

- Voice over IP (VoIP) and TFTP traffic with inspection enabled, and the endpoint is at least one hop away—Add a static route on the ASA for traffic destined for the remote endpoint so that secondary connections are successful. The ASA creates a temporary "pinhole" in the access control policy to allow the secondary connection; and because the connection might use a different set of IP addresses than the primary connection, the ASA needs to perform a route lookup to install the pinhole on the correct interface.

  Affected applications include:

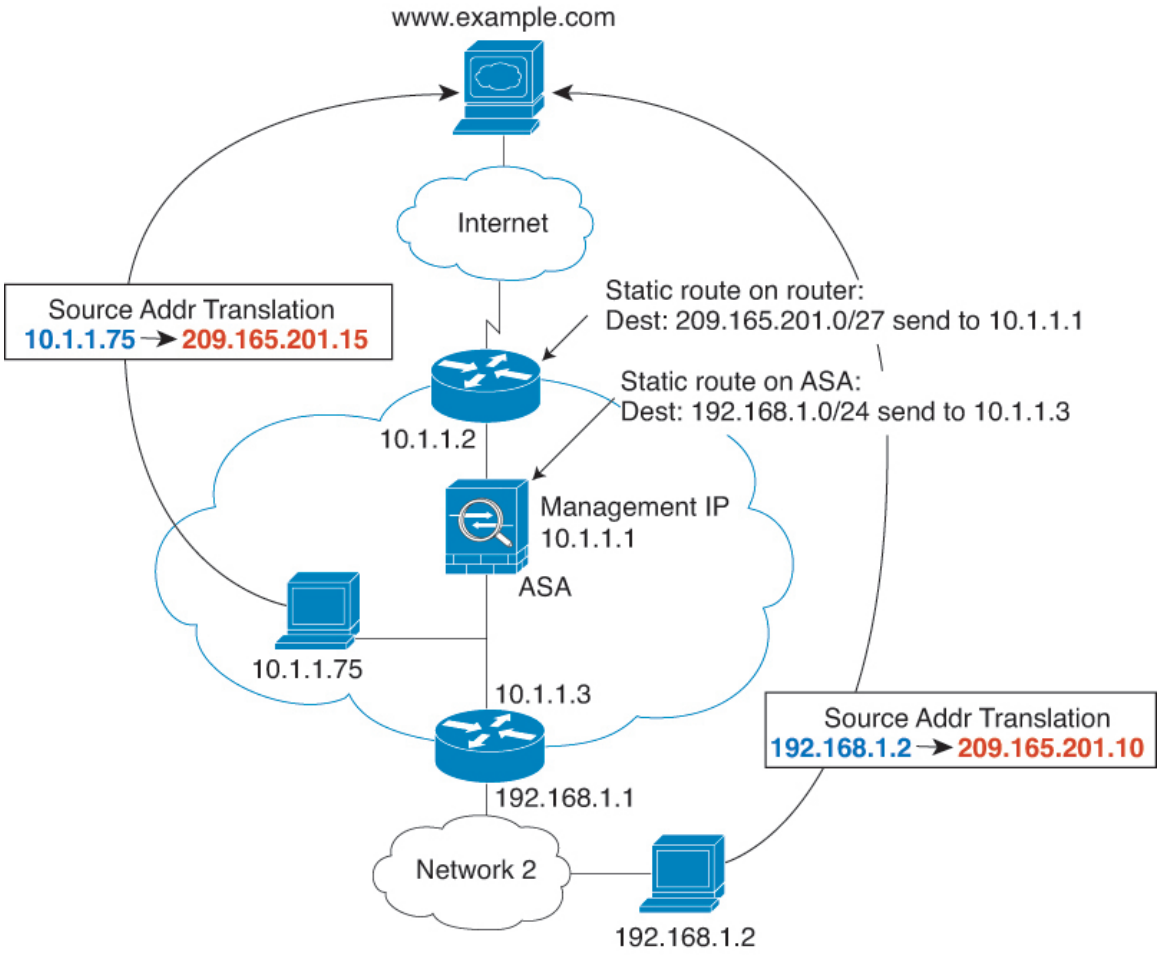
  - CTIQBE
  - GTP
  - H.323
  - MGCP
  - RTSP
  - SIP and Skinny (SCCP)
  - SQL*Net
  - SunRPC
  - TFTP

- Traffic at least one hop away for which the ASA performs NAT—Configure a static route on the ASA for traffic destined for the remote network. You also need a static route on the upstream router for traffic destined for the mapped addresses to be sent to the ASA.

  This routing requirement is also true for embedded IP addresses for VoIP and DNS with inspection and NAT enabled, and the embedded IP addresses are at least one hop away. The ASA needs to identify the correct egress interface so it can perform the translation.

NAT within a bridge group

Figure 22: NAT example: NAT within a bridge group



Unsupported features for bridge groups in transparent mode

This table lists the features that are not supported in bridge groups in transparent mode.

Table 4: Unsupported features in transparent mode

Feature	Description
Dynamic DNS	—
DHCPv6 stateless server	Only the DHCPv4 server is supported on bridge group member interfaces.

Feature	Description
DHCP relay	The transparent firewall can act as a DHCPv4 server, but it does not support DHCP relay. DHCP relay is not required because you can allow DHCP traffic to pass through using two access rules: one that allows DCHP requests from the inside interface to the outside, and one that allows the replies from the server in the other direction.
Dynamic routing protocols	You can, however, add static routes for traffic originating on the ASA for bridge group member interfaces. You can also allow dynamic routing protocols through the ASA using an access rule.
Multicast IP routing	You can allow multicast traffic through the ASA by allowing it in an access rule.
QoS	—
VPN termination for through traffic	The transparent firewall supports site-to-site VPN tunnels for management connections only on bridge group member interfaces. It does not terminate VPN connections for traffic through the ASA. You can pass VPN traffic through the ASA using an access rule, but it does not terminate non-management connections.
Unified Communications	—

Unsupported features for bridge groups in routed mode

This table lists the features are not supported in bridge groups in routed mode.

Table 5: Unsupported features in routed mode

Feature	Description
EtherChannel or VNI member interfaces	Only physical interfaces and subinterfaces are supported as bridge group member interfaces. Management interfaces are also not supported.
Clustering	Bridge groups are not supported in clustering.
Dynamic DNS	—
DHCPv6 stateless server	Only the DHCPv4 server is supported on BVIs.
DHCP relay	The routed firewall can act as a DHCPv4 server, but it does not support DHCP relay on BVIs or bridge group member interfaces.
Dynamic routing protocols	You can, however, add static routes for BVIs. You can also allow dynamic routing protocols through the ASA using an access rule. Non-bridge group interfaces support dynamic routing.
Multicast IP routing	You can allow multicast traffic through the ASA by allowing it in an access rule. Non-bridge group interfaces support multicast routing.

Feature	Description
Multiple Context Mode	Bridge groups are not supported in multiple context mode.
QoS	Non-bridge group interfaces support QoS.
VPN termination for through traffic	You cannot terminate a VPN connection on the BVI. Non-bridge group interfaces support VPN. Bridge group member interfaces support site-to-site VPN tunnels for management connections only. It does not terminate VPN connections for traffic through the ASA. You can pass VPN traffic through the bridge group using an access rule, but it does not terminate non-management connections. Clientless SSL VPN is also not supported.
Unified Communications	Non-bridge group interfaces support Unified Communications.

Default settings

Lists the default settings for firewall mode.

Default mode

The default mode is routed mode.

Bridge group defaults

By default, all ARP packets are passed within the bridge group.

Guidelines for firewall mode

Review these guidelines for firewall mode.

Context mode guidelines

Set the firewall mode per context.

Bridge group limits and network requirements

- Each directly-connected network must be on the same subnet.
- The Firewall Threat Defense device does not support traffic on secondary networks; only traffic on the same network as the BVI IP address is supported.

BVI IP address requirements

- An IP address for the BVI is required for each bridge group for to-the-device and from-the-device management traffic, as well as for data traffic to pass through the Firewall Threat Defense device. For IPv4 traffic, specify an IPv4 address. For IPv6 traffic, specify an IPv6 address.

- You can only configure IPv6 addresses manually.
- The BVI IP address must be on the same subnet as the connected network. You cannot set the subnet to a host subnet (255.255.255.255).

Unsupported bridge group members

- Management interfaces are not supported as bridge group members.
- For the ASAv50 on VMware with bridged ixgbevf interfaces, transparent mode is not supported, and bridge groups are not supported in routed mode.
- For the Firepower 2100 series, bridge groups are not supported in routed mode.
- For the 200/ 1010/ , you cannot mix logical VLAN interfaces and physical firewall interfaces in the same bridge group.

Transparent mode requirements and limitations

- In transparent mode, you must use at least 1 bridge group; data interfaces must belong to a bridge group.
- In transparent mode, do not specify the BVI IP address as the default gateway for connected devices; devices need to specify the router on the other side of the ASA as the default gateway.
- In transparent mode, the default route, which is required to provide a return path for management traffic, is only applied to management traffic from one bridge group network. This is because the default route specifies an interface in the bridge group as well as the router IP address on the bridge group network, and you can only define one default route. If you have management traffic from more than one bridge group network, you need to specify a regular static route that identifies the network from which you expect management traffic.
- In transparent mode, PPPoE is not supported for the Management interface.

Routed mode requirements and limitations

- In routed mode, to route between bridge groups and other routed interfaces, you must name the BVI.
- In routed mode, ASA-defined EtherChannel and VNI interfaces are not supported as bridge group members. EtherChannels on the Firepower 4100/9300 can be bridge group members.

Traffic handling limitation

- Bidirectional Forwarding Detection (BFD) echo packets are not allowed through the ASA when using bridge group members. If there are two neighbors on either side of the ASA running BFD, then the ASA will drop BFD echo packets because they have the same source and destination IP address and appear to be part of a LAND attack.

Set the Firewall Mode

This section describes how to change the firewall mode.

Note We recommend that you set the firewall mode before you perform any other configuration because changing the firewall mode clears the running configuration.

Before you begin

When you change modes, the ASA clears the running configuration (see Guidelines for firewall mode, on page 194 for more information).

- If you already have a populated configuration, be sure to back up your configuration before changing the mode; you can use this backup for reference when creating your new configuration. See Back Up and Restore Configurations or Other Files, on page 1233.
- Use the CLI at the console port to change the mode. If you use any other type of session, including the ASDM Command Line Interface tool or SSH, you will be disconnected when the configuration is cleared, and you will have to reconnect to the ASA using the console port in any case.
- Set the mode within the context.

Note To set the firewall mode to transparent and also configure ASDM management access after the configuration is cleared, see Configure ASDM Access, on page 21.

Procedure

Set the firewall mode to transparent:

firewall transparent

Example:

```
ciscoasa(config)# firewall transparent
```

To change the mode to routed, enter the **no firewall transparent** command.

Note
You are not prompted to confirm the firewall mode change; the change occurs immediately.

Examples for Firewall Mode

This section includes examples of how traffic moves through the ASA in the routed and transparent firewall mode.

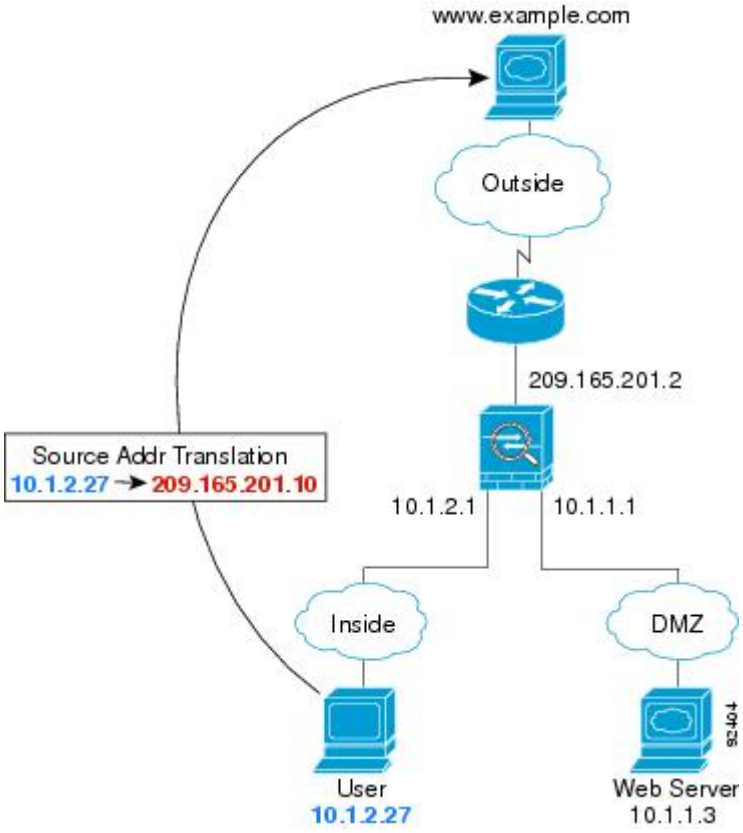
How Data Moves Through the ASA in Routed Firewall Mode

The following sections describe how data moves through the ASA in routed firewall mode in multiple scenarios.

An Inside User Visits a Web Server

The following figure shows an inside user accessing an outside web server.

Figure 23: Inside to Outside



The following steps describe how data moves through the ASA:

1. The user on the inside network requests a web page from www.example.com.
2. The ASA receives the packet and because it is a new session, it verifies that the packet is allowed according to the terms of the security policy.

   For multiple context mode, the ASA first classifies the packet to a context.
3. The ASA translates the real address (10.1.2.27) to the mapped address 209.165.201.10, which is on the outside interface subnet.

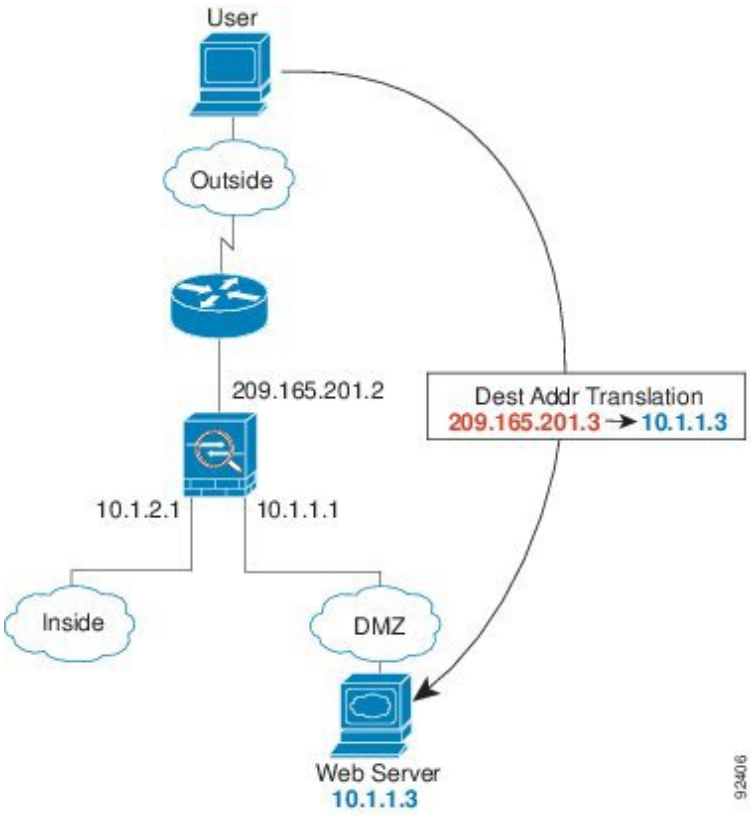
   The mapped address could be on any subnet, but routing is simplified when it is on the outside interface subnet.
4. The ASA then records that a session is established and forwards the packet from the outside interface.

5. When www.example.com responds to the request, the packet goes through the ASA, and because the session is already established, the packet bypasses the many lookups associated with a new connection. The ASA performs NAT by untranslating the global destination address to the local user address, 10.1.2.27.
6. The ASA forwards the packet to the inside user.

An Outside User Visits a Web Server on the DMZ

The following figure shows an outside user accessing the DMZ web server.

Figure 24: Outside to DMZ



The following steps describe how data moves through the ASA:

1. A user on the outside network requests a web page from the DMZ web server using the mapped address of 209.165.201.3, which is on the outside interface subnet.
2. The ASA receives the packet and untranslates the mapped address to the real address 10.1.1.3.
3. Because it is a new session, the ASA verifies that the packet is allowed according to the terms of the security policy.

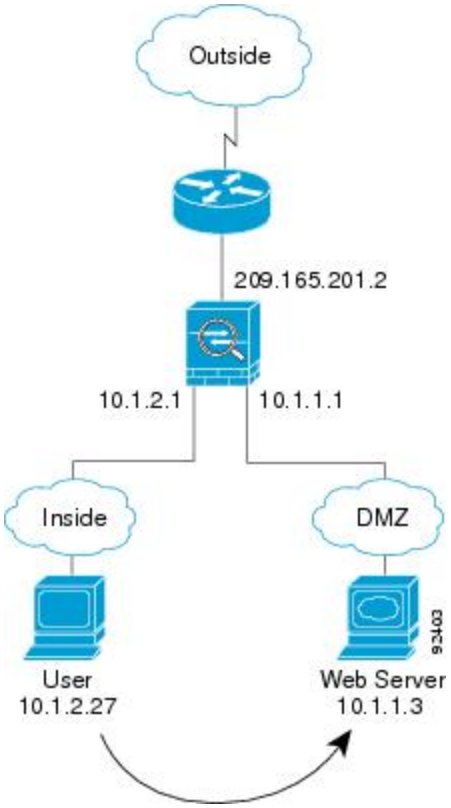
   For multiple context mode, the ASA first classifies the packet to a context.
4. The ASA then adds a session entry to the fast path and forwards the packet from the DMZ interface.

5. When the DMZ web server responds to the request, the packet goes through the ASA and because the session is already established, the packet bypasses the many lookups associated with a new connection. The ASA performs NAT by translating the real address to 209.165.201.3.
6. The ASA forwards the packet to the outside user.

An Inside User Visits a Web Server on the DMZ

The following figure shows an inside user accessing the DMZ web server.

Figure 25: Inside to DMZ



The following steps describe how data moves through the ASA:

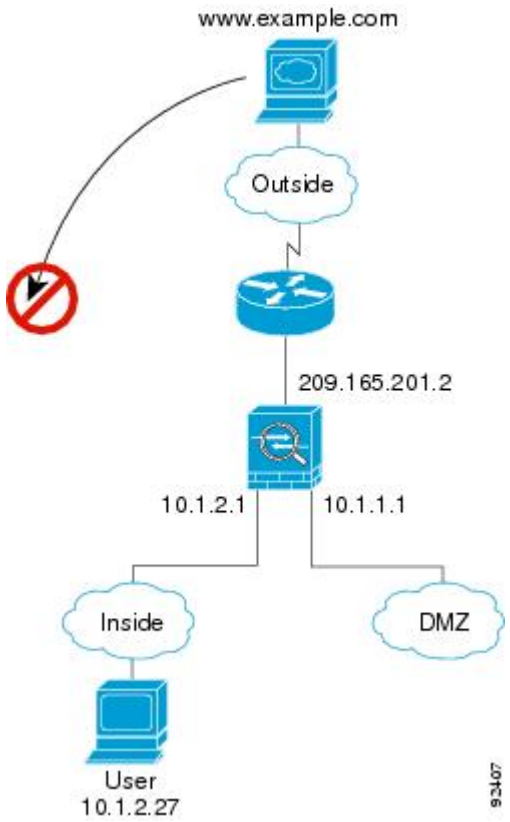
1. A user on the inside network requests a web page from the DMZ web server using the destination address of 10.1.1.3.
2. The ASA receives the packet and because it is a new session, the ASA verifies that the packet is allowed according to the terms of the security policy.

   For multiple context mode, the ASA first classifies the packet to a context.
3. The ASA then records that a session is established and forwards the packet out of the DMZ interface.
4. When the DMZ web server responds to the request, the packet goes through the fast path, which lets the packet bypass the many lookups associated with a new connection.
5. The ASA forwards the packet to the inside user.

An Outside User Attempts to Access an Inside Host

The following figure shows an outside user attempting to access the inside network.

Figure 26: Outside to Inside



The following steps describe how data moves through the ASA:

1. A user on the outside network attempts to reach an inside host (assuming the host has a routable IP address).

   If the inside network uses private addresses, no outside user can reach the inside network without NAT. The outside user might attempt to reach an inside user by using an existing NAT session.

2. The ASA receives the packet and because it is a new session, it verifies if the packet is allowed according to the security policy.

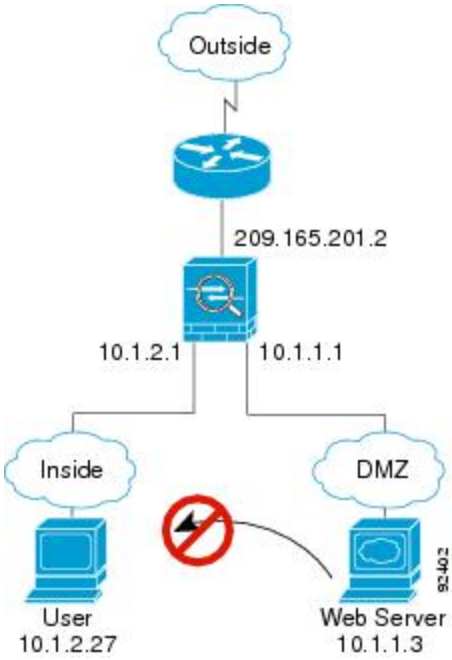
3. The packet is denied, and the ASA drops the packet and logs the connection attempt.

   If the outside user is attempting to attack the inside network, the ASA employs many technologies to determine if a packet is valid for an already established session.

A DMZ User Attempts to Access an Inside Host

The following figure shows a user in the DMZ attempting to access the inside network.

Figure 27: DMZ to Inside



The following steps describe how data moves through the ASA:

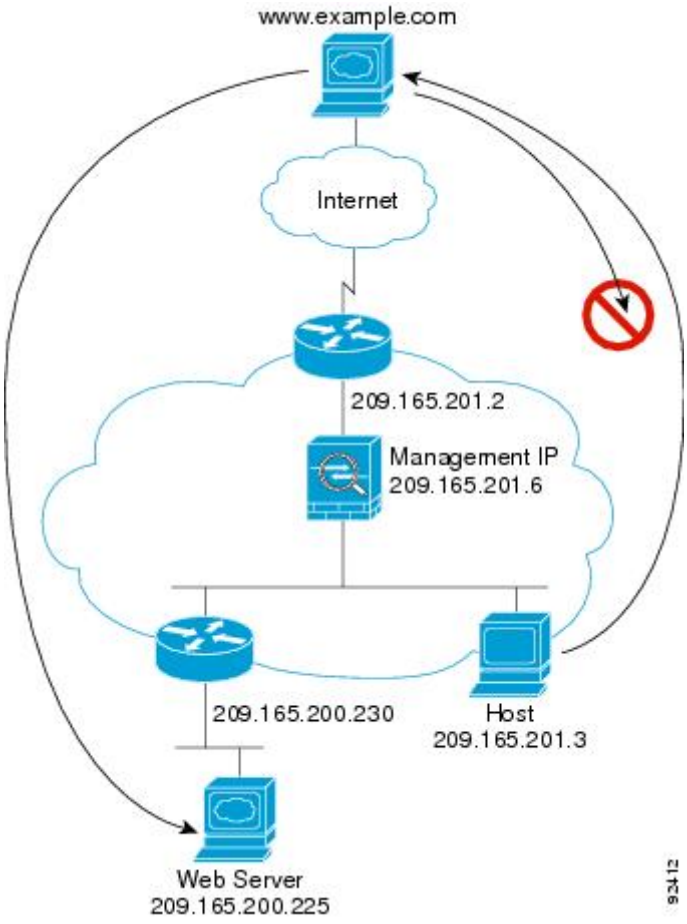
1. A user on the DMZ network attempts to reach an inside host. Because the DMZ does not have to route the traffic on the Internet, the private addressing scheme does not prevent routing.
2. The ASA receives the packet and because it is a new session, it verifies if the packet is allowed according to the security policy.

   The packet is denied, and the ASA drops the packet and logs the connection attempt.

How Data Moves Through the Transparent Firewall

The following figure shows a typical transparent firewall implementation with an inside network that contains a public web server. The ASA has an access rule so that the inside users can access Internet resources. Another access rule lets the outside users access only the web server on the inside network.

Figure 28: Typical Transparent Firewall Data Path

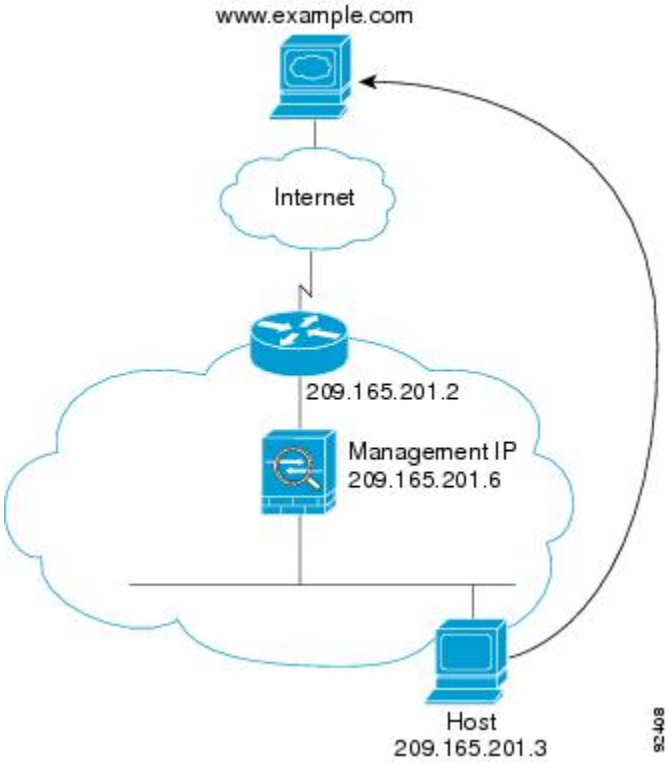


The following sections describe how data moves through the ASA.

An Inside User Visits a Web Server

The following figure shows an inside user accessing an outside web server.

Figure 29: Inside to Outside



The following steps describe how data moves through the ASA:

1. The user on the inside network requests a web page from www.example.com.
2. The ASA receives the packet and adds the source MAC address to the MAC address table, if required. Because it is a new session, it verifies that the packet is allowed according to the terms of the security policy.

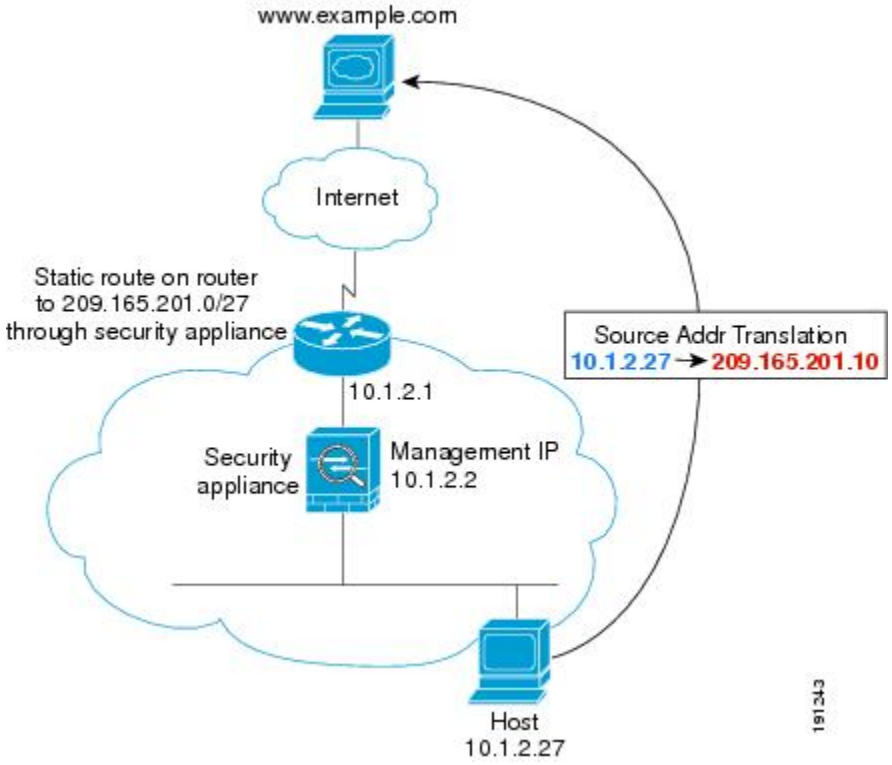
   For multiple context mode, the ASA first classifies the packet to a context.
3. The ASA records that a session is established.
4. If the destination MAC address is in its table, the ASA forwards the packet out of the outside interface. The destination MAC address is that of the upstream router, 209.165.201.2.

   If the destination MAC address is not in the ASA table, it attempts to discover the MAC address by sending an ARP request or a ping. The first packet is dropped.
5. The web server responds to the request; because the session is already established, the packet bypasses the many lookups associated with a new connection.
6. The ASA forwards the packet to the inside user.

An Inside User Visits a Web Server Using NAT

The following figure shows an inside user accessing an outside web server.

Figure 30: Inside to Outside with NAT



The following steps describe how data moves through the ASA:

1. The user on the inside network requests a web page from www.example.com.
2. The ASA receives the packet and adds the source MAC address to the MAC address table, if required. Because it is a new session, it verifies that the packet is allowed according to the terms of the security policy.

   For multiple context mode, the ASA first classifies the packet according to a unique interface.
3. The ASA translates the real address (10.1.2.27) to the mapped address 209.165.201.10.

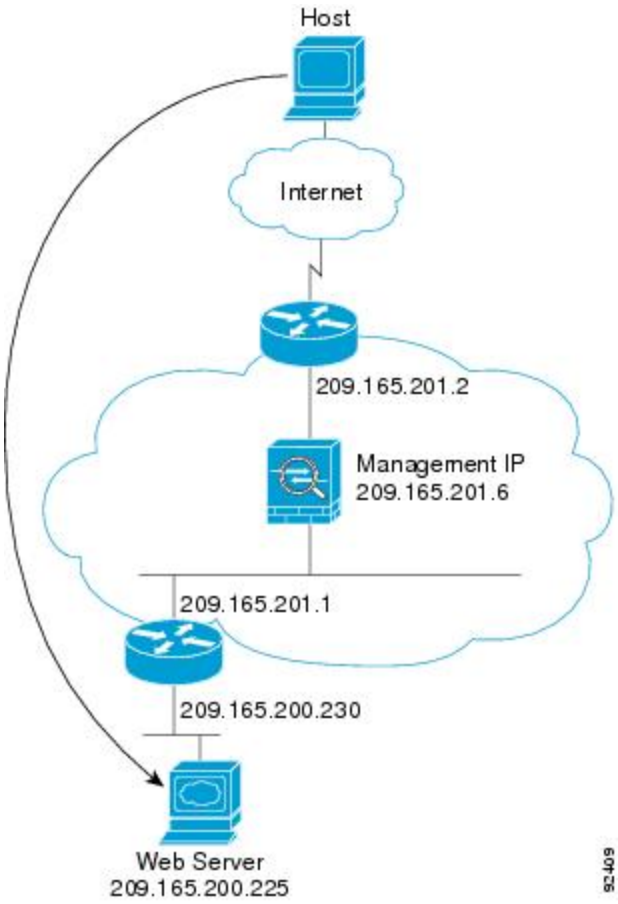
   Because the mapped address is not on the same network as the outside interface, then be sure the upstream router has a static route to the mapped network that points to the ASA.
4. The ASA then records that a session is established and forwards the packet from the outside interface.
5. If the destination MAC address is in its table, the ASA forwards the packet out of the outside interface. The destination MAC address is that of the upstream router, 10.1.2.1.

   If the destination MAC address is not in the ASA table, then it attempts to discover the MAC address by sending an ARP request and a ping. The first packet is dropped.
6. The web server responds to the request; because the session is already established, the packet bypasses the many lookups associated with a new connection.
7. The ASA performs NAT by untranslating the mapped address to the real address, 10.1.2.27.

An Outside User Visits a Web Server on the Inside Network

The following figure shows an outside user accessing the inside web server.

Figure 31: Outside to Inside



The following steps describe how data moves through the ASA:

1. A user on the outside network requests a web page from the inside web server.
2. The ASA receives the packet and adds the source MAC address to the MAC address table, if required. Because it is a new session, it verifies that the packet is allowed according to the terms of the security policy.

   For multiple context mode, the ASA first classifies the packet to a context.
3. The ASA records that a session is established.
4. If the destination MAC address is in its table, the ASA forwards the packet out of the inside interface. The destination MAC address is that of the downstream router, 209.165.201.1.

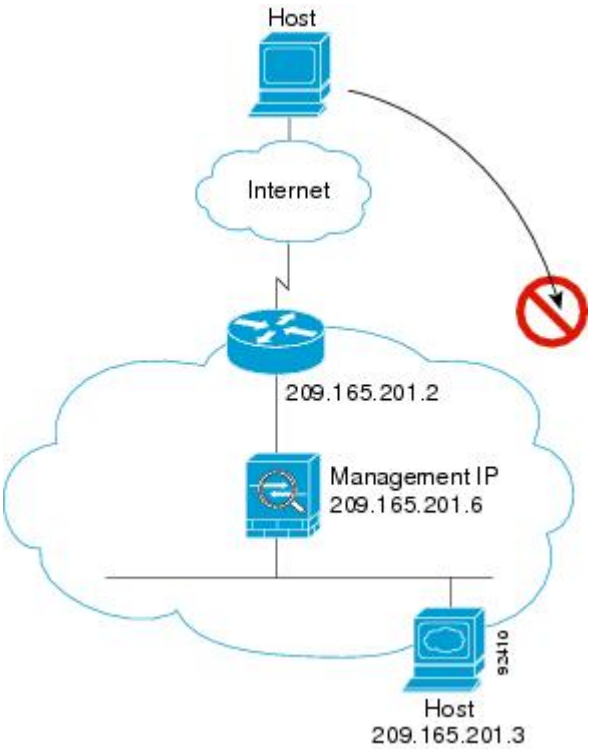
   If the destination MAC address is not in the ASA table, then it attempts to discover the MAC address by sending an ARP request and a ping. The first packet is dropped.
5. The web server responds to the request; because the session is already established, the packet bypasses the many lookups associated with a new connection.

6. The ASA forwards the packet to the outside user.

An Outside User Attempts to Access an Inside Host

The following figure shows an outside user attempting to access a host on the inside network.

Figure 32: Outside to Inside



The following steps describe how data moves through the ASA:

1. A user on the outside network attempts to reach an inside host.
2. The ASA receives the packet and adds the source MAC address to the MAC address table, if required. Because it is a new session, it verifies if the packet is allowed according to the terms of the security policy.

   For multiple context mode, the ASA first classifies the packet to a context.
3. The packet is denied because there is no access rule permitting the outside host, and the ASA drops the packet.
4. If the outside user is attempting to attack the inside network, the ASA employs many technologies to determine if a packet is valid for an already established session.

History for the Firewall Mode

Table 6: Feature History for Firewall Mode

Feature Name	Platform Releases	Feature Information
Transparent Firewall Mode	7.0(1)	A transparent firewall is a Layer 2 firewall that acts like a "bump in the wire," or a "stealth firewall," and is not seen as a router hop to connected devices. We introduced the following commands: **firewall transparent**, **show firewall**.
Transparent firewall bridge groups	8.4(1)	If you do not want the overhead of security contexts, or want to maximize your use of security contexts, you can group interfaces together in a bridge group, and then configure multiple bridge groups, one for each network. Bridge group traffic is isolated from other bridge groups. You can configure up to 8 bridge groups in single mode or per context in multiple mode, with 4 interfaces maximum per bridge group. **Note** Although you can configure multiple bridge groups on the ASA 5505, the restriction of 2 data interfaces in transparent mode on the ASA 5505 means you can only effectively use 1 bridge group. We introduced the following commands: **interface bvi**, **bridge-group**, **show bridge-group**.
Mixed firewall mode support in multiple context mode	8.5(1)/9.0(1)	You can set the firewall mode independently for each security context in multiple context mode, so some can run in transparent mode while others run in routed mode. We modified the following command: **firewall transparent**.
Transparent mode bridge group maximum increased to 250	9.3(1)	The bridge group maximum was increased from 8 to 250 bridge groups. You can configure up to 250 bridge groups in single mode or per context in multiple mode, with 4 interfaces maximum per bridge group. We modified the following commands: **interface bvi**, **bridge-group**.
Transparent mode maximum interfaces per bridge group increased to 64	9.6(2)	The maximum interfaces per bridge group was increased from 4 to 64. We did not modify any commands.

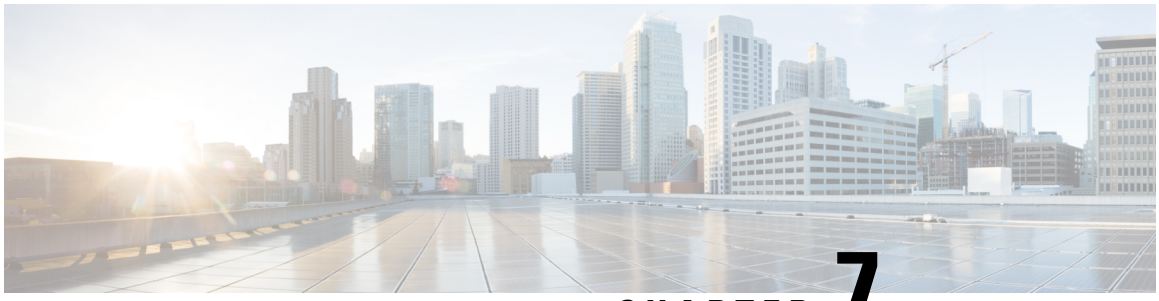
Feature Name	Platform Releases	Feature Information
Integrated Routing and Bridging	9.7(1)	Integrated Routing and Bridging provides the ability to route between a bridge group and a routed interface. A bridge group is a group of interfaces that the ASA bridges instead of routes. The ASA is not a true bridge in that the ASA continues to act as a firewall: access control between interfaces is controlled, and all of the usual firewall checks are in place. Previously, you could only configure bridge groups in transparent firewall mode, where you cannot route between bridge groups. This feature lets you configure bridge groups in routed firewall mode, and to route between bridge groups and between a bridge group and a routed interface. The bridge group participates in routing by using a Bridge Virtual Interface (BVI) to act as a gateway for the bridge group. Integrated Routing and Bridging provides an alternative to using an external Layer 2 switch if you have extra interfaces on the ASA to assign to the bridge group. In routed mode, the BVI can be a named interface and can participate separately from member interfaces in some features, such as access rules and DHCP server. The following features that are supported in transparent mode are not supported in routed mode: multiple context mode, ASA clustering. The following features are also not supported on BVIs: dynamic routing and multicast routing. We modified the following commands: **access-group, access-list ethertype, arp-inspection, dhcpd, mac-address-table static, mac-address-table aging-time, mac-learn, route, show arp-inspection, show bridge-group, show mac-address-table, show mac-learn**
Support for transparent mode deployment for a Firepower 4100/9300 ASA logical device	9.10(1)	You can now specify transparent or routed mode when you deploy the ASA on a Firepower 4100/9300. New/Modified FXOS commands: **enter bootstrap-key FIREWALL_MODE**, **set value routed**, **set value transparent**



PART II

High Availability and Scalability

- Multiple Context Mode, on page 211
- Failover for High Availability, on page 257
- Failover for High Availability in the Public Cloud, on page 313
- ASA Cluster for the Secure Firewall 3100, on page 333
- ASA Cluster for the Firepower 4100/9300, on page 423
- ASA Cluster for the ASAv for the Private Cloud, on page 523



CHAPTER 7

Multiple Context Mode

This chapter describes how to configure multiple security contexts on the ASA.

- About Security Contexts, on page 211
- Licensing for Multiple Context Mode, on page 221
- Prerequisites for Multiple Context Mode, on page 222
- Guidelines for Multiple Context Mode, on page 222
- Defaults for Multiple Context Mode, on page 224
- Configure Multiple Contexts, on page 224
- Change Between Contexts and the System Execution Space, on page 235
- Manage Security Contexts, on page 235
- Monitoring Security Contexts, on page 239
- Examples for Multiple Context Mode, on page 251
- History for Multiple Context Mode, on page 252

About Security Contexts

You can partition a single ASA into multiple virtual devices, known as security contexts. Each context acts as an independent device, with its own security policy, interfaces, and administrators. Multiple contexts are similar to having multiple standalone devices. For unsupported features in multiple context mode, see Guidelines for Multiple Context Mode, on page 222.

This section provides an overview of security contexts.

Common Uses for Security Contexts

You might want to use multiple security contexts in the following situations:

- You are a service provider and want to sell security services to many customers. By enabling multiple security contexts on the ASA, you can implement a cost-effective, space-saving solution that keeps all customer traffic separate and secure, and also eases configuration.
- You are a large enterprise or a college campus and want to keep departments completely separate.
- You are an enterprise that wants to provide distinct security policies to different departments.
- You have any network that requires more than one ASA.

Context Configuration Files

This section describes how the ASA implements multiple context mode configurations.

Context Configurations

For each context, the ASA includes a configuration that identifies the security policy, interfaces, and all the options you can configure on a standalone device. You can store context configurations in flash memory, or you can download them from a TFTP, FTP, or HTTP(S) server.

System Configuration

The system administrator adds and manages contexts by configuring each context configuration location, allocated interfaces, and other context operating parameters in the system configuration, which, like a single mode configuration, is the startup configuration. The system configuration identifies basic settings for the ASA. The system configuration does not include any network interfaces or network settings for itself; rather, when the system needs to access network resources (such as downloading the contexts from the server), it uses one of the contexts that is designated as the *admin context*. The system configuration does include a specialized failover interface for failover traffic only.

Admin Context Configuration

The admin context is just like any other context, except that when a user logs in to the admin context, then that user has system administrator rights and can access the system and all other contexts. The admin context is not restricted in any way, and can be used as a regular context. However, because logging into the admin context grants you administrator privileges over all contexts, you might need to restrict access to the admin context to appropriate users. The admin context must reside on flash memory, and not remotely.

If your system is already in multiple context mode, or if you convert from single mode, the admin context is created automatically as a file on the internal flash memory called admin.cfg. This context is named “admin.” If you do not want to use admin.cfg as the admin context, you can change the admin context.

How the ASA Classifies Packets

Each packet that enters the ASA must be classified, so that the ASA can determine to which context to send a packet.



Note If the destination MAC address is a multicast or broadcast MAC address, the packet is duplicated and delivered to each context.

Valid Classifier Criteria

This section describes the criteria used by the classifier.



Note For management traffic destined for an interface, the interface IP address is used for classification.

The routing table is not used for packet classification.

Unique Interfaces

If only one context is associated with the ingress interface, the ASA classifies the packet into that context. In transparent firewall mode, unique interfaces for contexts are required, so this method is used to classify packets at all times.

Unique MAC Addresses

If multiple contexts share an interface, then the classifier uses unique MAC addresses assigned to the interface in each context. An upstream router cannot route directly to a context without unique MAC addresses. You can enable auto-generation of MAC addresses. You can also set the MAC addresses manually when you configure each interface.

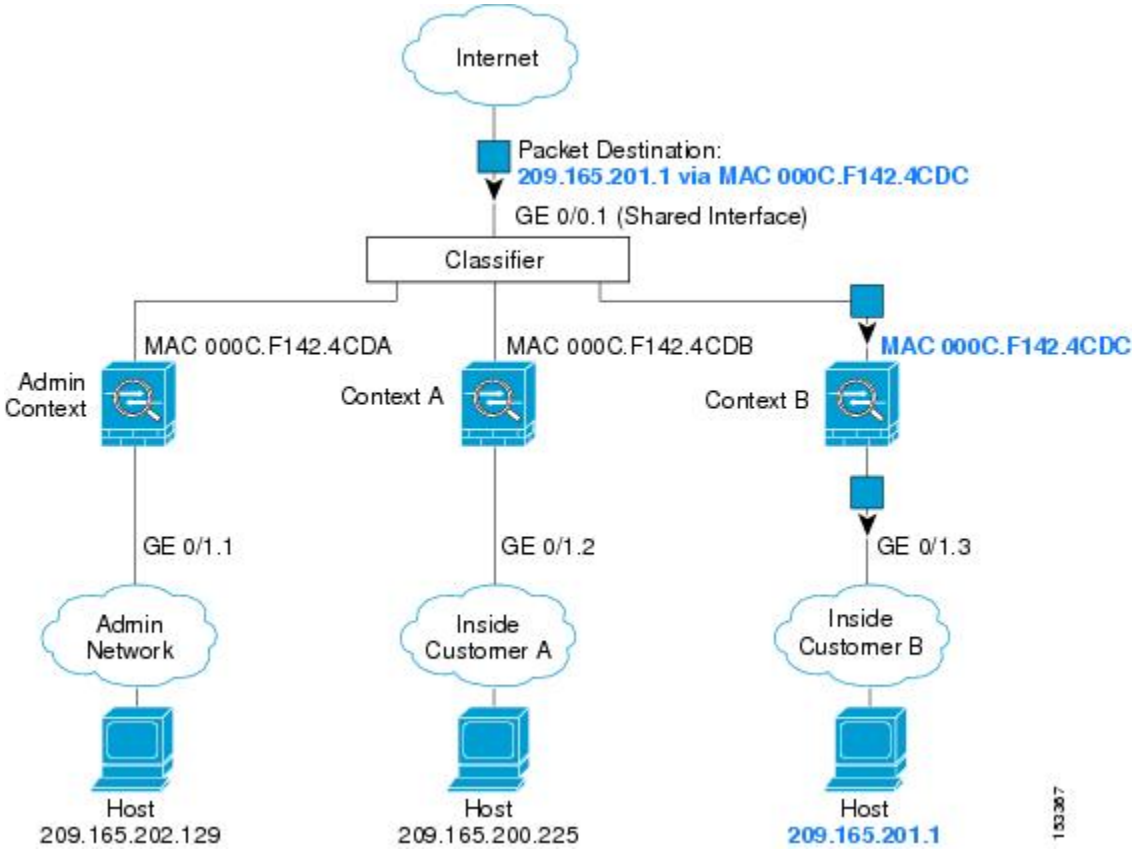
NAT Configuration

If you do not enable use of unique MAC addresses, then the ASA uses the mapped addresses in your NAT configuration to classify packets. We recommend using MAC addresses instead of NAT, so that traffic classification can occur regardless of the completeness of the NAT configuration.

Classification Examples

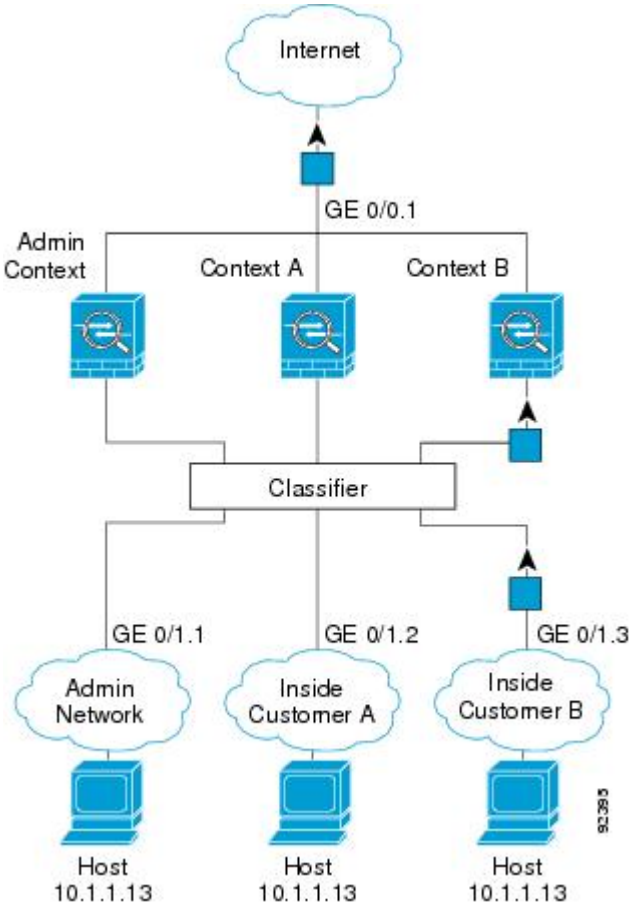
The following figure shows multiple contexts sharing an outside interface. The classifier assigns the packet to Context B because Context B includes the MAC address to which the router sends the packet.

Figure 33: Packet Classification with a Shared Interface Using MAC Addresses



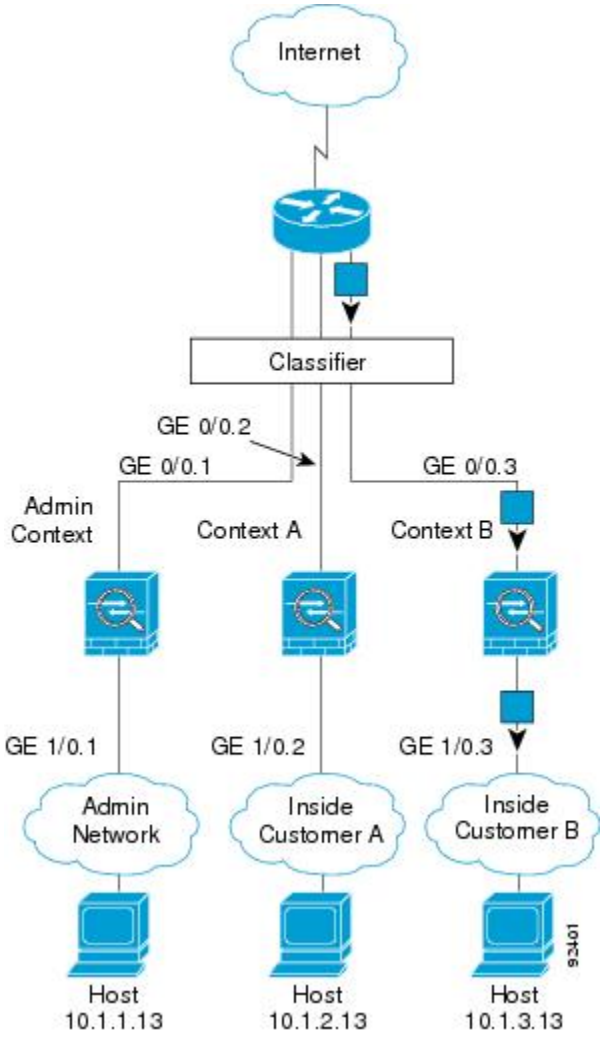
Note that all new incoming traffic must be classified, even from inside networks. The following figure shows a host on the Context B inside network accessing the Internet. The classifier assigns the packet to Context B because the ingress interface is Gigabit Ethernet 0/1.3, which is assigned to Context B.

Figure 34: Incoming Traffic from Inside Networks



For transparent firewalls, you must use unique interfaces. The following figure shows a packet destined to a host on the Context B inside network from the Internet. The classifier assigns the packet to Context B because the ingress interface is Gigabit Ethernet 1/0.3, which is assigned to Context B.

Figure 35: Transparent Firewall Contexts



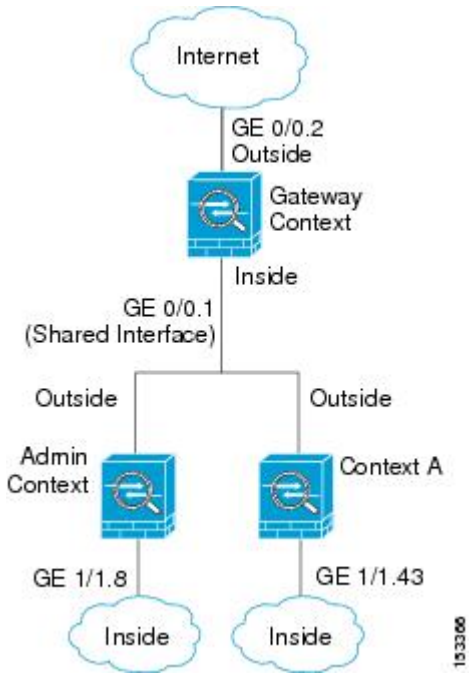
Cascading Security Contexts

Placing a context directly in front of another context is called *cascading contexts*; the outside interface of one context is the same interface as the inside interface of another context. You might want to cascade contexts if you want to simplify the configuration of some contexts by configuring shared parameters in the top context.

Note Cascading contexts requires unique MAC addresses for each context interface. Because of the limitations of classifying packets on shared interfaces without MAC addresses, we do not recommend using cascading contexts without unique MAC addresses.

The following figure shows a gateway context with two contexts behind the gateway.

Figure 36: Cascading Contexts



Management Access to Security Contexts

The ASA provides system administrator access in multiple context mode as well as access for individual context administrators.

System Administrator Access

You can access the ASA as a system administrator in two ways:

- Access the ASA console.

  From the console, you access the *system execution space*, which means that any commands you enter affect only the system configuration or the running of the system (for run-time commands).

- Access the admin context using Telnet, SSH, or ASDM.

As the system administrator, you can access all contexts.

The system execution space does not support any AAA commands, but you can configure its own enable password, as well as usernames in the local database to provide individual logins.

Context Administrator Access

You can access a context using Telnet, SSH, or ASDM. If you log in to a non-admin context, you can only access the configuration for that context. You can provide individual logins to the context.

Management Interface Usage

The Management interface is a separate interface just for management traffic.

In routed firewall mode, you can share the Management interface across all contexts.

In transparent firewall mode, the Management interface is special. In addition to the maximum allowed through-traffic interfaces, you can also use the Management interface as a separate management-only interface. However, in multiple context mode, you cannot share any interfaces across transparent contexts. You can instead use subinterfaces of the Management interface, and assign one to each context. However, only Firepower device models allow subinterfaces on the Management interface. For ASA models, you must use a data interface or a subinterface of a data interface, and add it to a bridge group within the context.

For the Firepower 4100/9300 chassis transparent context, neither the Management interface nor subinterface retains its special status. In this case, you must treat it as a data interface, and add it to a bridge group. (Note that in single context mode, the Management interface does retain its special status.)

Another consideration about transparent mode: when you enable multiple context mode, all configured interfaces are automatically assigned to the Admin context. For example, if your default configuration includes the Management interface, then that interface will be assigned to the Admin context. One option is to leave the main interface allocated to the Admin context and manage it using the native VLAN, and then use subinterfaces to manage each context. Keep in mind that if you make the Admin context transparent, its IP address will be removed; you have to assign it to a bridge group and assign the IP address to the BVI.

About Resource Management

By default, all security contexts have unlimited access to the resources of the ASA, except where maximum limits per context are enforced; the only exception is VPN resources, which are disabled by default. If you find that one or more contexts use too many resources, and they cause other contexts to be denied connections, for example, then you can configure resource management to limit the use of resources per context. For VPN resources, you must configure resource management to allow any VPN tunnels.

Resource Classes

The ASA manages resources by assigning contexts to resource classes. Each context uses the resource limits set by the class. To use the settings of a class, assign the context to the class when you define the context. All contexts belong to the default class if they are not assigned to another class; you do not have to actively assign a context to default. You can only assign a context to one resource class. The exception to this rule is that limits that are undefined in the member class are inherited from the default class; so in effect, a context could be a member of default plus another class.

Resource Limits

You can set the limit for individual resources as a percentage (if there is a hard system limit) or as an absolute value.

For most resources, the ASA does not set aside a portion of the resources for each context assigned to the class; rather, the ASA sets the maximum limit for a context. If you oversubscribe resources, or allow some resources to be unlimited, a few contexts can “use up” those resources, potentially affecting service to other contexts. The exception is VPN resource types, which you cannot oversubscribe, so the resources assigned to each context are guaranteed. To accommodate temporary bursts of VPN sessions beyond the amount assigned, the ASA supports a “burst” VPN resource type, which is equal to the remaining unassigned VPN sessions. The burst sessions *can* be oversubscribed, and are available to contexts on a first-come, first-served basis.

Default Class

All contexts belong to the default class if they are not assigned to another class; you do not have to actively assign a context to the default class.

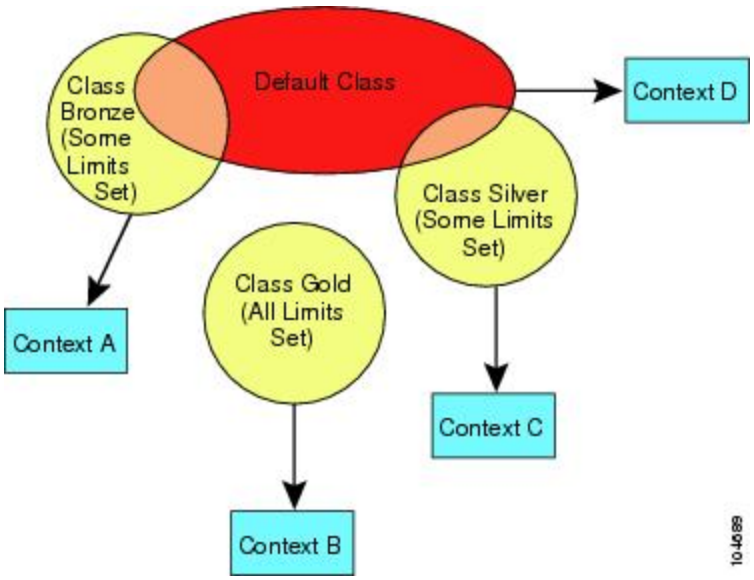
If a context belongs to a class other than the default class, those class settings always override the default class settings. However, if the other class has any settings that are not defined, then the member context uses the default class for those limits. For example, if you create a class with a 2 percent limit for all concurrent connections, but no other limits, then all other limits are inherited from the default class. Conversely, if you create a class with a limit for all resources, the class uses no settings from the default class.

For most resources, the default class provides unlimited access to resources for all contexts, except for the following limits:

- Telnet sessions—5 sessions. (The maximum per context.)
- SSH sessions—5 sessions. (The maximum per context.)
- ASDM sessions—5 sessions. (The maximum per context.)
- MAC addresses—(varies per model). (The maximum for the system.)
- AnyConnect Client peers—0 sessions. (You must manually configure the class to allow any AnyConnect Client peers.)
- VPN site-to-site tunnels—0 sessions. (You must manually configure the class to allow any VPN sessions.)

The following figure shows the relationship between the default class and other classes. Contexts A and C belong to classes with some limits set; other limits are inherited from the default class. Context B inherits no limits from default because all limits are set in its class, the Gold class. Context D was not assigned to a class, and is by default a member of the default class.

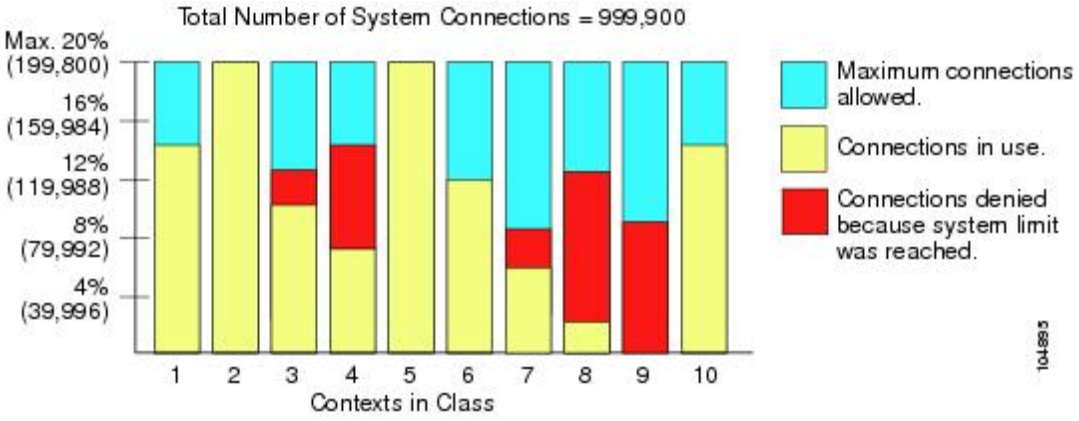
Figure 37: Resource Classes



Use Oversubscribed Resources

You can oversubscribe the ASA by assigning more than 100 percent of a resource across all contexts (with the exception of non-burst VPN resources). For example, you can set the Bronze class to limit connections to 20 percent per context, and then assign 10 contexts to the class for a total of 200 percent. If contexts concurrently use more than the system limit, then each context gets less than the 20 percent you intended.

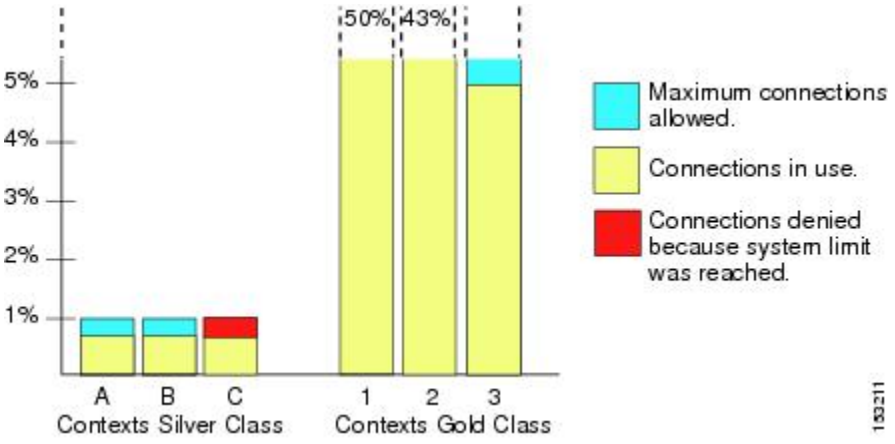
Figure 38: Resource Oversubscription



Use Unlimited Resources

The ASA lets you assign unlimited access to one or more resources in a class, instead of a percentage or absolute number. When a resource is unlimited, contexts can use as much of the resource as the system has available. For example, Context A, B, and C are in the Silver Class, which limits each class member to 1 percent of the connections, for a total of 3 percent; but the three contexts are currently only using 2 percent combined. Gold Class has unlimited access to connections. The contexts in the Gold Class can use more than the 97 percent of "unassigned" connections; they can also use the 1 percent of connections not currently in use by Context A, B, and C, even if that means that Context A, B, and C are unable to reach their 3 percent combined limit. Setting unlimited access is similar to oversubscribing the ASA, except that you have less control over how much you oversubscribe the system.

Figure 39: Unlimited Resources



About MAC Addresses

You can manually assign MAC addresses to override the default. For multiple context mode, you can automatically generate unique MAC addresses (for all interfaces assigned to a context) and single context mode (for subinterfaces)..

Note You might want to assign unique MAC addresses to subinterfaces defined on the ASA, because they use the same burned-in MAC address of the parent interface. For example, your service provider might perform access control based on the MAC address. Also, because IPv6 link-local addresses are generated based on the MAC address, assigning unique MAC addresses to subinterfaces allows for unique IPv6 link-local addresses, which can avoid traffic disruption in certain instances on the ASA device.

MAC Addresses in Multiple Context Mode

The MAC address is used to classify packets within a context. If you share an interface, but do not have unique MAC addresses for the interface in each context, then other classification methods are attempted that might not provide full coverage.

To allow contexts to share interfaces, you should enable auto-generation of virtual MAC addresses to each shared context interface.

Automatic MAC Addresses

In multiple context mode, auto-generation assigns unique MAC addresses to all interfaces assigned to a context.

If you manually assign a MAC address and also enable auto-generation, then the manually assigned MAC address is used. If you later remove the manual MAC address, the auto-generated address is used, if enabled.

In the rare circumstance that the generated MAC address conflicts with another private MAC address in your network, you can manually set the MAC address for the interface.

Because auto-generated addresses (when using a prefix) start with A2, you cannot start manual MAC addresses with A2 if you also want to use auto-generation.

The ASA generates the MAC address using the following format:

A2*xx.yyzz.zzzz*

Where *xx.yy* is a user-defined prefix or an autogenerated prefix based on the last two bytes of the interface MAC address, and *zz.zzzz* is an internal counter generated by the ASA. For the standby MAC address, the address is identical except that the internal counter is increased by 1.

For an example of how the prefix is used, if you set a prefix of 77, then the ASA converts 77 into the hexadecimal value 004D (*yyxx*). When used in the MAC address, the prefix is reversed (*xxyy*) to match the ASA native form:

A2**4D.00***zz.zzzz*

For a prefix of 1009 (03F1), the MAC address is:

A2**F1.03***zz.zzzz*

Note The MAC address format without a prefix is a legacy version. See the **mac-address auto** command in the command reference for more information about the legacy format.

VPN Support

For VPN resources, you must configure resource management to allow any VPN tunnels.

You can use site-to-site VPN in multiple context mode.

For remote access VPN, you must use AnyConnect 3.x and later for SSL VPN and IKEv2 protocol. You can customize flash storage per context for AnyConnect Client images and customizations, as well as using shared flash memory across all contexts. For unsupported features, see Guidelines for Multiple Context Mode, on page 222. For a detailed list of supported VPN features per ASA release, see History for Multiple Context Mode, on page 252.

Note The AnyConnect Apex license is required for multiple context mode; you cannot use the default or legacy license.

Licensing for Multiple Context Mode

Model	License Requirement
Firepower 1010	No support.
Firepower 1100	Standard License: 2 contexts. *Optional License, Maximum:* *Firepower 1120: 5* *Firepower 1140: 10* *Firepower 1150: 25*
Firepower 2100	Standard License: 2 contexts. *Optional License, Maximum:* *Firepower 2110: 25* *Firepower 2120: 25* *Firepower 2130: 30* *Firepower 2140: 40*

Model	License Requirement
Secure Firewall 3100	Standard License: 2 contexts. *Optional License, Maximum:* *Secure Firewall 3110: 100* *Secure Firewall 3120: 100* *Secure Firewall 3130: 100* *Secure Firewall 3140: 100*
Firepower 4100	Standard License: 10 contexts. *Optional License: up to 250 contexts.*
Firepower 9300	Standard License: 10 contexts. *Optional License: up to 250 contexts.*
ISA 3000	No support.
ASAv	No support.

Note If the Admin context only contains management-only interfaces, and does not include any data interfaces for through traffic, then it does not count against the limit.

Note The AnyConnect Apex license is required for multiple context mode; you cannot use the default or legacy license.

Prerequisites for Multiple Context Mode

After you are in multiple context mode, connect to the system or the admin context to access the system configuration. You cannot configure the system from a non-admin context. By default, after you enable multiple context mode, you can connect to the admin context by using the default management IP address.

Guidelines for Multiple Context Mode

Failover

Active/Active mode failover is only supported in multiple context mode.

IPv6

Cross-context IPv6 routing is not supported.

Unsupported Features

Multiple context mode does not support the following features:

- RIP
- OSPFv3. (OSPFv2 is supported.)
- Multicast routing
- Threat Detection
- Unified Communications
- QoS
- Virtual Tunnel Interfaces (VTIs)
- Static route tracking

Multiple context mode does not currently support the following features for remote access VPN:

- AnyConnect 2.x and earlier
- IKEv1
- SAML
- WebLaunch
- VLAN Mapping
- HostScan
- VPN load balancing
- Customization
- L2TP

Additional Guidelines

- The context mode (single or multiple) is not stored in the configuration file, even though it does endure reboots. If you need to copy your configuration to another device, set the mode on the new device to match.
- If you store context configurations in the root directory of flash memory, on some models you might run out of room in that directory, even though there is available memory. In this case, create a subdirectory for your configuration files. Background: some models use the FAT 16 file system for internal flash memory, and if you do not use 8.3-compliant short names, or use uppercase characters, then fewer than 512 files and folders can be stored because the file system uses up slots to store long file names (see https://support.microsoft.com).
- In ACI, Policy Based Redirect (PBR) health check is executed (L2 pings) using the same MAC address on all the leaves. This results in a MAC flap. To resolve the MAC flap, you can configure the tap-mode

option on the inline set. However, if the Firewall Threat Defense high availability is configured, you must enable MAC learning for connection handling during a failover. Thus, in an ACI environment with Firewall Threat Defense in HA pair using inline-set interfaces, to avoid packet losses, deploy the Firewall Threat Defense in standalone or in a cluster.

Defaults for Multiple Context Mode

- By default, the ASA is in single context mode.
- See Default Class, on page 218.

Configure Multiple Contexts

Procedure

Step 1 Enable or Disable Multiple Context Mode, on page 224.

Step 2 (Optional) Configure a Class for Resource Management, on page 226.

Note
For VPN support, you must configure VPN resources in a resource class; the default class does not allow VPN.

Step 3 Configure interfaces in the system execution space.

- Firepower 1100, Firepower 2100 in Appliance mode, Secure Firewall 3100—Basic Interface Configuration, on page 593.
- Firepower 2100 in Platform mode—See the getting started guide.
- Firepower 4100/9300—Logical Devices for the Firepower 4100/9300, on page 163

Step 4 Configure a Security Context, on page 230.

Step 5 (Optional) Assign MAC Addresses to Context Interfaces Automatically, on page 234.

Step 6 Complete interface configuration in the context. See Routed and Transparent Mode Interfaces, on page 669.

Enable or Disable Multiple Context Mode

Your ASA might already be configured for multiple security contexts depending on how you ordered it from Cisco. If you need to convert from single mode to multiple mode, follow the procedures in this section.

Enable Multiple Context Mode

When you convert from single mode to multiple mode, the ASA converts the running configuration into two files: a new startup configuration that comprises the system configuration, and admin.cfg that comprises the

admin context (in the root directory of the internal flash memory). The original running configuration is saved as old_running.cfg (in the root directory of the internal flash memory). The original startup configuration is not saved. The ASA automatically adds an entry for the admin context to the system configuration with the name “admin.”

Before you begin

Back up your startup configuration if it differs from the running configuration. When you convert from single mode to multiple mode, the ASA converts the running configuration into two files. The original startup configuration is not saved. See Back Up and Restore Configurations or Other Files, on page 1233.

Procedure

Change to multiple context mode.

mode multiple

Example:

You are prompted to change the mode and convert the configuration, and then the system reloads.

Note
You will have to regenerate the RSA key pair in the Admin context before you can reestablish an SSH connection. From the console, enter the **crypto key generate rsa modulus** command. See Configure SSH Access, on page 1161 for more information.

Example:

```
ciscoasa(config)# mode multiple
WARNING: This command will change the behavior of the device
WARNING: This command will initiate a Reboot
Proceed with change mode? [confirm]
Convert the system configuration? [confirm]
!
The old running configuration file will be written to flash

Converting the configuration - this may take several minutes for a large configuration

The admin context configuration will be written to flash

The new running configuration file was written to flash
Security context mode: multiple
ciscoasa(config)#


***
*** --- START GRACEFUL SHUTDOWN ---
***
*** Message to all terminals:
***
***   change mode
Shutting down isakmp
Shutting down webvpn
Shutting down License Controller
Shutting down File system
```

```
***
*** --- SHUTDOWN NOW ---
***
*** Message to all terminals:
***
***   change mode
```

Restore Single Context Mode

To copy the old running configuration to the startup configuration and to change the mode to single mode, perform the following steps.

Before you begin

Perform this procedure in the system execution space.

Procedure

Step 1 Copy the backup version of your original running configuration to the current startup configuration:

copy disk0:old_running.cfg startup-config

Example:

```
ciscoasa(config)# copy disk0:old_running.cfg startup-config
```

Step 2 Set the mode to single mode:

mode single

Example:

```
ciscoasa(config)# mode single
```

You are prompted to reboot the ASA.

Configure a Class for Resource Management

To configure a class in the system configuration, perform the following steps. You can change the value of a particular resource limit by reentering the command with a new value.

Before you begin

- Perform this procedure in the system execution space.
- The following table lists the resource types and the limits. See also the **show resource types** command.

Note If the System Limit is N/A, then you cannot set a percentage of the resource because there is no hard system limit for the resource.

Table 7: Resource Names and Limits

Resource Name	Rate or Concurrent	Minimum and Maximum Number per Context	System Limit	Description
asdm	Concurrent	1 minimum 5 maximum	200	ASDM management sessions. ASDM sessions use two HTTPS connections: one for monitoring that is always present, and one for making configuration changes that is present only when you make changes. For example, the system limit of 200 ASDM sessions represents a limit of 400 HTTPS sessions.
conns	Concurrent or Rate	N/A	Concurrent connections: See Licenses Per Model, on page 142 for the connection limit available for your model. Rate: N/A	TCP or UDP connections between any two hosts, including connections between one host and multiple other hosts. **Note** Syslog messages are generated for whichever limit is lower, xlates or conns. For example, if you set the xlates limit to 7 and the conns to 9, then the ASA only generates syslog message 321001 (“Resource 'xlates' limit of 7 reached for context 'ctx1'”) and not 321002 (“Resource 'conn rate' limit of 5 reached for context 'ctx1'”).
hosts	Concurrent	N/A	N/A	Hosts that can connect through the ASA.
inspects	Rate	N/A	N/A	Application inspections per second.
mac-addresses	Concurrent	N/A	(varies per model)	For transparent firewall mode, the number of MAC addresses allowed in the MAC address table.
routes	Concurrent	N/A	N/A	Dynamic routes.

Resource Name	Rate or Concurrent	Minimum and Maximum Number per Context	System Limit	Description
vpn burst anyconnect	Concurrent	N/A	The AnyConnect Client Premium Peers for your model minus the sum of the sessions assigned to all contexts for **vpn anyconnect**.	The number of AnyConnect Client sessions allowed beyond the amount assigned to a context with **vpn anyconnect**. For example, if your model supports 5000 peers, and you assign 4000 peers across all contexts with **vpn anyconnect**, then the remaining 1000 sessions are available for **vpn burst anyconnect**. Unlike **vpn anyconnect**, which guarantees the sessions to the context, **vpn burst anyconnect** can be oversubscribed; the burst pool is available to all contexts on a first-come, first-served basis.
vpn anyconnect	Concurrent	N/A	See Licenses Per Model, on page 142 for the AnyConnect Client Premium Peers available for your model.	AnyConnect Client peers. You cannot oversubscribe this resource; all context assignments combined cannot exceed the model limit. The peers you assign for this resource are guaranteed to the context.
vpn burst other	Concurrent	N/A	The Other VPN session amount for your model minus the sum of the sessions assigned to all contexts for **vpn other**.	The number of site-to-site VPN sessions allowed beyond the amount assigned to a context with **vpn other**. For example, if your model supports 5000 sessions, and you assign 4000 sessions across all contexts with **vpn other**, then the remaining 1000 sessions are available for **vpn burst other**. Unlike **vpn other**, which guarantees the sessions to the context, **vpn burst other** can be oversubscribed; the burst pool is available to all contexts on a first-come, first-served basis.
vpn other	Concurrent	N/A	See Supported Feature Licenses Per Model, on page 70 for the Other VPN sessions available for your model.	Site-to-site VPN sessions. You cannot oversubscribe this resource; all context assignments combined cannot exceed the model limit. The sessions you assign for this resource are guaranteed to the context.
ikev1 in-negotiation	Concurrent (percentage only)	N/A	A percentage of the Other VPN sessions assigned to this context. See the **vpn other** resources to assign sessions to the context.	Incoming IKEv1 SA negotiations, as a percentage of the context Other VPN limit.

Resource Name	Rate or Concurrent	Minimum and Maximum Number per Context	System Limit	Description
ssh	Concurrent	1 minimum 5 maximum	100	SSH sessions.
storage	MB	The maximum depends on your specified flash memory drive	The maximum depends on your specified flash memory drive	Storage limit of context directory in MB. Specify the drive using the **storage-url** command.
syslogs	Rate	N/A	N/A	Syslog messages per second.
telnet	Concurrent	1 minimum 5 maximum	100	Telnet sessions.
xlates	Concurrent	N/A	N/A	Network address translations.

Procedure

Step 1 Specify the class name and enter the class configuration mode:

class *name*

Example:

```
ciscoasa(config)# class gold
```

The *name* is a string up to 20 characters long. To set the limits for the default class, enter **default** for the name.

Step 2 Set the resource limit for a resource type:

limit-resource [**rate**] *resource_name number*[**%**]

Example:

```
ciscoasa(config-class)# limit-resource rate inspects 10
```

- See the preceding table for a list of resource types. If you specify **all**, then all resources are configured with the same value. If you also specify a value for a particular resource, the limit overrides the limit set for **all**.
- Enter the **rate** argument to set the rate per second for certain resources.
- For most resources, specify **0** for the *number* to set the resource to be unlimited or to be the system limit, if available. For VPN resources, **0** sets the limit to none.
- For resources that do not have a system limit, you cannot set the percentage (**%**); you can only set an absolute value.

- If you also set the **quota management-session** command within a context to set the maximum administrative sessions (SSH, etc.), then the lower value will be used.

Example

For example, to set the default class limit for conns to 10 percent instead of unlimited, and to allow 5 site-to-site VPN tunnels with 2 tunnels allowed for VPN burst, enter the following commands:

```
ciscoasa(config)# class default
ciscoasa(config-class)# limit-resource conns 10%
ciscoasa(config-class)# limit-resource vpn other 5
ciscoasa(config-class)# limit-resource vpn burst other 2
```

All other resources remain at unlimited.

To add a class called gold, enter the following commands:

```
ciscoasa(config)# class gold
ciscoasa(config-class)# limit-resource mac-addresses 10000
ciscoasa(config-class)# limit-resource conns 15%
ciscoasa(config-class)# limit-resource rate conns 1000
ciscoasa(config-class)# limit-resource rate inspects 500
ciscoasa(config-class)# limit-resource hosts 9000
ciscoasa(config-class)# limit-resource asdm 5
ciscoasa(config-class)# limit-resource ssh 5
ciscoasa(config-class)# limit-resource rate syslogs 5000
ciscoasa(config-class)# limit-resource telnet 5
ciscoasa(config-class)# limit-resource xlates 36000
ciscoasa(config-class)# limit-resource routes 5000
ciscoasa(config-class)# limit-resource vpn other 10
ciscoasa(config-class)# limit-resource vpn burst other 5
```

When a context is configured with a resource class, a check is made. A warning is generated if the proper licenses were not installed prior to attempting VPN remote-access connections. The administrator must then obtain an AnyConnect Apex license. For example, a warning like the following may appear:

```
ciscoasa(config)# class vpn
ciscoasa(config-class)# limit-resource vpn anyconnect 10.0%
ciscoasa(config-class)# context test
Creating context 'text'...Done. (3)
ciscoasa(config-ctx)# member vpn
WARNING: Multi-mode remote access VPN support requires an AnyConnect Apex license.
Warning: An Access Context license is required for remote-access VPN support in multi-mode.
ciscoasa(config-ctx)#
```

Configure a Security Context

The security context definition in the system configuration identifies the context name, configuration file URL, interfaces that a context can use, and other settings.

Before you begin

- Perform this procedure in the system execution space.

- Configure interfaces. For transparent mode contexts, you cannot share interfaces between contexts, so you might want to use subinterfaces. To plan for Management interface usage, see Management Interface Usage, on page 216.
  - Firepower 1100, Firepower 2100 in Appliance mode, Secure Firewall 3100—Basic Interface Configuration, on page 593.
  - Firepower 2100 in Platform mode—See the getting started guide.
  - Firepower 4100/9300—Logical Devices for the Firepower 4100/9300, on page 163
- If you do not have an admin context (for example, if you clear the configuration) then you must first specify the admin context name by entering the following command:

  ```
  ciscoasa(config)# admin-context name
  ```

  Although this context does not exist yet in your configuration, you can subsequently enter the **context** *name* command to continue the admin context configuration.

Procedure

Step 1 Add or modify a context:

context *name*

Example:

```
ciscoasa(config)# context admin
```

The *name* is a string up to 32 characters long. This name is case sensitive, so you can have two contexts named "customerA" and "CustomerA," for example. You can use letters, digits, or hyphens, but you cannot start or end the name with a hyphen.

Note
"System" or "Null" (in upper or lower case letters) are reserved names, and cannot be used.

Step 2 (Optional) Add a description for this context:

description *text*

Example:

```
ciscoasa(config-ctx)# description Admin Context
```

Step 3 Specify the interfaces you can use in the context:

To allocate an interface:

allocate-interface *interface_id* [*mapped_name*] [**visible** | **invisible**]

To allocate one or more subinterfaces:

allocate-interface *interface_id***.***subinterface* [**-***interface_id***.***subinterface*] [*mapped_name*[**-***mapped_name*]] [**visible** | **invisible**]

Example:

```
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/1.100 int1
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/1.200 int2
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/2.300-gigabitethernet0/2.305
int3-int8
```

Note
Do not include a space between the interface type and the port number.

- Enter these commands multiple times to specify different ranges. If you remove an allocation with the **no** form of this command, then any context commands that include this interface are removed from the running configuration.
- You can assign the same interfaces to multiple contexts in routed mode, if desired. Transparent mode does not allow shared interfaces.
- The *mapped_name* is an alphanumeric alias for the interface that can be used within the context instead of the interface ID. If you do not specify a mapped name, the interface ID is used within the context. For security purposes, you might not want the context administrator to know which interfaces the context is using. A mapped name must start with a letter, end with a letter or digit, and have as interior characters only letters, digits, or an underscore. For example, you can use the following names: **int0**, **inta**, **int_0**.
- If you specify a range of subinterfaces, you can specify a matching range of mapped names. Follow these guidelines for ranges:
  - The mapped name must consist of an alphabetic portion followed by a numeric portion. The alphabetic portion of the mapped name must match for both ends of the range. For example, enter the following range: **int0-int10**. If you enter gig0/1.1-gig0/1.5 **happy**1-**sad**5, for example, the command fails.
  - The numeric portion of the mapped name must include the same quantity of numbers as the subinterface range. For example, both ranges include 100 interfaces:**gigabitethernet0/0.100-gigabitethernet0/0.199 int1-int100**. If you enter gig0/0.**100**-gig0/0.**199** int**1**-int**15**, for example, the command fails.
- Specify **visible** to see the real interface ID in the **show interface** command if you set a mapped name. The default **invisible** keyword shows only the mapped name.

Step 4 Identify the URL from which the system downloads the context configuration:

config-url *url*

You can specify a file in flash memory or on a remote server.

Example:

```
ciscoasa(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/test.cfg
```

Step 5 (Optional) Allow each context to use flash memory to store VPN packages, such as AnyConnect Client, as well as providing storage for AnyConnect Client portal customizations. For example, if you are using multiple context mode to configure an AnyConnect Client profile with Dynamic Access Policies, you must plan for context specific private storage. Each context can use a private storage space as well as a shared read-only storage space. **Note:** Make sure the target directory is already present on the specified disk using the **mkdir** command.

storage-url {**private** | **shared**} [**disk***n***:/**]*path* [*context_label*]

Example:

```
ciscoasa(config)# mkdir disk1:/private-storage
ciscoasa(config)# mkdir disk1:/shared-storage
ciscoasa(config)# context admin
ciscoasa(config-ctx)# storage-url private disk1:/private-storage context
ciscoasa(config-ctx)# storage-url shared disk1:/shared-storage shared
```

You can specify one **private** storage space per context. You can read/write/delete from this directory within the context (as well as from the system execution space). If you do not specify the disk number, the default is disk0. Under the specified *path*, the ASA creates a sub-directory named after the context. For example, for contextA if you specify **disk1:/private-storage** for the path, then the ASA creates a sub-directory for this context at **disk1:/private-storage/contextA/**. You can also optionally name the path within the context with a *context_label*, so that the file system is not exposed to context administrators. For example, if you specify the *context_label* as **context**, then from within the context, this directory is called **context:**. To control how much disk space is allowed per context, see Configure a Class for Resource Management, on page 226.

You can specify one read-only **shared** storage space per context, but you can create multiple shared directories. To reduce duplication of common large files that can be shared among all contexts, such as AnyConnect Client packages, you can use the shared storage space. The ASA does not create context sub-directories for this storage space because it is a shared space for multiple contexts. Only the system execution space can write and delete from the shared directory.

Step 6 (Optional) Assign the context to a resource class:

member *class_name*

Example:

```
ciscoasa(config-ctx)# member gold
```

If you do not specify a class, the context belongs to the default class. You can only assign a context to one resource class.

Step 7 (Optional) Assign a context to a failover group in Active/Active failover:

join-failover-group {**1** | **2**}

Example:

```
ciscoasa(config-ctx)# join-failover-group 2
```

By default, contexts are in group 1. The admin context must always be in group 1.

Step 8 (Optional) Enable Cloud Web Security for this context:

scansafe [**license** *key*]

Example:

```
ciscoasa(config-ctx)# scansafe
```

If you do not specify a **license**, the context uses the license configured in the system configuration. The ASA sends the authentication key to the Cloud Web Security proxy servers to indicate from which organization the request comes. The authentication key is a 16-byte hexidecimal number.

See the firewall configuration guide for detailed information about ScanSafe.

Example

The following example sets the admin context to be "administrator," creates a context called "administrator" on the internal flash memory, and then adds two contexts from an FTP server:

```
ciscoasa(config)# admin-context admin
ciscoasa(config)# context admin
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/0.1
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/1.1
ciscoasa(config-ctx)# config-url disk0:/admin.cfg

ciscoasa(config-ctx)# context test
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/0.100 int1
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/0.102 int2
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/0.110-gigabitethernet0/0.115
int3-int8
ciscoasa(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/test.cfg
ciscoasa(config-ctx)# member gold


ciscoasa(config-ctx)# context sample
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/1.200 int1
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/1.212 int2
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/1.230-gigabitethernet0/1.235
int3-int8
ciscoasa(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/sample.cfg
ciscoasa(config-ctx)# member silver
```

Assign MAC Addresses to Context Interfaces Automatically

This section describes how to configure auto-generation of MAC addresses. The MAC address is used to classify packets within a context.

Before you begin

- When you configure a **nameif** command for the interface in a context, the new MAC address is generated immediately. If you enable this feature after you configure context interfaces, then MAC addresses are generated for all interfaces immediately after you enable it. If you disable this feature, the MAC address for each interface reverts to the default MAC address. For example, subinterfaces of GigabitEthernet 0/1 revert to using the MAC address of GigabitEthernet 0/1.
- In the rare circumstance that the generated MAC address conflicts with another private MAC address in your network, you can manually set the MAC address for the interface within the context.

Procedure

Automatically assign private MAC addresses to each context interface:

mac-address auto [**prefix** *prefix*]

Example:

```
ciscoasa(config)# mac-address auto prefix 19
```

If you do not enter a prefix, then the ASA autogenerates the prefix based on the last two bytes of the interface.

If you manually enter a prefix, then the *prefix* is a decimal value between 0 and 65535. This prefix is converted to a four-digit hexadecimal number, and used as part of the MAC address.

Change Between Contexts and the System Execution Space

If you log in to the system execution space (or the admin context), you can change between contexts and perform configuration and monitoring tasks within each context. The running configuration that you edit in a configuration mode, or that is used in the **copy** or **write** commands, depends on your location. When you are in the system execution space, the running configuration consists only of the system configuration; when you are in a context, the running configuration consists only of that context. For example, you cannot view all running configurations (system plus all contexts) by entering the **show running-config** command. Only the current configuration displays.

Procedure

Step 1 Change to a context:

changeto context *name*

The prompt changes to ciscoasa/*name*#

Step 2 Change to the system execution space:

changeto system

The prompt changes to ciscoasa#

Manage Security Contexts

This section describes how to manage security contexts.

Remove a Security Context

You cannot remove the current admin context, unless you remove all contexts using the **clear context** command.

Note If you use failover, there is a delay between when you remove the context on the active unit and when the context is removed on the standby unit. You might see an error message indicating that the number of interfaces on the active and standby units are not consistent; this error is temporary and can be ignored.

Before you begin

Perform this procedure in the system execution space.

Procedure

Step 1 Remove a single context:

no context *name*

All context commands are also removed. The context configuration file is not removed from the config URL location.

Step 2 Remove all contexts (including the admin context):

clear context

The context configuration files are not removed from the config URL locations.

Change the Admin Context

The system configuration does not include any network interfaces or network settings for itself; rather, when the system needs to access network resources (such as downloading the contexts from the server), it uses one of the contexts that is designated as the admin context.

The admin context is just like any other context, except that when a user logs in to the admin context, then that user has system administrator rights and can access the system and all other contexts. The admin context is not restricted in any way, and can be used as a regular context. However, because logging into the admin context grants you administrator privileges over all contexts, you might need to restrict access to the admin context to appropriate users.

Before you begin

- You can set any context to be the admin context, as long as the configuration file is stored in the internal flash memory.
- Perform this procedure in the system execution space.

Procedure

Set the admin context:

admin-context *context_name*

Example:

```
ciscoasa(config)# admin-context administrator
```

Any remote management sessions, such as Telnet, SSH, or HTTPS, that are connected to the admin context are terminated. You must reconnect to the new admin context.

A few system configuration commands, including **ntp server**, identify an interface name that belongs to the admin context. If you change the admin context, and that interface name does not exist in the new admin context, be sure to update any system commands that refer to the interface.

Change the Security Context URL

This section describes how to change the context URL.

Before you begin

- You cannot change the security context URL without reloading the configuration from the new URL. The ASA merges the new configuration with the current running configuration.
- Reentering the same URL also merges the saved configuration with the running configuration.
- A merge adds any new commands from the new configuration to the running configuration.
  - If the configurations are the same, no changes occur.
  - If commands conflict or if commands affect the running of the context, then the effect of the merge depends on the command. You might get errors, or you might have unexpected results. If the running configuration is blank (for example, if the server was unavailable and the configuration was never downloaded), then the new configuration is used.
- If you do not want to merge the configurations, you can clear the running configuration, which disrupts any communications through the context, and then reload the configuration from the new URL.
- Perform this procedure in the system execution space.

Procedure

Step 1 (Optional, if you do not want to perform a merge) Change to the context and clear configuration:

changeto context *name*

clear configure all

Example:

```
ciscoasa(config)# changeto context ctx1
ciscoasa/ctx1(config)# clear configure all
```

If you want to perform a merge, skip to Step 2.

Step 2 Change to the system execution space:

changeto system

Example:

```
ciscoasa/ctx1(config)# changeto system
ciscoasa(config)#
```

Step 3 Enter the context configuration mode for the context you want to change.

context *name*

Example:

```
ciscoasa(config)# context ctx1
```

Step 4 Enter the new URL. The system immediately loads the context so that it is running.

config-url *new_url*

Example:

```
ciscoasa(config)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/ctx1.cfg
```

Reload a Security Context

You can reload the context in two ways:

- Clear the running configuration and then import the startup configuration.

  This action clears most attributes associated with the context, such as connections and NAT tables.

- Remove the context from the system configuration.

  This action clears additional attributes, such as memory allocation, which might be useful for troubleshooting. However, to add the context back to the system requires you to respecify the URL and interfaces.

Reload by Clearing the Configuration

Procedure

Step 1 Change to the context that you want to reload:

changeto context *name*

Example:

```
ciscoasa(config)# changeto context ctx1
ciscoasa/ctx1(comfig)#
```

Step 2 Clear the running configuration:

clear configure all

This command clears all connections.

Step 3 Reload the configuration:

copy startup-config running-config

Example:

```
ciscoasa/ctx1(config)# copy startup-config running-config
```

The ASA copies the configuration from the URL specified in the system configuration. You cannot change the URL from within a context.

Reload by Removing and Re-adding the Context

To reload the context by removing the context and then re-adding it, perform the steps.

Procedure

Step 1 Remove a Security Context, on page 236. **Also delete config URL file from the disk**

Step 2 Configure a Security Context, on page 230

Monitoring Security Contexts

This section describes how to view and monitor context information.

View Context Information

From the system execution space, you can view a list of contexts including the name, allocated interfaces, and configuration file URL.

Procedure

Show all contexts:

show context [*name* | **detail**| **count**]

If you want to show information for a particular context, specify the *name*.

The **detail** option shows additional information. See the following sample outputs below for more information.

The **count** option shows the total number of contexts.

Example

The following is sample output from the **show context** command. The following sample output shows three contexts:

```
ciscoasa# show context

Context Name      Interfaces                  URL
*admin            GigabitEthernet0/1.100      disk0:/admin.cfg
                  GigabitEthernet0/1.101
contexta          GigabitEthernet0/1.200      disk0:/contexta.cfg
                  GigabitEthernet0/1.201
contextb          GigabitEthernet0/1.300      disk0:/contextb.cfg
                  GigabitEthernet0/1.301
Total active Security Contexts: 3
```

The following table shows each field description.

Table 8: show context Fields

Field	Description
Context Name	Lists all context names. The context name with the asterisk (*) is the admin context.
Interfaces	The interfaces assigned to the context.
URL	The URL from which the ASA loads the context configuration.

The following is sample output from the **show context detail** command:

```
ciscoasa# show context detail

Context "admin", has been created, but initial ACL rules not complete
  Config URL: disk0:/admin.cfg
  Real Interfaces: Management0/0
```

```
  Mapped Interfaces: Management0/0
  Flags: 0x00000013, ID: 1

Context "ctx", has been created, but initial ACL rules not complete
  Config URL: ctx.cfg
  Real Interfaces: GigabitEthernet0/0.10, GigabitEthernet0/1.20,
     GigabitEthernet0/2.30
  Mapped Interfaces: int1, int2, int3
  Flags: 0x00000011, ID: 2

Context "system", is a system resource
  Config URL: startup-config
  Real Interfaces:
  Mapped Interfaces: Control0/0, GigabitEthernet0/0,
     GigabitEthernet0/0.10, GigabitEthernet0/1, GigabitEthernet0/1.10,
     GigabitEthernet0/1.20, GigabitEthernet0/2, GigabitEthernet0/2.30,
     GigabitEthernet0/3, Management0/0, Management0/0.1
  Flags: 0x00000019, ID: 257

Context "null", is a system resource
  Config URL: ... null ...
  Real Interfaces:
  Mapped Interfaces:
  Flags: 0x00000009, ID: 258
```

See the command reference for more information about the **detail** output.

The following is sample output from the **show context count** command:

```
ciscoasa# show context count
Total active contexts: 2
```

View Resource Allocation

From the system execution space, you can view the allocation for each resource across all classes and class members.

Procedure

Show the resource allocation:

show resource allocation [**detail**]

This command shows the resource allocation, but does not show the actual resources being used. See View Resource Usage, on page 244 for more information about actual resource usage.

The **detail** argument shows additional information. See the following sample outputs for more information.

Example

The following sample output shows the total allocation of each resource as an absolute value and as a percentage of the available system resources:

```
ciscoasa# show resource allocation
Resource                    Total       % of Avail
 Conns [rate]               35000           N/A
 Inspects [rate]            35000           N/A
 Syslogs [rate]             10500           N/A
 Conns                     305000           30.50%
 Hosts                      78842           N/A
 SSH                           35           35.00%
 Routes                      5000           N/A
 Telnet                        35           35.00%
 Xlates                     91749           N/A
 AnyConnect                  1000           10%
 AnyConnectBurst              200           2%
 Other VPN Sessions            20           2.66%
 Other VPN Burst               20           2.66%
 All                    unlimited
```

The following table shows each field description.

Table 9: show resource allocation Fields

Field	Description
Resource	The name of the resource that you can limit.
Total	The total amount of the resource that is allocated across all contexts. The amount is an absolute number of concurrent instances or instances per second. If you specified a percentage in the class definition, the ASA converts the percentage to an absolute number for this display.
% of Avail	The percentage of the total system resources that is allocated across all contexts, if the resource has a hard system limit. If a resource does not have a system limit, this column shows N/A.

The following is sample output from the **show resource allocation detail** command:

```
ciscoasa# show resource allocation detail
Resource Origin:
    A    Value was derived from the resource 'all'
    C    Value set in the definition of this class
    D    Value set in default class
Resource         Class          Mmbrs  Origin      Limit      Total    Total %
Conns [rate]     default          all      CA  unlimited
                 gold               1       C      34000      34000     N/A
                 silver             1      CA      17000      17000     N/A
                 bronze             0      CA       8500
                 All Contexts:      3                         51000     N/A

Inspects [rate]  default          all      CA  unlimited
                 gold               1      DA  unlimited
                 silver             1      CA      10000      10000     N/A
                 bronze             0      CA       5000
                 All Contexts:      3                         10000     N/A

Syslogs [rate]   default          all      CA  unlimited
                 gold               1       C       6000       6000     N/A
                 silver             1      CA       3000       3000     N/A
                 bronze             0      CA       1500
```

```
                All Contexts:      3                        9000     N/A

Conns           default          all      CA  unlimited
                gold               1       C     200000     200000     20.00%
                silver             1      CA     100000     100000     10.00%
                bronze             0      CA      50000
                All Contexts:      3                       300000     30.00%

Hosts           default          all      CA  unlimited
                gold               1      DA  unlimited
                silver             1      CA      26214      26214      N/A
                bronze             0      CA      13107
                All Contexts:      3                       26214      N/A

SSH             default          all       C          5
                gold               1       D          5          5      5.00%
                silver             1      CA         10         10     10.00%
                bronze             0      CA          5
                All Contexts:      3                           20     20.00%

Telnet          default          all       C          5
                gold               1       D          5          5      5.00%
                silver             1      CA         10         10     10.00%
                bronze             0      CA          5
                All Contexts:      3                           20     20.00%

Routes          default          all       C  unlimited                 N/A
                gold               1       D  unlimited          5      N/A
                silver             1      CA         10         10      N/A
                bronze             0      CA          5                 N/A
                All Contexts:      3                           20      N/A

Xlates          default          all      CA  unlimited
                gold               1      DA  unlimited
                silver             1      CA      23040      23040     N/A
                bronze             0      CA      11520
                All Contexts:      3                       23040     N/A

mac-addresses   default          all       C      65535
                gold               1       D      65535      65535    100.00%
                silver             1      CA       6553       6553      9.99%
                bronze             0      CA       3276
                All Contexts:      3                       137623    209.99%
```

The following table shows each field description.

Table 10: show resource allocation detail Fields

Field	Description
Resource	The name of the resource that you can limit.
Class	The name of each class, including the default class. The All contexts field shows the total values across all classes.
Mmbrs	The number of contexts assigned to each class.

Field	Description
Origin	The origin of the resource limit, as follows: • A—You set this limit with the **all** option, instead of as an individual resource. • C—This limit is derived from the member class. • D—This limit was not defined in the member class, but was derived from the default class. For a context assigned to the default class, the value will be "C" instead of "D." The ASA can combine "A" with "C" or "D."
Limit	The limit of the resource per context, as an absolute number. If you specified a percentage in the class definition, the ASA converts the percentage to an absolute number for this display.
Total	The total amount of the resource that is allocated across all contexts in the class. The amount is an absolute number of concurrent instances or instances per second. If the resource is unlimited, this display is blank.
% of Avail	The percentage of the total system resources that is allocated across all contexts in the class. If the resource is unlimited, this display is blank. If the resource does not have a system limit, then this column shows N/A.

View Resource Usage

From the system execution space, you can view the resource usage for each context and display the system resource usage.

Procedure

View resource usage for each context:

show resource usage [**context** *context_name* | **top** *n* | **all** | **summary** | **system**] [**resource** {*resource_name* | **all**} | **detail**] [**counter** *counter_name* [*count_threshold*]]

- By default, **all** context usage is displayed; each context is listed separately.
- Enter the **top** *n* keyword to show the contexts that are the top *n* users of the specified resource. You must specify a single resource type, and not **resource all**, with this option.
- The **summary** option shows all context usage combined.
- The **system** option shows all context usage combined, but shows the system limits for resources instead of the combined context limits.
- For the **resource** *resource_name*, see Configure a Class for Resource Management, on page 226 for available resource names. See also the **show resource type** command. Specify **all** (the default) for all types.

- The **detail** option shows the resource usage of all resources, including those you cannot manage. For example, you can view the number of TCP intercepts.
- The **counter** *counter_name* is one of the following keywords:
  - **current**—Shows the active concurrent instances or the current rate of the resource.
  - **denied**—Shows the number of instances that were denied because they exceeded the resource limit shown in the Limit column.
  - **peak**—Shows the peak concurrent instances, or the peak rate of the resource since the statistics were last cleared, either using the **clear resource usage** command or because the device rebooted.
  - **all**—(Default) Shows all statistics.
- The *count_threshold* sets the number above which resources are shown. The default is 1. If the usage of the resource is below the number you set, then the resource is not shown. If you specify **all** for the counter name, then the *count_threshold* applies to the current usage.
- To show all resources, set the *count_threshold* to **0**.

Examples

The following is sample output from the **show resource usage context** command, which shows the resource usage for the admin context:

```
ciscoasa# show resource usage context admin

Resource              Current         Peak      Limit    Denied  Context
Telnet                      1            1          5         0  admin
Conns                      44           55        N/A         0  admin
Hosts                      45           56        N/A         0  admin
```

The following is sample output from the **show resource usage summary** command, which shows the resource usage for all contexts and all resources. This sample shows the limits for six contexts.

```
ciscoasa# show resource usage summary

Resource              Current         Peak      Limit     Denied Context
Syslogs [rate]           1743         2132      N/A            0 Summary
Conns                     584          763      280000(S)      0 Summary
Xlates                   8526         8966      N/A            0 Summary
Hosts                     254          254      N/A            0 Summary
Conns [rate]              270          535      N/A         1704 Summary
Inspects [rate]           270          535      N/A            0 Summary
AnyConnect                  2           25      1000           0 Summary
AnyConnectBurst             0            0      200            0 Summary
Other VPN Sessions          0           10      10           740 Summary
Other VPN Burst             0           10      10           730 Summary

S = System: Combined context limits exceed the system limit; the system limit is shown.
```

The following is sample output from the **show resource usage summary** command, which shows the limits for 25 contexts. Because the context limit for Telnet and SSH connections is 5 per context, then the combined limit is 125. The system limit is only 100, so the system limit is shown.

```
ciscoasa# show resource usage summary

Resource              Current         Peak      Limit    Denied   Context
Telnet                      1            1        100[S]      0   Summary
SSH                         2            2        100[S]      0   Summary
Conns                      56           90        130000(S)   0   Summary
Hosts                      89          102        N/A         0   Summary
S = System: Combined context limits exceed the system limit; the system limit is shown.
```

The following is sample output from the **show resource usage system** command, which shows the resource usage for all contexts, but it shows the system limit instead of the combined context limits. The **counter all 0** option is used to show resources that are not currently in use. The Denied statistics indicate how many times the resource was denied due to the system limit, if available.

```
ciscoasa# show resource usage system counter all 0

Resource              Current         Peak      Limit        Denied    Context
Telnet                      0            0        100             0    System
SSH                         0            0        100             0    System
ASDM                        0            0         32             0    System
Routes                      0            0        N/A             0    System
IPSec                       0            0          5             0    System
Syslogs [rate]              1           18        N/A             0    System
Conns                       0            1     280000             0    System
Xlates                      0            0        N/A             0    System
Hosts                       0            2        N/A             0    System
Conns [rate]                1            1        N/A             0    System
Inspects [rate]             0            0        N/A             0    System
AnyConnect                  2           25      10000             0    System
AnyConnectBurst             0            0        200             0    System
Other VPN Sessions          0           10        750           740    System
Other VPN Burst             0           10        750           730    System
```

Monitor SYN Attacks in Contexts

The ASA prevents SYN attacks using TCP Intercept. TCP Intercept uses the SYN cookies algorithm to prevent TCP SYN-flooding attacks. A SYN-flooding attack consists of a series of SYN packets usually originating from spoofed IP addresses. The constant flood of SYN packets keeps the server SYN queue full, which prevents it from servicing connection requests. When the embryonic connection threshold of a connection is crossed, the ASA acts as a proxy for the server and generates a SYN-ACK response to the client SYN request. When the ASA receives an ACK back from the client, it can then authenticate the client and allow the connection to the server.

Procedure

Step 1 Monitor the rate of attacks for individual contexts:

show perfmon

Step 2 Monitor the amount of resources being used by TCP intercept for individual contexts:

show resource usage detail

Step 3 Monitor the resources being used by TCP intercept for the entire system:

show resource usage summary detail

Examples

The following is sample output from the **show perfmon** command that shows the rate of TCP intercepts for a context called admin.

```
ciscoasa/admin# show perfmon

Context:admin
PERFMON STATS:   Current      Average
Xlates               0/s          0/s
Connections          0/s          0/s
TCP Conns            0/s          0/s
UDP Conns            0/s          0/s
URL Access           0/s          0/s
URL Server Req       0/s          0/s
WebSns Req           0/s          0/s
TCP Fixup            0/s          0/s
HTTP Fixup           0/s          0/s
FTP Fixup            0/s          0/s
AAA Authen           0/s          0/s
AAA Author           0/s          0/s
AAA Account          0/s          0/s
TCP Intercept    322779/s      322779/s
```

The following is sample output from the **show resource usage detail** command that shows the amount of resources being used by TCP Intercept for individual contexts. (Sample text in **bold** shows the TCP intercept information.)

```
ciscoasa(config)# show resource usage detail
Resource              Current         Peak      Limit        Denied Context
memory                 843732       847288  unlimited             0 admin
chunk:channels             14           15  unlimited             0 admin
chunk:fixup                15           15  unlimited             0 admin
chunk:hole                  1            1  unlimited             0 admin
chunk:ip-users             10           10  unlimited             0 admin
chunk:list-elem            21           21  unlimited             0 admin
chunk:list-hdr              3            4  unlimited             0 admin
chunk:route                 2            2  unlimited             0 admin
chunk:static                1            1  unlimited             0 admin
tcp-intercepts         328787       803610  unlimited             0 admin
np-statics                  3            3  unlimited             0 admin
statics                     1            1  unlimited             0 admin
ace-rules                   1            1  unlimited             0 admin
console-access-rul          2            2  unlimited             0 admin
fixup-rules                14           15  unlimited             0 admin
memory                 959872       960000  unlimited             0 c1
chunk:channels             15           16  unlimited             0 c1
chunk:dbgtrace              1            1  unlimited             0 c1
chunk:fixup                15           15  unlimited             0 c1
```

```
chunk:global                1            1  unlimited            0 c1
chunk:hole                  2            2  unlimited            0 c1
chunk:ip-users             10           10  unlimited            0 c1
chunk:udp-ctrl-blk          1            1  unlimited            0 c1
chunk:list-elem            24           24  unlimited            0 c1
chunk:list-hdr              5            6  unlimited            0 c1
chunk:nat                   1            1  unlimited            0 c1
chunk:route                 2            2  unlimited            0 c1
chunk:static                1            1  unlimited            0 c1
tcp-intercept-rate      16056        16254  unlimited            0 c1
globals                     1            1  unlimited            0 c1
np-statics                  3            3  unlimited            0 c1
statics                     1            1  unlimited            0 c1
nats                        1            1  unlimited            0 c1
ace-rules                   2            2  unlimited            0 c1
console-access-rul          2            2  unlimited            0 c1
fixup-rules                14           15  unlimited            0 c1
memory              232695716    232020648  unlimited            0 system
chunk:channels             17           20  unlimited            0 system
chunk:dbgtrace              3            3  unlimited            0 system
chunk:fixup                15           15  unlimited            0 system
chunk:ip-users              4            4  unlimited            0 system
chunk:list-elem          1014         1014  unlimited            0 system
chunk:list-hdr              1            1  unlimited            0 system
chunk:route                 1            1  unlimited            0 system
block:16384               510          885  unlimited            0 system
block:2048                 32           34  unlimited            0 system
```

The following sample output shows the resources being used by TCP intercept for the entire system. (Sample text in **bold** shows the TCP intercept information.)

```
ciscoasa(config)# show resource usage summary detail
Resource              Current         Peak      Limit        Denied Context
memory              238421312    238434336  unlimited            0 Summary
chunk:channels             46           48  unlimited            0 Summary
chunk:dbgtrace              4            4  unlimited            0 Summary
chunk:fixup                45           45  unlimited            0 Summary
chunk:global                1            1  unlimited            0 Summary
chunk:hole                  3            3  unlimited            0 Summary
chunk:ip-users             24           24  unlimited            0 Summary
chunk:udp-ctrl-blk          1            1  unlimited            0 Summary
chunk:list-elem          1059         1059  unlimited            0 Summary
chunk:list-hdr             10           11  unlimited            0 Summary
chunk:nat                   1            1  unlimited            0 Summary
chunk:route                 5            5  unlimited            0 Summary
chunk:static                2            2  unlimited            0 Summary
block:16384               510          885  unlimited            0 Summary
block:2048                 32           35  unlimited            0 Summary
tcp-intercept-rate     341306       811579  unlimited            0 Summary
globals                     1            1  unlimited            0 Summary
np-statics                  6            6  unlimited            0 Summary
statics                     2            2        N/A            0 Summary
nats                        1            1        N/A            0 Summary
ace-rules                   3            3        N/A            0 Summary
console-access-rul          4            4        N/A            0 Summary
fixup-rules                43           44        N/A            0 Summary
```

View Assigned MAC Addresses

You can view auto-generated MAC addresses within the system configuration or within the context.

View MAC Addresses in the System Configuration

This section describes how to view MAC addresses in the system configuration.

Before you begin

If you manually assign a MAC address to an interface, but also have auto-generation enabled, the auto-generated address continues to show in the configuration even though the manual MAC address is the one that is in use. If you later remove the manual MAC address, the auto-generated one shown will be used.

Procedure

Show the assigned MAC addresses from the system execution space:

show running-config all context [*name*]

The **all** option is required to view the assigned MAC addresses. Although the **mac-address auto** command is user-configurable in global configuration mode only, the command appears as a read-only entry in context configuration mode along with the assigned MAC address. Only allocated interfaces that are configured with a **nameif** command within the context have a MAC address assigned.

Examples

The following output from the **show running-config all context admin** command shows the primary and standby MAC address assigned to the Management0/0 interface:

```
ciscoasa# show running-config all context admin

context admin
  allocate-interface Management0/0
  mac-address auto Management0/0 a24d.0000.1440 a24d.0000.1441
  config-url disk0:/admin.cfg
```

The following output from the **show running-config all context** command shows all the MAC addresses (primary and standby) for all context interfaces. Note that because the GigabitEthernet0/0 and GigabitEthernet0/1 main interfaces are not configured with a **nameif** command inside the contexts, no MAC addresses have been generated for them.

```
ciscoasa# show running-config all context

admin-context admin
context admin
  allocate-interface Management0/0
  mac-address auto Management0/0 a2d2.0400.125a a2d2.0400.125b
  config-url disk0:/admin.cfg
!
```

```
context CTX1
  allocate-interface GigabitEthernet0/0
  allocate-interface GigabitEthernet0/0.1-GigabitEthernet0/0.5
  mac-address auto GigabitEthernet0/0.1 a2d2.0400.11bc a2d2.0400.11bd
  mac-address auto GigabitEthernet0/0.2 a2d2.0400.11c0 a2d2.0400.11c1
  mac-address auto GigabitEthernet0/0.3 a2d2.0400.11c4 a2d2.0400.11c5
  mac-address auto GigabitEthernet0/0.4 a2d2.0400.11c8 a2d2.0400.11c9
  mac-address auto GigabitEthernet0/0.5 a2d2.0400.11cc a2d2.0400.11cd
  allocate-interface GigabitEthernet0/1
  allocate-interface GigabitEthernet0/1.1-GigabitEthernet0/1.3
  mac-address auto GigabitEthernet0/1.1 a2d2.0400.120c a2d2.0400.120d
  mac-address auto GigabitEthernet0/1.2 a2d2.0400.1210 a2d2.0400.1211
  mac-address auto GigabitEthernet0/1.3 a2d2.0400.1214 a2d2.0400.1215
  config-url disk0:/CTX1.cfg
!

context CTX2
  allocate-interface GigabitEthernet0/0
  allocate-interface GigabitEthernet0/0.1-GigabitEthernet0/0.5
  mac-address auto GigabitEthernet0/0.1 a2d2.0400.11ba a2d2.0400.11bb
  mac-address auto GigabitEthernet0/0.2 a2d2.0400.11be a2d2.0400.11bf
  mac-address auto GigabitEthernet0/0.3 a2d2.0400.11c2 a2d2.0400.11c3
  mac-address auto GigabitEthernet0/0.4 a2d2.0400.11c6 a2d2.0400.11c7
  mac-address auto GigabitEthernet0/0.5 a2d2.0400.11ca a2d2.0400.11cb
  allocate-interface GigabitEthernet0/1
  allocate-interface GigabitEthernet0/1.1-GigabitEthernet0/1.3
  mac-address auto GigabitEthernet0/1.1 a2d2.0400.120a a2d2.0400.120b
  mac-address auto GigabitEthernet0/1.2 a2d2.0400.120e a2d2.0400.120f
  mac-address auto GigabitEthernet0/1.3 a2d2.0400.1212 a2d2.0400.1213
  config-url disk0:/CTX2.cfg
!
```

View MAC Addresses Within a Context

This section describes how to view MAC addresses within a context.

Procedure

Show the MAC address in use by each interface within the context:

show interface | include (Interface)|(MAC)

Example

For example:

```
ciscoasa/context# show interface | include (Interface)|(MAC)

Interface GigabitEthernet1/1.1 "g1/1.1", is down, line protocol is down
        MAC address a201.0101.0600, MTU 1500
Interface GigabitEthernet1/1.2 "g1/1.2", is down, line protocol is down
        MAC address a201.0102.0600, MTU 1500
Interface GigabitEthernet1/1.3 "g1/1.3", is down, line protocol is down
        MAC address a201.0103.0600, MTU 1500
...
```

Note The **show interface** command shows the MAC address in use; if you manually assign a MAC address and also have auto-generation enabled, then you can only view the unused auto-generated address from within the system configuration.

Examples for Multiple Context Mode

The following example:

- Automatically sets the MAC addresses in contexts with a custom prefix.
- Sets the default class limit for conns to 10 percent instead of unlimited, and sets the VPN other sessions to 10, with a burst of 5.
- Creates a gold resource class.
- Sets the admin context to be "administrator."
- Creates a context called "administrator" on the internal flash memory to be part of the default resource class.
- Adds two contexts from an FTP server as part of the gold resource class.

```
ciscoasa(config)# mac-address auto prefix 19

ciscoasa(config)# class default
ciscoasa(config-class)# limit-resource conns 10%
ciscoasa(config-class)# limit-resource vpn other 10
ciscoasa(config-class)# limit-resource vpn burst other 5

ciscoasa(config)# class gold
ciscoasa(config-class)# limit-resource mac-addresses 10000
ciscoasa(config-class)# limit-resource conns 15%
ciscoasa(config-class)# limit-resource rate conns 1000
ciscoasa(config-class)# limit-resource rate inspects 500
ciscoasa(config-class)# limit-resource hosts 9000
ciscoasa(config-class)# limit-resource asdm 5
ciscoasa(config-class)# limit-resource ssh 5
ciscoasa(config-class)# limit-resource rate syslogs 5000
ciscoasa(config-class)# limit-resource telnet 5
ciscoasa(config-class)# limit-resource xlates 36000
ciscoasa(config-class)# limit-resource routes 700
ciscoasa(config-class)# limit-resource vpn other 100
ciscoasa(config-class)# limit-resource vpn burst other 50

ciscoasa(config)# admin-context administrator
ciscoasa(config)# context administrator
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/0.1
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/1.1
ciscoasa(config-ctx)# config-url disk0:/admin.cfg

ciscoasa(config-ctx)# context test
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/0.100 int1
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/0.102 int2
```

```
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/0.110-gigabitethernet0/0.115
int3-int8
ciscoasa(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/test.cfg
ciscoasa(config-ctx)# member gold

ciscoasa(config-ctx)# context sample
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/1.200 int1
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/1.212 int2
ciscoasa(config-ctx)# allocate-interface gigabitethernet0/1.230-gigabitethernet0/1.235
int3-int8
ciscoasa(config-ctx)# config-url ftp://user1:passw0rd@10.1.1.1/configlets/sample.cfg
ciscoasa(config-ctx)# member gold
```

History for Multiple Context Mode

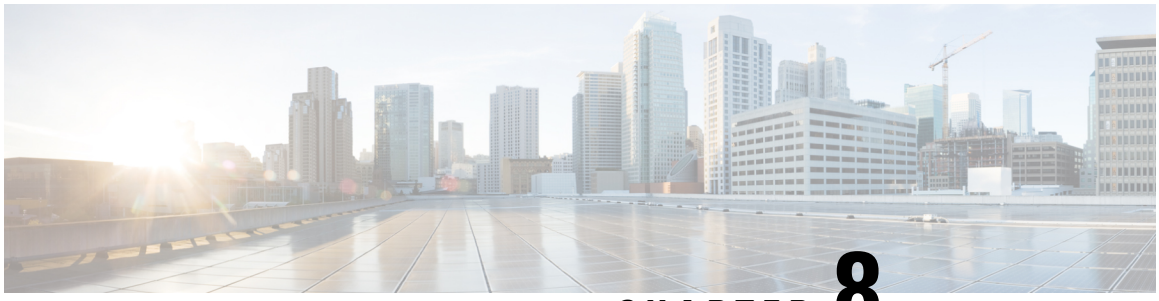
Table 11: History for Multiple Context Mode

Feature Name	Platform Releases	Feature Information
Multiple security contexts	7.0(1)	Multiple context mode was introduced. We introduced the following commands: **context**, **mode**, and **class**.
Automatic MAC address assignment	7.2(1)	Automatic assignment of MAC address to context interfaces was introduced. We introduced the following command: **mac-address auto**.
Resource management	7.2(1)	Resource management was introduced. We introduced the following commands: **class**, **limit-resource**, and **member**.
Virtual sensors for IPS	8.0(2)	The AIP SSM running IPS software Version 6.0 and above can run multiple virtual sensors, which means you can configure multiple security policies on the AIP SSM. You can assign each context or single mode ASA to one or more virtual sensors, or you can assign multiple security contexts to the same virtual sensor. We introduced the following command: **allocate-ips**.
Automatic MAC address assignment enhancements	8.0(5)/8.2(2)	The MAC address format was changed to use a prefix, to use a fixed starting value (A2), and to use a different scheme for the primary and secondary unit MAC addresses in a failover pair. The MAC addresses are also now persistent across reloads. The command parser now checks if auto-generation is enabled; if you want to also manually assign a MAC address, you cannot start the manual MAC address with A2. We modified the following command: **mac-address auto prefix**.
Maximum contexts increased for the ASA 5550 and 5580	8.4(1)	The maximum security contexts for the ASA 5550 was increased from 50 to 100. The maximum for the ASA 5580 was increased from 50 to 250.
Automatic MAC address assignment enabled by default	8.5(1)	Automatic MAC address assignment is now enabled by default. We modified the following command: **mac-address auto**.

Feature Name	Platform Releases	Feature Information
Automatic generation of a MAC address prefix	8.6(1)	In multiple context mode, the ASA now converts the automatic MAC address generation configuration to use a default prefix. The ASA auto-generates the prefix based on the last two bytes of the interface (ASA 5500-X) or backplane (ASASM) MAC address. This conversion happens automatically when you reload, or if you reenable MAC address generation. The prefix method of generation provides many benefits, including a better guarantee of unique MAC addresses on a segment. You can view the auto-generated prefix by entering the **show running-config mac-address** command. If you want to change the prefix, you can reconfigure the feature with a custom prefix. The legacy method of MAC address generation is no longer available. **Note** To maintain hitless upgrade for failover pairs, the ASA does *not* convert the MAC address method in an existing configuration upon a reload if failover is enabled. However, we strongly recommend that you manually change to the prefix method of generation when using failover, especially for the ASASM. Without the prefix method, ASASMs installed in different slot numbers experience a MAC address change upon failover, and can experience traffic interruption. After upgrading, to use the prefix method of MAC address generation, reenable MAC address generation to use the default prefix. We modified the following command: **mac-address auto**.
Automatic MAC address assignment disabled by default on all models except for the ASASM	9.0(1)	Automatic MAC address assignment is now disabled by default except for the ASASM. We modified the following command: **mac-address auto**.
Dynamic routing in Security Contexts	9.0(1)	EIGRP and OSPFv2 dynamic routing protocols are now supported in multiple context mode. OSPFv3, RIP, and multicast routing are not supported.
New resource type for routing table entries	9.0(1)	A new resource type, routes, was created to set the maximum number of routing table entries in each context. We modified the following commands: **limit-resource, show resource types, show resource usage, show resource allocation**.
Site-to-Site VPN in multiple context mode	9.0(1)	Site-to-site VPN tunnels are now supported in multiple context mode.
New resource type for site-to-site VPN tunnels	9.0(1)	New resource types, vpn other and vpn burst other, were created to set the maximum number of site-to-site VPN tunnels in each context. We modified the following commands: **limit-resource, show resource types, show resource usage, show resource allocation**.
New resource type for IKEv1 SA negotiations	9.1(2)	New resource type, ikev1 in-negotiation, was created to set the maximum percentage of IKEv1 SA negotiations in each context to prevent overwhelming the CPU and crypto engines. Under certain conditions (large certificates, CRL checking), you might want to restrict this resource. We modified the following commands: **limit-resource, show resource types, show resource usage, show resource allocation**.

Feature Name	Platform Releases	Feature Information
Support for Remote Access VPN in multiple context mode	9.5(2)	You can now use the following remote access features in multiple context mode: • AnyConnect 3.x and later (SSL VPN only; no IKEv2 support) • Centralized AnyConnect Client image configuration • AnyConnect Client image upgrade • Context Resource Management for AnyConnect Client connections **Note** The AnyConnect Apex license is required for multiple context mode; you cannot use the default or legacy license. We introduced the following commands: **limit-resource vpn anyconnect, limit-resource vpn burst anyconnect**
Pre-fill/Username-from-cert feature for multiple context mode	9.6(2)	AnyConnect Client SSL support is extended, allowing pre-fill/username-from-certificate feature CLIs, previously available only in single mode, to be enabled in multiple context mode as well. We did not modify any commands.
Flash Virtualization for Remote Access VPN	9.6(2)	Remote access VPN in multiple context mode now supports flash virtualization. Each context can have a private storage space and a shared storage place based on the total flash that is available: • Private storage—Store files associated only with that user and specific to the content that you want for that user. • Shared storage—Upload files to this space and have it accessible to any user context for read/write access once you enable it. We introduced the following commands: **limit-resource storage, storage-url**
AnyConnect Clientprofiles supported in multi-context devices	9.6(2)	AnyConnect Client profiles are supported in multi-context devices. To add a new profile using ASDM, you must have the AnyConnect Client release 4.2.00748 or 4.3.03013 and later.
Stateful failover for AnyConnect Client connections in multiple context mode	9.6(2)	Stateful failover is now supported for AnyConnect Client connections in multiple context mode. We did not modify any commands.
Remote Access VPN Dynamic Access Policy (DAP) is supported in multiple context mode	9.6(2)	You can now configure DAP per context in multiple context mode. We did not modify any commands.
Remote Access VPN CoA (Change of Authorization) is supported in multiple context mode	9.6(2)	You can now configure CoA per context in multiple context mode. We did not modify any commands.

Feature Name	Platform Releases	Feature Information
Remote Access VPN localization is supported in multiple context mode	9.6(2)	Localization is supported globally. There is only one set of localization files that are shared across different contexts. We did not modify any commands.
Remote Access VPN for IKEv2 is supported in multiple context mode	9.9(2)	You can configure Remote Access VPN in multiple context mode for IKEv2.
Configurable limitation of admin sessions	9.12(1)	You can configure the maximum number of aggregate, per user, and per-protocol administrative sessions. Formerly, you could configure only the aggregate number of sessions. This feature does not affect console sessions. Note that in multiple context mode, you cannot configure the number of HTTPS sessions, where the maximum is fixed at 5 sessions. The **quota management-session** command is also no longer accepted in the system configuration, and is instead available in the context configuration. The maximum aggregate sessions is now 15; if you configured 0 (unlimited) or 16+, then when you upgrade, the value is changed to 15. New/Modified commands: **quota management-session**, **show quota management-session**
Firepower 1140 maximum contexts increased from 5 to 10	9.16(1)	The Firepower 1140 now supports up to 10 contexts.



CHAPTER 8

Failover for High Availability

This chapter describes how to configure Active/Standby or Active/Active failover to accomplish high availability of the ASA.

- About Failover, on page 257
- Licensing for Failover, on page 279
- Guidelines for Failover, on page 280
- Defaults for Failover, on page 283
- Configure Active/Standby Failover, on page 283
- Configure Active/Active Failover, on page 288
- Configure Optional Failover Parameters, on page 294
- Manage Failover, on page 302
- Monitoring Failover, on page 308
- History for Failover, on page 309

About Failover

Configuring failover requires two identical ASAs connected to each other through a dedicated failover link and, optionally, a state link. The health of the active units and interfaces is monitored to determine whether they meet the specific failover conditions. If those conditions are met, failover occurs.

Failover Modes

The ASA supports two failover modes, Active/Active failover and Active/Standby failover. Each failover mode has its own method for determining and performing failover.

- In Active/Standby failover, one device functions as the Active unit and passes traffic. The second device, designated as the Standby unit, does not actively pass traffic. When a failover occurs, the Active unit fails over to the Standby unit, which then becomes Active. You can use Active/Standby failover for ASAs in single or multiple context mode.
- In an Active/Active failover configuration, both ASAs can pass network traffic. Active/Active failover is only available to ASAs in multiple context mode. In Active/Active failover, you divide the security contexts on the ASA into 2 *failover groups*. A failover group is simply a logical group of one or more security contexts. One group is assigned to be Active on the primary ASA, and the other group is assigned to be active on the Secondary ASA. When a failover occurs, it occurs at the failover group level.

Both failover modes support stateful or stateless failover.

Failover system requirements

Failover system requirements are specifications that define the hardware, software, and license prerequisites for ASAs to function in a Failover configuration.

Hardware requirements

Hardware requirements are specifications that ensure two units in a Failover configuration can

- maintain identical hardware configurations including model, interfaces, modules, and RAM
- support proper synchronization and failover functionality, and
- ensure reliable high availability operation.

Hardware requirement specifications

The two units in a Failover configuration must meet these requirements:

- Be the same model.

  For the Firepower 9300, High Availability is only supported between same-type modules. However, the two chassis can include mixed modules. For example, each chassis has an SM-56, SM-48, and SM-40. You can create High Availability pairs between the SM-56 modules, between the SM-48 modules, and between the SM-40 modules.

- Have the same number and types of interfaces.

  For the Firepower 2100 in Platform mode and Firepower 4100/9300 chassis, all interfaces must be preconfigured in FXOS identically before you enable Failover. If you change the interfaces after you enable Failover, make the interface changes in FXOS on the Standby unit, and then make the same changes on the Active unit. If you remove an interface in FXOS (for example, if you remove a network module, remove an EtherChannel, or reassign an interface to an EtherChannel), then the ASA configuration retains the original commands so that you can make any necessary adjustments; removing an interface from the configuration can have wide effects. You can manually remove the old interface configuration in the ASA OS.

- Have the same modules installed (if any).
- Have the same RAM installed.

If you are using units with different flash memory sizes in your Failover configuration, make sure the unit with the smaller flash memory has enough space to accommodate the software image files and the configuration files. If it does not, configuration synchronization from the unit with the larger flash memory to the unit with the smaller flash memory will fail.

Software requirements

Software requirements are configuration specifications that Failover units must meet to function properly in a failover configuration.

Required software specifications

The two units in a Failover configuration must meet these requirements:

- Be in the same context mode (single or multiple).
- For single mode: Be in the same firewall mode (routed or transparent).

  In multiple context mode, the firewall mode is set at the context-level, and you can use mixed modes.
- Have the same major (first number) and minor (second number) software version. However, you can temporarily use different versions of the software during an upgrade process; for example, you can upgrade one unit from Version 8.3(1) to Version 8.3(2) and have failover remain active. We recommend upgrading both units to the same version to ensure long-term compatibility.
- Have the same AnyConnect Client images. If the failover pair has mismatched images when a hitless upgrade is performed, then the clientless SSL VPN connection terminates in the final reboot step of the upgrade process, the database shows an orphaned session, and the IP pool shows that the IP address assigned to the client is "in use."
- Be in the same FIPS mode.
- (Firepower 4100/9300) Have the same flow offload mode, either both enabled or both disabled.

License Requirements

The two units in a failover configuration do not need to have identical licenses; the licenses combine to make a failover cluster license.

Failover and stateful failover links

A failover link and stateful failover link are dedicated connections that

- establish communication between the two units in a failover configuration
- require the same interface to be used between corresponding devices for consistency, and
- transmit critical failover and state information between the paired units.

Configuration recommendations

Cisco recommends to use the same interface between two devices in a failover link or a stateful failover link. For example, in a failover link, if you have used eth0 in device 1, use the same interface (eth0) in device 2 as well.



Caution All information sent over the failover and state links is sent in clear text unless you secure the communication with an IPsec tunnel or a failover key. If the ASA is used to terminate VPN tunnels, this information includes any usernames, passwords and preshared keys used for establishing the tunnels. Transmitting this sensitive data in clear text could pose a significant security risk. We recommend securing the failover communication with an IPsec tunnel or a failover key if you are using the ASA to terminate VPN tunnels.

Failover link

A failover link is a communication mechanism that enables two units in a failover pair to constantly communicate to determine the operating status of each unit.

Failover link data

This below information is communicated over the failover link:

- The unit state (active or standby)
- Hello messages (keep-alives)
- Network link status
- MAC address exchange
- Configuration replication and synchronization

Interface for the failover link

A failover link interface is a dedicated communication channel that

- exists solely for failover communication between high availability units
- uses an unused data interface (physical, subinterface, or EtherChannel)
- cannot be shared with normal networking interfaces or user data traffic, and
- requires specific interface sizing based on the device model.

Interface restrictions and requirements

You can use an unused data interface (physical, subinterface, or EtherChannel) as the failover link; however, you cannot specify an interface that is currently configured with a name. The failover link interface is not configured as a normal networking interface; it exists for failover communication only. This interface can only be used for the failover link (and also for the state link). For most models, you cannot use a management interface for failover unless explicitly described below.

The ASA does not support sharing interfaces between user data and the failover link. You also cannot use separate subinterfaces on the same parent for the failover link and for data.

See these guidelines for the failover link:

- 5506-X through 5555-X—You cannot use the Management interface as the failover link; you must use a data interface. The only exception is for the 5506H-X, where you can use the management interface as the failover link.
- 5506H-X—You *can* use the Management 1/1 interface as the failover link. If you configure it for failover, you must reload the device for the change to take effect. In this case, you cannot also use the ASA Firepower module, because it requires the Management interface for management purposes.
- Firepower 4100/9300—You cannot use the management-type interface for the failover link.
- See these guidelines for sizing the link.

Table 12: Failover link size

Model	Interface Size for Combined Failover and State Link
Firepower 1010	1 Gbps
Firepower 1100	1 Gbps
Firepower 2100	1 Gbps
Secure Firewall 3100	Secure Firewall 3110—1 Gbps Secure Firewall 3120—1 Gbps Secure Firewall 3130—10 Gbps Secure Firewall 3140—10 Gbps
Firepower 4100	10 Gbps
Firepower 9300	10 Gbps

The alternation frequency is equal to the unit hold time (the **failover polltime unit** command).

Note If you have a large configuration and a low unit hold time, alternating between the member interfaces can prevent the secondary unit from joining/re-joining. In this case, disable one of the member interfaces until after the secondary unit joins.

For an EtherChannel used as the failover link, to prevent out-of-order packets, only one interface in the EtherChannel is used. If that interface fails, then the next interface in the EtherChannel is used. You cannot alter the EtherChannel configuration while it is in use as a failover link.

Connect the failover link

The failover link enables communication and coordination between units in a failover configuration.

You need to establish a physical connection between the failover units to enable proper failover functionality.

Procedure

Connect the failover link using one of the following methods:

- Using a switch, with no other device on the same network segment (broadcast domain or VLAN) as the failover interfaces of the ASA.
- Using an Ethernet cable to connect the units directly, without the need for an external switch.

If you do not use a switch between the units and the interface fails, the link is brought down on both peers. This condition may hamper troubleshooting efforts because you cannot easily determine which unit has the failed interface and caused the link to come down.

The ASA supports Auto-MDI/MDIX on its copper Ethernet ports, so you can either use a crossover cable or a straight-through cable. If you use a straight-through cable, the interface automatically detects the cable and swaps one of the transmit/receive pairs to MDIX.

The failover link is physically connected and ready for failover configuration.

Stateful failover links

A stateful failover link is a network connection that

- passes connection state information between failover units, and
- must be configured to enable stateful failover functionality.

State link configuration requirements

To use Stateful Failover, you must configure a Stateful Failover link (also known as the state link) to pass connection state information.

Shared failover links

A shared failover link is an interface conservation method that enables multiple network functions to use the same failover connection, though dedicated interfaces for state and failover links should be considered for large configurations and high traffic networks.

Dedicated interface

A dedicated interface is a data interface configuration that uses a dedicated physical or EtherChannel interface specifically for the state link in failover deployments.

Performance considerations

You can use a dedicated data interface—such as a physical or EtherChannel interface—for the state link. For requirements about a dedicated state link, refer to Interface for the failover link, on page 260. See the section on connecting the state link for additional information Connect the failover link, on page 261.

For optimum performance with long-distance failover, the state link latency should be less than 10 milliseconds but no more than 250 milliseconds. If latency exceeds 10 milliseconds, performance might degrade due to retransmission of failover messages.

Avoiding Interrupted Failover and Data Links

We recommend that failover links and data interfaces travel through different paths to decrease the chance that all interfaces fail at the same time. If the failover link is down, the ASA can use the data interfaces to determine if a failover is required. Subsequently, the failover operation is suspended until the health of the failover link is restored.

See the following connection scenarios to design a resilient failover network.

Scenario 1—Not Recommended

If a single switch or a set of switches are used to connect both failover and data interfaces between two ASAs, then when a switch or inter-switch-link is down, both ASAs become active. Therefore, the following two connection methods shown in the following figures are NOT recommended.

Figure 40: Connecting with a Single Switch—Not Recommended

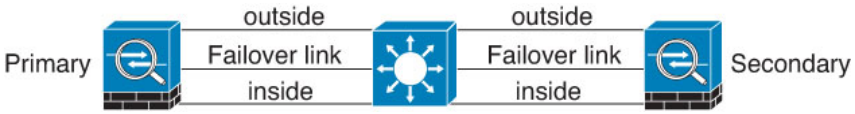
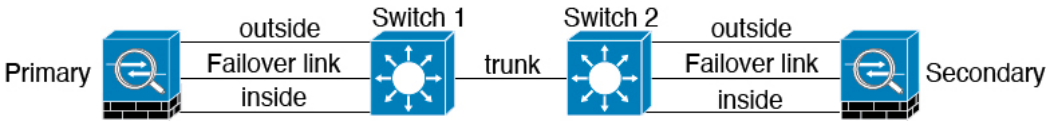


Figure 41: Connecting with a Double-Switch—Not Recommended



Scenario 2—Recommended

We recommend that failover links NOT use the same switch as the data interfaces. Instead, use a different switch or use a direct cable to connect the failover link, as shown in the following figures.

Figure 42: Connecting with a Different Switch

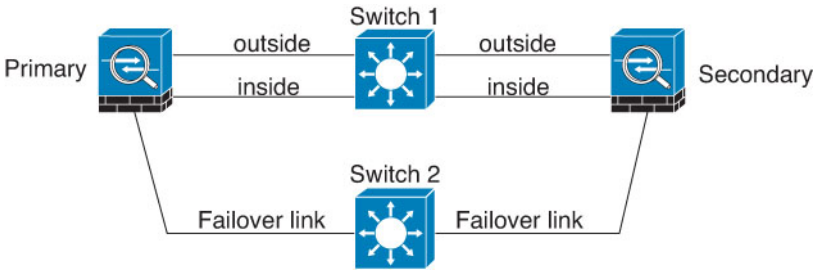
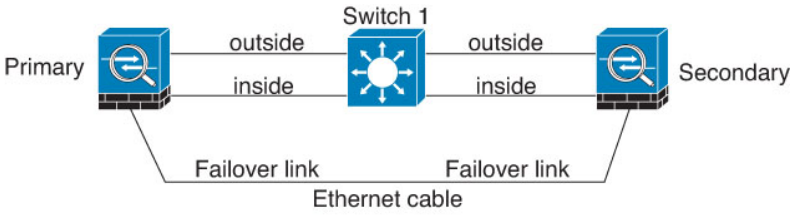


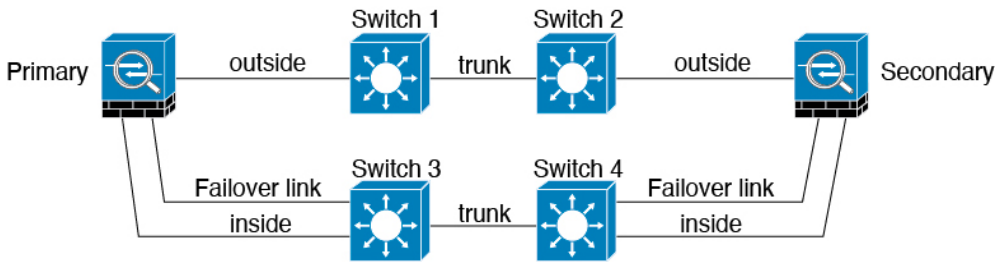
Figure 43: Connecting with a Cable



Scenario 3—Recommended

If the ASA data interfaces are connected to more than one set of switches, then a failover link can be connected to one of the switches, preferably the switch on the secure (inside) side of network, as shown in the following figure.

Figure 44: Connecting with a Secure Switch



MAC addresses and IP addresses in Failover

MAC addresses and IP addresses in Failover are network addressing mechanisms that

- provide unique identification for network interfaces during failover scenarios
- maintain network connectivity when primary devices become unavailable, and
- ensure seamless traffic flow between active and standby configurations.

Address configuration types

When you configure your interfaces, you can specify an active IP address and a standby IP address on the same network. Generally, when a failover occurs, the new active unit takes over the active IP addresses and MAC addresses. Because network devices see no change in the MAC to IP address pairing, no ARP entries change or time out anywhere on the network.

Note Although recommended, the standby address is not required. Without a standby IP address, the active unit cannot perform network tests to check the standby interface health; it can only track the link state. You also cannot connect to the standby unit on that interface for management purposes.

The IP address and MAC address for the state link do not change at failover.

Active/standby configurations use specific address handling methods:

For Active/Standby Failover, see the following for IP address and MAC address usage during a failover event:

1. The active unit always uses the primary unit's IP addresses and MAC addresses.
2. When the active unit fails over, the standby unit assumes the IP addresses and MAC addresses of the failed unit and begins passing traffic.
3. When the failed unit comes back online, it is now in a standby state and takes over the standby IP addresses and MAC addresses.

However, if the secondary unit boots without detecting the primary unit, then the secondary unit becomes the active unit and uses its own MAC addresses, because it does not know the primary unit MAC addresses. When the primary unit becomes available, the secondary (active) unit changes the MAC addresses to those of the primary unit, which can cause an interruption in your network traffic. Similarly, if you swap out the primary unit with new hardware, a new MAC address is used.

If you disable failover and set the failover configurations to a disabled state, you will need to manually resume failover, or reboot the device. It is recommended to use the command **failover reset** and resume the failover instead of rebooting the device. If you reload the standby unit with the failover configuration disabled, the standby unit boots up as the active unit and uses the primary unit's IP addresses and MAC addresses. This leads to duplicate IP addresses and causes network traffic disruptions. Use the command **failover reset** to enable failover and restore the traffic flow.



Note If you enable failover on a standalone device, the data interfaces go down at negotiation state of failover, interrupting traffic.

Virtual MAC addresses guard against this disruption, because the active MAC addresses are known to the secondary unit at startup, and remain the same in the case of new primary unit hardware. We recommend that you configure the virtual MAC address on both the primary and secondary units to ensure that the secondary unit uses the correct MAC addresses when it is the active unit, even if it comes online before the primary unit. If you do not configure virtual MAC addresses, you might need to clear the ARP tables on connected routers to restore traffic flow. The ASA does not send gratuitous ARPs for static NAT addresses when the MAC address changes, so connected routers do not learn of the MAC address change for these addresses.

For Active/Active failover, see the following for IP address and MAC address usage during a failover event:

1. The primary unit autogenerates active and standby MAC addresses for all interfaces in failover group 1 and 2 contexts. You can also manually configure the MAC addresses if necessary, for example, if there are MAC address conflicts.
2. Each unit uses the active IP addresses and MAC addresses for its active failover group, and the standby addresses for its standby failover group. For example, the primary unit is active for failover group 1, so it uses the active addresses for contexts in failover group 1. It is standby for the contexts in failover group 2, where it uses the standby addresses.
3. When a unit fails over, the other unit assumes the active IP addresses and MAC addresses of the failed failover group and begins passing traffic.
4. When the failed unit comes back online, and you enabled the preempt option, it resumes the failover group.

Active/active configurations implement additional address management:

Virtual MAC addresses provide specialized functionality:

The ASA has multiple methods to configure virtual MAC addresses. We recommend using only one method. If you set the MAC address using multiple methods, the MAC address used depends on many variables, and might not be predictable. Manual methods include the interface mode **mac-address** command, the **failover mac address** command, and for Active/Active failover, the failover group mode **mac address** command, in addition to autogeneration methods described below.

In multiple context mode, you can configure the ASA to generate virtual active and standby MAC addresses automatically for shared interfaces, and these assignments are synced to the secondary unit (see the **mac-address auto** command). For non-shared interfaces, you can manually set the MAC addresses for Active/Standby mode (Active/Active mode autogenerates MAC addresses for all interfaces).

For Active/Active failover, virtual MAC addresses are always used, either with default values or with values you can set per interface.

MAC address table updates occur during failover events:

During failover, the device that is designated as the new active device generates multicast packets for each MAC address entry in the MAC table and sends them to all the bridge group interfaces. This action prompts the upstream switches in the bridge group to update their routing tables with the new active device's interface to ensure accurate traffic forwarding.

The time taken to generate multicast packets and update the routing tables of the upstream switches depends on the number of entries in the MAC address table and the number of bridge group interfaces. Use the **show failover statistics state-switch-delay** command to display statistics related to the delays encountered during failover events.

Stateless and Stateful Failover

The ASA supports two types of failover, stateless and stateful for both the Active/Standby and Active/Active modes.



Note Some configuration elements for clientless SSL VPN (such as bookmarks and customization) use the VPN failover subsystem, which is part of Stateful Failover. You must use Stateful Failover to synchronize these elements between the members of the failover pair. Stateless failover is not recommended for clientless SSL VPN.

Stateless Failover

When a failover occurs, all active connections are dropped. Clients need to reestablish connections when the new active unit takes over.



Note Some configuration elements for clientless SSL VPN (such as bookmarks and customization) use the VPN failover subsystem, which is part of Stateful Failover. You must use Stateful Failover to synchronize these elements between the members of the failover pair. Stateless (regular) failover is not recommended for clientless SSL VPN.

Stateful failover

A stateful failover is a high availability mechanism that

- continuously passes per-connection state information from the active unit to the standby unit
- maintains connection information after a failover occurs, and
- allows supported end-user applications to continue communication sessions without reconnecting.

Stateful failover operation

When Stateful Failover is enabled, the active unit continually passes per-connection state information to the standby unit, or in Active/Active failover, between the active and standby failover groups. After a failover occurs, the same connection information is available at the new active unit. Supported end-user applications are not required to reconnect to keep the same communication session.

Supported features

Supported features are high availability capabilities that

- enable state information to be passed to the standby ASA device during Stateful Failover
- maintain connection states and critical information for seamless traffic flow, and
- ensure minimal disruption during failover events.

State information types

For Stateful Failover, these state information types are passed to the standby device:

- NAT translation table.
- TCP and UDP connections and states. Other types of IP protocols, and ICMP, are not parsed by the active unit, because they get established on the new active unit when a new packet arrives.
- The HTTP connection table (unless you enable HTTP replication).
- The HTTP connection states (if HTTP replication is enabled)—By default, the ASA does not replicate HTTP session information when Stateful Failover is enabled. We suggest that you enable HTTP replication.
- SCTP connection states are included. SCTP inspection stateful failover is best effort. If any SACK packets are lost during failover, the new active unit drops all other out-of-order packets in the queue until the missing packet is received.
- The ARP table
- The Layer 2 bridge table (for bridge groups)
- The ISAKMP and IPsec SA table
- GTP PDP connection database
- SIP signaling sessions and pin holes.
- ICMP connection state—ICMP connection replication is enabled only if the respective interface is assigned to an asymmetric routing group.
- Static and dynamic routing tables—Stateful Failover participates in dynamic routing protocols, like OSPF and EIGRP, so routes that are learned through dynamic routing protocols on the active unit are maintained in a Routing Information Base (RIB) table on the standby unit. Upon a failover event, packets travel normally with minimal disruption to traffic because the active secondary unit initially has rules that mirror the primary unit. Immediately after failover, the re-convergence timer starts on the newly active unit. Then the epoch number for the RIB table increments. During re-convergence, OSPF and EIGRP routes become updated with a new epoch number. Once the timer is expired, stale route entries (determined by the epoch number) are removed from the table. The RIB then contains the newest routing protocol forwarding information on the newly active unit.



Note Routes are synchronized only for link-up or link-down events on an active unit. If the link goes up or down on the standby unit, dynamic routes sent from the active unit may be lost. This is normal, expected behavior.

- DHCP Server—DHCP address leases are not replicated. However, a DHCP server configured on an interface will send a ping to make sure an address is not being used before granting the address to a DHCP client, so there is no impact to the service. State information is not relevant for DHCP relay or DDNS.
- Cisco IP SoftPhone sessions—If a failover occurs during an active Cisco IP SoftPhone session, the call remains active because the call session state information is replicated to the standby unit. When the call is terminated, the IP SoftPhone client loses connection with the Cisco Call Manager. This connection loss occurs because there is no session information for the CTIQBE hangup message on the standby unit. When the IP SoftPhone client does not receive a response back from the Call Manager within a certain time period, it considers the Call Manager unreachable and unregisters itself.
- RA VPN—Remote access VPN end users do not have to reauthenticate or reconnect the VPN session after a failover. However, applications operating over the VPN connection could lose packets during the failover process and not recover from the packet loss.
- From all the connections, only established ones will be replicated on the Standby device.

Unsupported features

Unsupported features are device capabilities that

- are not passed to the standby ASA in Stateful Failover
- do not maintain state information during failover events, and
- may require re-establishment after a failover occurs.

State information limitations

For Stateful Failover, the following state information is not passed to the standby ASA:

- The user authentication (uauth) table
- Multicast routing.

Bridge Group Requirements for Failover

There are special considerations for failover when using bridge groups.

Bridge Group Requirements for Appliances, ASAv

When the active unit fails over to the standby unit, the connected switch port running Spanning Tree Protocol (STP) can go into a blocking state for 30 to 50 seconds when it senses the topology change. To avoid traffic loss while the port is in a blocking state, you can configure one of the following workarounds depending on the switch port mode:

- Access mode—Enable the STP PortFast feature on the switch:

```
interface interface_id
  spanning-tree portfast
```

The PortFast feature immediately transitions the port into STP forwarding mode upon linkup. The port still participates in STP. So if the port is to be a part of the loop, the port eventually transitions into STP blocking mode.

- Trunk mode—Block BPDUs on the ASA on a bridge group's member interfaces with an EtherType access rule.

```
access-list id ethertype deny bpdu
access-group id in interface name1
access-group id in interface name2
```

Blocking BPDUs disables STP on the switch. Be sure not to have any loops involving the ASA in your network layout.

If neither of the above options are possible, then you can use one of the following less desirable workarounds that impacts failover functionality or STP stability:

- Disable interface monitoring.
- Increase interface holdtime to a high value that will allow STP to converge before the ASAs fail over.
- Decrease STP timers to allow STP to converge faster than the interface holdtime.

Failover health monitoring

Failover health monitoring is a system capability that monitors each unit for overall health and interface health.

Health monitoring tests

The ASA performs tests to determine the state of each unit. This section includes information about how the ASA performs tests to determine the state of each unit.

Unit health monitoring

Unit health monitoring is a failover mechanism that

- determines the health of peer units by monitoring the failover link with hello messages
- sends LANTEST messages on each data interface when three consecutive hello messages are missed, and
- initiates appropriate failover actions based on peer unit responsiveness.

Unit health monitoring behavior

The ASA determines the health of the other unit by monitoring the failover link with hello messages. If a unit does not receive three consecutive hello messages on the failover link, the sends LANTEST messages on each data interface, including the failover link, to validate whether the peer is responsive. For the Firepower 9300 and 4100 series, you can enable Bidirectional Forwarding Detection (BFD) monitoring, which is more reliable than hello messages. The action that the ASA takes depends on the response from the other unit. These are the possible actions:

- If the ASA receives a response on the failover link, then it does not fail over.

- If the ASA does not receive a response on the failover link, but it does receive a response on a data interface, then the unit does not failover. The failover link is marked as failed. You should restore the failover link as soon as possible because the unit cannot fail over to the standby while the failover link is down.
- If the ASA does not receive a response on any interface, then the standby unit switches to active mode and classifies the other unit as failed.

Interface monitoring

Interface monitoring is a failover capability that

- tracks the health of up to 1025 interfaces to detect interface failures
- performs interface tests when hello messages are not received within 15 seconds, and
- triggers failover when the number of failed interfaces meets the defined threshold.

Interface monitoring operation

You can monitor up to 1025 interfaces (in multiple context mode, divided between all contexts). You should monitor important interfaces. For example in multiple context mode, you might configure one context to monitor a shared interface: because the interface is shared, all contexts benefit from the monitoring.

When a unit does not receive hello messages on a monitored interface for 15 seconds (the default), it runs interface tests. (To change the period, see the **failover polltime interface** command, or for Active/Active failover, the **polltime interface** command) If one of the interface tests fails for an interface, but this same interface on the other unit continues to successfully pass traffic, then the interface is considered to be failed, and the ASA stops running tests.

If the threshold you define for the number of failed interfaces is met (see the **failover interface-policy** command, or for Active/Active failover, the **interface-policy** command), and the active unit has more failed interfaces than the standby unit, then a failover occurs. If an interface fails on both units, then both interfaces go into the "Unknown" state and do not count towards the failover limit defined by failover interface policy.

An interface becomes operational again if it receives any traffic. A failed ASA returns to standby mode if the interface failure threshold is no longer met.

If an interface has IPv4 and IPv6 addresses configured on it, the ASA uses the IPv4 addresses to perform the health monitoring. If an interface has only IPv6 addresses configured on it, then the ASA uses IPv6 neighbor discovery instead of ARP to perform the health monitoring tests. For the broadcast ping test, the ASA uses the IPv6 all nodes address (FE02::1).



Note If a failed unit does not recover and you believe it should not be failed, you can reset the state by entering the **failover reset** command. If the failover condition persists, however, the unit will fail again

Interface tests

Interface tests are monitoring mechanisms that:

- evaluate interface operational status during failover scenarios

- run sequentially with approximately 1.5 second duration per test by default, or 1/16 of the failover interface holdtime(see the **failover polltime interface** command, or for Active/Active failover, the **interface-policy** command), and
- determine interface failure status on ASA units.

Interface test sequence

The ASA uses these interface tests in sequence:

1. Link Up/Down test—A test of the interface status. If the Link Up/Down test indicates that the interface is down, then the ASA considers it failed, and testing stops. If the status is Up, then the ASA performs the Network Activity test.
2. Network Activity test—A received network activity test. At the start of the test, each unit clears its received packet count for its interfaces. As soon as a unit receives any eligible packets during the test, then the interface is considered operational. If both units receive traffic, then testing stops. If one unit receives traffic and the other unit does not, then the interface on the unit that does not receive traffic is considered failed, and testing stops. If neither unit receives traffic, then the ASA starts the ARP test.
3. ARP test—A test for successful ARP replies. Each unit sends a single ARP request for the IP address in the most recent entry in its ARP table. If the unit receives an ARP reply or other network traffic during the test, then the interface is considered operational. If the unit does not receive an ARP reply, then the ASA sends a single ARP request for the IP address in the *next* entry in the ARP table. If the unit receives an ARP reply or other network traffic during the test, then the interface is considered operational. If both units receive traffic, then testing stops. If one unit receives traffic, and the other unit does not, then the interface on the unit that does not receive traffic is considered failed, and testing stops. If neither unit receives traffic, then the ASA starts the Broadcast Ping test.
4. Broadcast Ping test—A test for successful ping replies. Each unit sends a broadcast ping, and then counts all received packets. If the unit receives any packets during the test, then the interface is considered operational. If both units receive traffic, then testing stops. If one unit receives traffic, and the other unit does not, then the interface on the unit that does not receive traffic is considered failed, and testing stops. If neither unit receives traffic, then testing starts over again with the ARP test. If both units continue to receive no traffic from the ARP and Broadcast Ping tests, then these tests will continue running in perpetuity.

Interface status

Interface status is a monitoring indicator that

- shows the current operational state of monitored interfaces in a failover configuration
- tracks traffic reception and hello packet communication between peer units, and
- determines whether an interface is actively monitored by the failover process.

Interface status types

Monitored interfaces can have these status types:

- **Unknown**: Initial status. This status can also mean the status cannot be determined.
- **Normal**: The interface is receiving traffic.

- **Testing**: Hello messages are not heard on the interface for five poll times.
- **Link Down**: The interface or VLAN is administratively down.
- **No Link**: The physical link for the interface is down.
- **Failed**: No traffic is received on the interface, yet traffic is heard on the peer interface.

Failover times

Failover times are high availability mechanisms that:

- initiate automatic switchover when specific failure conditions occur on the active unit,
- monitor system health through configurable detection thresholds and timing parameters, and
- ensure service continuity by transferring control to the standby unit when failures exceed defined limits.

Failover triggering events and timing parameters

These events trigger failover in a Firepower high availability pair:

- More than 50% of the Snort instances on the active unit are down.
- Disk space on the active unit is more than 90% full.
- The **no failover active** command is run on the active unit or the **failover active** command is run on the standby unit.
- The active unit has more failed interfaces than the standby unit.
- Interface failure on the active device exceeds the threshold configured.

  By default, failure of a single interface causes failover. You can change the default value by configuring a threshold for the number of interfaces or a percentage of monitored interfaces that must fail for the failover to occur. If the threshold is exceeded on the active device, failover occurs. If the threshold is exceeded on the standby device, the unit moves to **Fail** state.

  To change the default failover criteria, enter this command in global configuration mode:

Table 13: Interface policy command

Command	Purpose
failover interface-policy num [%] `hostname (config)# failover interface-policy 20%`	Changes the default failover criteria. When specifying a specific number of interfaces, the *num* argument can be from 1 to 250. When specifying a percentage of interfaces, the *num* argument can be from 1 to 100.

Note If you manually fail over using the CLI or ASDM, or you reload the ASA, the failover starts immediately and is not subject to the timers listed below.

Table 14: ASA

Failover condition	Minimum	Default	Maximum
Active unit loses power, hardware goes down, or the software reloads or crashes. When any of these occur, the monitored interfaces or failover link do not receives any hello message.	800 milliseconds	15 seconds	45 seconds
Active unit main board interface link down.	500 milliseconds	5 seconds	15 seconds
Active unit 4GE module interface link down.	2 seconds	5 seconds	15 seconds
Active unit interface up, but connection problem causes interface testing.	5 seconds	25 seconds	75 seconds

Configuration Synchronization

Failover includes various types of configuration synchronization.

Running Configuration Replication

Running configuration replication occurs when any one or both of the devices in the failover pair boot.

In Active/Standby failover, configurations are always synchronized from the active unit to the standby unit.

In Active/Active failover, whichever unit boots second obtains the running configuration from the unit that boots first, regardless of the primary or secondary designation of the booting unit. After both units are up, commands entered in the system execution space are replicated from the unit on which failover group 1 is in the active state.

When the standby/second unit completes its initial startup, it clears its running configuration (except for the **failover** commands needed to communicate with the active unit), and the active unit sends its entire configuration to the standby/second unit. When the replication starts, the ASA console on the active unit displays the message “Beginning configuration replication: Sending to mate,” and when it is complete, the ASA displays the message “End Configuration Replication to mate.” Depending on the size of the configuration, replication can take from a few seconds to several minutes.

On the unit receiving the configuration, the configuration exists only in running memory. You should save the configuration to flash memory according to Save Configuration Changes, on page 42. For example, in Active/Active failover, enter the **write memory all** command in the system execution space on the unit that has failover group 1 in the active state. The command is replicated to the peer unit, which proceeds to write its configuration to flash memory.

Note During replication, commands entered on the unit sending the configuration may not replicate properly to the peer unit, and commands entered on the unit receiving the configuration may be overwritten by the configuration being received. Avoid entering commands on either unit in the failover pair during the configuration replication process.

File Replication

Configuration syncing does not replicate the following files and configuration components, so you must copy these files manually so they match:

- AnyConnect Client images
- CSD images
- AnyConnect Client profiles

  The ASA uses a cached file for the AnyConnect Client profile stored in cache:/stc/profiles, and not the file stored in the flash file system. To replicate the AnyConnect Client profile to the standby unit, perform one of the following:

  - Enter the **write standby** command on the active unit.
  - Reapply the profile on the active unit.
  - Reload the standby unit.

- Local Certificate Authorities (CAs)
- ASA images
- ASDM images

Command Replication

After startup, commands that you enter on the active unit are immediately replicated on the standby unit. You do not have to save the active configuration to flash memory to replicate the commands.

In Active/Active failover, commands entered in the system execution space are replicated from the unit on which failover group 1 is in the active state.

Failure to enter the commands on the appropriate unit for command replication to occur causes the configurations to be out of synchronization. Those changes may be lost the next time the initial configuration synchronization occurs.

The following commands are replicated to the standby ASA:

- All configuration commands except for **mode**, **firewall**, and **failover lan unit**
- **copy running-config startup-config**
- **delete**
- **mkdir**
- **rename**
- **rmdir**
- **write memory**

The following commands are *not* replicated to the standby ASA:

- All forms of the **copy** command except for **copy running-config startup-config**
- All forms of the **write** command except for **write memory**

- **debug**
- **failover lan unit**
- **firewall**
- **show**
- **terminal pager** and **pager**

Active/standby failover

Active/standby failover is a high availability configuration that

- lets you use a standby ASA device to take over the functionality of a failed unit,
- enables the standby unit to become the active unit when the active unit fails, and
- allows the failed unit to come back online as the standby unit if the problem is temporary.

The failed unit must be replaced if the problem is not temporary. Set the standby unit to primary before replacing the failed unit to retain the configuration of the secondary unit.

For multiple context mode, the ASA can fail over the entire unit (including all contexts) but cannot fail over individual contexts separately.

Primary/secondary roles and active/standby status

Primary/secondary roles and active/standby status are failover configuration concepts that

- define unit hierarchy, with primary units taking precedence during simultaneous startup
- determine IP address and MAC address usage patterns during failover operations, and
- establish traffic flow responsibilities between paired units in Active/Standby failover configurations.

Primary and secondary unit differences

The main differences between the two units in a failover pair are related to which unit is active and which unit is standby, namely which IP addresses to use and which unit actively passes traffic.

However, a few differences exist between the units depending on which unit is primary (as specified in the configuration) and which unit is secondary:

- The primary unit always becomes the active unit if both units start up at the same time (and are of equal operational health).
- The primary unit MAC addresses are always coupled with the active IP addresses. The exception to this rule occurs when the secondary unit becomes active and cannot obtain the primary unit MAC addresses over the failover link. In this case, the secondary unit MAC addresses are used.

Active unit determination at startup

Active unit determination at startup is a high availability process that

- sets a unit to standby if it boots and detects a peer already running as active,

- sets a unit to active if it boots and does not detect a peer, and
- sets the primary unit as active and secondary unit as standby if both units boot simultaneously.

Failover events

A failover event is a system condition that

- triggers an active unit to transfer control to a standby unit in Active/Standby failover configurations
- occurs on a unit basis and cannot fail over individual or groups of contexts even on systems running in multiple context mode, and
- follows specific failover policies that determine whether failover occurs and what actions each unit takes.

Failover event policies and actions

This table shows the failover action for each failure event. For each failure event, the table shows the failover policy (failover or no failover), the action taken by the active unit, the action taken by the standby unit, and any special notes about the failover condition and actions.

Table 15: Failover events

Failure Event	Policy	Active Unit Action	Standby Unit Action	Notes
Active unit failed (power or hardware)	Failover	n/a	Become active Mark active as failed	No hello messages are received on any monitored interface or the failover link.
Formerly active unit recovers	No failover	Become standby	No action	None.
Standby unit failed (power or hardware)	No failover	Mark standby as failed	n/a	When the standby unit is marked as failed, then the active unit does not attempt to fail over, even if the interface failure threshold is surpassed.
Failover link failed during operation	No failover	Mark failover link as failed	Mark failover link as failed	You should restore the failover link as soon as possible because the unit cannot fail over to the standby unit while the failover link is down.
Failover link failed at startup	No failover	Become active Mark failover link as failed	Become active Mark failover link as failed	If the failover link is down at startup, both units become active.
State link failed	No failover	No action	No action	State information becomes out of date, and sessions are terminated if a failover occurs.
Interface failure on active unit above threshold	Failover	Mark active as failed	Become active	None.

Failure Event	Policy	Active Unit Action	Standby Unit Action	Notes
Interface failure on standby unit above threshold	No failover	No action	Mark standby as failed	When the standby unit is marked as failed, then the active unit does not attempt to fail over even if the interface failure threshold is surpassed.

About Active/Active Failover

This section describes Active/Active failover.

Active/Active Failover Overview

In an Active/Active failover configuration, both ASAs can pass network traffic. Active/Active failover is only available to ASAs in multiple context mode. In Active/Active failover, you divide the security contexts on the ASA into a maximum of 2 failover groups.

A failover group is simply a logical group of one or more security contexts. You can assign failover group to be active on the primary ASA, and failover group 2 to be active on the secondary ASA. When a failover occurs, it occurs at the failover group level. For example, depending on interface failure patterns, it is possible for failover group 1 to fail over to the secondary ASA, and subsequently failover group 2 to fail over to the primary ASA. This event could occur if the interfaces in failover group 1 are down on the primary ASA but up on the secondary ASA, while the interfaces in failover group 2 are down on the secondary ASA but up on the primary ASA.

The admin context is always a member of failover group 1. Any unassigned security contexts are also members of failover group 1 by default. If you want Active/Active failover, but are otherwise uninterested in multiple contexts, the simplest configuration would be to add one additional context and assign it to failover group 2.

Note When configuring Active/Active failover, make sure that the combined traffic for both units is within the capacity of each unit.

Note You can assign both failover groups to one ASA if desired, but then you are not taking advantage of having two active ASAs.

Primary/Secondary Roles and Active/Standby Status for a Failover Group

As in Active/Standby failover, one unit in an Active/Active failover pair is designated the primary unit, and the other unit the secondary unit. Unlike Active/Standby failover, this designation does not indicate which unit becomes active when both units start simultaneously. Instead, the primary/secondary designation does two things:

- The primary unit provides the running configuration to the pair when they boot simultaneously.

- Each failover group in the configuration is configured with a primary or secondary unit preference. When used with preemption, this preference ensures that the failover group runs on the correct unit after it starts up. Without preemption, both groups run on the first unit to boot up.

Active Unit Determination for Failover Groups at Startup

The unit on which a failover group becomes active is determined as follows:

- When a unit boots while the peer unit is not available, both failover groups become active on the unit.
- When a unit boots while the peer unit is active (with both failover groups in the active state), the failover groups remain in the active state on the active unit regardless of the primary or secondary preference of the failover group until one of the following occurs:
  - A failover occurs.
  - A failover is manually forced.
  - A preemption for the failover group is configured, which causes the failover group to automatically become active on the preferred unit when the unit becomes available.

Failover Events

In an Active/Active failover configuration, failover occurs on a failover group basis, not a system basis. For example, if you designate both failover groups as Active on the primary unit, and failover group 1 fails, then failover group 2 remains Active on the primary unit while failover group 1 becomes active on the secondary unit.

Because a failover group can contain multiple contexts, and each context can contain multiple interfaces, it is possible for all interfaces in a single context to fail without causing the associated failover group to fail.

The following table shows the failover action for each failure event. For each failure event, the policy (whether or not failover occurs), actions for the active failover group, and actions for the standby failover group are given.

Table 16: Failover Events

Failure Event	Policy	Active Group Action	Standby Group Action	Notes
A unit experiences a power or software failure	Failover	Become standby Mark as failed	Become active Mark active as failed	When a unit in a failover pair fails, any active failover groups on that unit are marked as failed and become active on the peer unit.
Interface failure on active failover group above threshold	Failover	Mark active group as failed	Become active	None.
Interface failure on standby failover group above threshold	No failover	No action	Mark standby group as failed	When the standby failover group is marked as failed, the active failover group does not attempt to fail over, even if the interface failure threshold is surpassed.

Failure Event	Policy	Active Group Action	Standby Group Action	Notes
Formerly active failover group recovers	No failover	No action	No action	Unless failover group preemption is configured, the failover groups remain active on their current unit.
Failover link failed at startup	No failover	Become active	Become active	If the failover link is down at startup, both failover groups on both units become active.
State link failed	No failover	No action	No action	State information becomes out of date, and sessions are terminated if a failover occurs.
Failover link failed during operation	No failover	n/a	n/a	Each unit marks the failover link as failed. You should restore the failover link as soon as possible because the unit cannot fail over to the standby unit while the failover link is down.

Licensing for Failover

For most models, failover units do not require the same license on each unit. If you have licenses on both units, they combine into a single running failover cluster license. There are some exceptions to this rule. See the following table for precise licensing requirements for failover.

Model	License Requirement
ASAv	See Failover Licenses for the ASAv, on page 98.
Firepower 1010	Security Plus license on both units. See Failover Licenses for the Firepower 1010, on page 98.
Firepower 1100	See Failover Licenses for the Firepower 1100, on page 99.
Firepower 2100	See Failover Licenses for the Firepower 2100, on page 101.
Secure Firewall 3100	See Failover Licenses for the Secure Firewall 3100, on page 102.
Firepower 4100/9300	See Failover Licenses for the Firepower 4100/9300, on page 103.
ISA 3000	Security Plus license on both units. **Note** Each unit must have the same encryption license.

Note A valid permanent key is required; in rare instances on the ISA 3000, your PAK authentication key can be removed. If your key consists of all 0's, then you need to reinstall a valid authentication key before failover can be enabled.

Guidelines for Failover

Context Mode

- Active/Active mode is supported only in multiple context mode.
- For multiple context mode, perform all steps in the system execution space unless otherwise noted.

Model Support

- / 1010/ :
  - You should not use the switch port functionality when using Failover. Because the switch ports operate in hardware, they continue to pass traffic on both the active *and* the standby units. Failover is designed to prevent traffic from passing through the standby unit, but this feature does not extend to switch ports. In a normal Failover network setup, active switch ports on both units will lead to network loops. We suggest that you use external switches for any switching capability. Note that VLAN interfaces can be monitored by failover, while switch ports cannot. Theoretically, you can put a single switch port on a VLAN and successfully use Failover, but a simpler setup is to use physical firewall interfaces instead.
  - You can only use a firewall interface as the failover link.
- Firepower 9300—We recommend that you use inter-chassis Failover for the best redundancy.
- The ASAv on public cloud networks such as Microsoft Azure and Amazon Web Services are not supported with regular Failover because Layer 2 connectivity is required. Instead, see Failover for High Availability in the Public Cloud, on page 313.

ASAv Failover for High Availability

When creating a failover pair with the ASAv, it is necessary to add the data interfaces to each ASAv in the same order. If the exact same interfaces are added to each ASAv, but in different order, errors may be presented at the ASAv Console. Failover functionality may also be affected

Additional Guidelines

- When the active unit fails over to the standby unit, the connected switch port running Spanning Tree Protocol (STP) can go into a blocking state for 30 to 50 seconds when it senses the topology change. To avoid traffic loss while the port is in a blocking state, you can enable the STP PortFast feature on the switch:

  interface *interface_id* **spanning-tree portfast**

This workaround applies to switches connected to both routed mode and bridge group interfaces. The PortFast feature immediately transitions the port into STP forwarding mode upon linkup. The port still participates in STP. So if the port is to be a part of the loop, the port eventually transitions into STP blocking mode.

- Configuring port security on the switches connected to the ASA failover pair can cause communication problems when a failover event occurs. This problem occurs when a secure MAC address configured or learned on one secure port moves to another secure port, a violation is flagged by the switch port security feature.
- You can monitor up to 1025 interfaces on a unit, across all contexts.
- For Active/Standby Failover and a VPN IPsec tunnel, you cannot monitor both the active and standby units using SNMP over the VPN tunnel. The standby unit does not have an active VPN tunnel, and will drop traffic destined for the NMS. You can instead use SNMPv3 with encryption so the IPsec tunnel is not required.
- When no failover is triggered during a healthy active/standby state, none of the interfaces on the standby unit are shut down. However, if this occurs while HA is in an active/failed state, all interfaces—including the management interface—are shut down on the failed unit, except for the failover LAN and state interfaces. This is done to prevent potential traffic issues if the failed interfaces become active after failover has been disabled. The failover LAN and state interfaces remain up to ensure ongoing failover communication.
- For Active/Active failover, no two interfaces in the same context should be configured in the same ASR group.
- For Active/Active failover, you can define a maximum of two failover groups.
- For Active/Active failover, when removing failover groups, you must remove failover group 1 last. Failover group1 always contains the admin context. Any context not assigned to a failover group defaults to failover group 1. You cannot remove a failover group that has contexts explicitly assigned to it.
- Immediately after failover, the source address of syslog messages will be the failover interface address for a few seconds.
- For better convergence (during a failover), you must shut down the interfaces on a HA pair that are not associated with any configuration or instance.
- In high availability deployments with a shared failover and state interfaces, if the connectivity over failover interface fails while the standby unit is in sync config state, and eventually when the standby unit switches to active state, it results in a split brain. This is observed even when there is a monitored interface.

  The connectivity over failover interface can fail in the following situations:

  - After the standby unit reboots, the connectivity over failover interface fails while the standby unit is in sync config state.
  - After enabling previously disabled failover on the standby unit, the connectivity over failover interface fails while the standby unit is in sync config state.

  To avoid split brain, use failover and state interfaces. If the connectivity failures over the failover interfaces persist, isolate the new active (previously standby) unit from the network.
- If you configure failover encryption in evaluation mode, the systems use DES for the encryption. If you then register the devices using an export-compliant account, the devices will use AES after a reboot.

Thus, if a system reboots for any reason, including after installing an upgrade, the peers will be unable to communicate and both units will become the active unit. We recommend that you do not configure encryption until after you register the devices. If you do configure this in evaluation mode, we recommend you remove the encryption before registering the devices.

- When using SNMPv3 with failover, if you replace a failover unit, then SNMPv3 users are not replicated to the new unit. You must re-add the SNMPv3 users to the active unit to force the users to replicate to the new unit; or you can add the users directly on the new unit. Reconfigure each user by entering the **snmp-server user** *username group-name* **v3** command on the active unit or directly to the standby unit with the *priv-password* option and *auth-password* option in their unencrypted forms.
- The device does not share SNMP client engine data with its peer.
- If you have a very large number of access control and NAT rules, the size of the configuration can prevent efficient configuration replication, resulting in the standby unit taking an excessively long time to reach standby ready state. This can also impact your ability to connect to the standby unit during replication through the console or SSH session. To enhance configuration replication performance, enable transactional commit for both access rules and NAT, using the **asp rule-engine transactional-commit access-group** and **asp rule-engine transactional-commit nat** commands.
- A unit in a Failover pair transitioning to the standby role synchronizes its clock with the active unit.

  Example:

```
firepower#show clock
01:00:52 UTC Mar 1 2022

...
01:01:18 UTC Mar 1 2022 <======= Incorrect (previous) clock
Cold Standby               Sync Config               Detected an Active mate

19:38:21 UTC Apr 9 2022 <======= Updated clock
Sync Config                Sync File System          Detected an Active mate
...
firepower/sec/stby#show clock
19:38:40 UTC Apr 9 2022
```

- The units in Failover do not dynamically synchronize the clock. Here are some examples of events when synchronization takes place:
  - A new Failover pair is created.
  - Failover is broken and re-created.
  - Communication over the failover link was disrupted and reestablished.
  - Failover status was manually changed at the CLI .
- Enabling Failover forces all routes to be deleted and are re-added after the Failover progression changes to the Active state. You could experience connection loss during this phase.
- If you enable failover on a standalone device, the data interfaces go down at negotiation state of failover, interrupting traffic.
- In the Failover configuration, short-lived connections, usually using port 53, are closed quickly and never transferred or synchronized from Active to Standby, so there might be a difference in the number of connections on both Failover devices. This is expected behavior for short-lived connections. You can try to compare the connections that are long-lived (for example, more than 30-60 seconds).

- In the Failover configuration, embryonic connections—connection requests that have not yet completed the three-way handshake process—are closed quickly and not synchronized between the active and standby devices. This design ensures HA system efficiency and security. For this reason, there might be a difference in the number of connections on both Failover devices, which is to be expected.
- If the failover LAN link is not connected back-to-back and instead connected through one or more switches, a failure within the intermediate path can cause the active unit to lose connectivity with the standby unit, resulting in inconsistent active/standby states. Although this does not impact Failover functionality, it is recommended to check and recover the failover-link path between the active and standby units.

  When the failover LAN link is down, it is not recommended to deploy any configuration, as it may not be replicated to the peer unit.
- In OSPF in ASA, if a nearby switch is down and if the ASA interface is connected to the same switch, the interfaces in the firewall also go down with the switch failure. This is an expected behavior. This will trigger an High Availablity Failover, as designed.
- In OSPF, after a failover, the OSPF connection with the peer device becomes invalid. Terminate the invalid connection, and then establish a new OSPF connection.
- If the standby ASA is connected to a different switch, in this case when the interface comes up, the routing tables will be different than that of the active ASA. This will lead to an outage for a short duration (approximately 15-17 seconds) until the routes and adjacency is updated.
- In transparent mode, if you have problems with the active unit losing the MAC address of the hot-standby router (HSRP), create a static mapping for the MAC address.

Defaults for Failover

By default, the failover policy consists of the following:

- No HTTP replication in Stateful Failover.
- A single interface failure causes failover.
- The interface poll time is 5 seconds.
- The interface hold time is 25 seconds.
- The unit poll time is 1 second.
- The unit hold time is 15 seconds.
- Virtual MAC addresses are disabled in multiple context mode.
- Monitoring on all physical interfaces.

Configure Active/Standby Failover

To configure Active/Standby failover, configure basic failover settings on both the primary and secondary units. All other configuration occurs only on the primary unit, and is then synched to the secondary unit.

Configure the Primary Unit for Active/Standby Failover

Follow the steps in this section to configure the primary in an Active/Standby failover configuration. These steps provide the minimum configuration needed to enable failover on the primary unit.

Before you begin

- We recommend that you configure standby IP addresses for all interfaces except for the failover and state links. If you use a 31-bit subnet mask for point-to-point connections, do not configure a standby IP address. You will not be able to enable failover if any interfaces are configured for DHCP.
- Do not configure a **nameif** for the failover and state links.
- For multiple context mode, complete this procedure in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.

Procedure

Step 1 Designate this unit as the primary unit:

failover lan unit primary

Step 2 Specify the interface to be used as the failover link:

failover lan interface *if_name interface_id*

Example:

```
ciscoasa(config)# failover lan interface folink gigabitethernet0/3
```

This interface cannot be used for any other purpose (except, optionally, the state link).

The *if_name* argument assigns a name to the interface.

The *interface_id* argument can be a data physical interface, subinterface, or EtherChannel interface ID. On the Firepower 1010, the interface is a firewall interface ID; you cannot specify a switch port ID or VLAN ID. For the Firepower 4100/9300, you can use any data-type interface.

Step 3 Assign the active and standby IP addresses to the failover link:

failover interface ip *failover_if_name* {*ip_address mask* | *ipv6_address* / *prefix*} **standby** *ip_address*

Example:

```
ciscoasa(config)# failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
```

Or:

```
ciscoasa(config)# failover interface ip folink 2001:a0a:b00::a0a:b70/64 standby
2001:a0a:b00::a0a:b71
```

This address should be on an unused subnet. This subnet can be 31-bits (255.255.255.254) with only two IP addresses. 169.254.1.0/24 and fd00:0:0:*::/64 are internally used subnets, and you cannot use them for the failover or state links.

The standby IP address must be in the same subnet as the active IP address.

Step 4 Enable the failover link:

interface *failover_interface_id*

no shutdown

Example:

```
ciscoasa(config)# interface gigabitethernet 0/3
ciscoasa(config-if)# no shutdown
```

Step 5 (Optional) If you want to use a separate interface for the state link, specify the interface.

failover link *if_name interface_id*

Example:

```
ciscoasa(config)# failover link folink gigabitethernet0/4
```

If you do not specify a separate interface, then the failover link is used for the statelink.

The *if_name* argument assigns a name to the interface.

The *interface_id* argument can be a physical interface, subinterface, or EtherChannel interface ID. On the Firepower 1010, the interface is a firewall interface ID; you cannot specify a switch port ID or VLAN ID.

Step 6 If you specified a separate state link, assign the active and standby IP addresses to the state link:

failover interface ip *state_if_name* {*ip_address mask* | *ipv6_address*/*prefix*} **standby** *ip_address*

Example:

```
ciscoasa(config)# failover interface ip statelink 172.27.49.1 255.255.255.0 standby
172.27.49.2
```

Or:

```
ciscoasa(config)# failover interface ip statelink 2001:a0a:b00:a::a0a:b70/64 standby
2001:a0a:b00:a::a0a:b71
```

This address should be on an unused subnet, different from the failover link. This subnet can be 31-bits (255.255.255.254) with only two IP addresses. 169.254.1.0/24 and fd00:0:0:*::/64 are internally used subnets, and you cannot use them for the failover or state links.

The standby IP address must be in the same subnet as the active IP address.

Skip this step if you are sharing the state link.

Step 7 If you specified a separate state link, enable the state link.

interface *state_interface_id*

no shutdown

Example:

```
ciscoasa(config)# interface gigabitethernet 0/4
ciscoasa(config-if)# no shutdown
```

Skip this step if you are sharing the state link.

Step 8 (Optional) Do one of the following to encrypt communications on the failover and state links:

- (Preferred) Establish IPsec LAN-to-LAN tunnels on the failover and state links between the units to encrypt all failover communications:

  failover ipsec pre-shared-key [**0** | **8**] *key*

  Example:

  ```
  ciscoasa(config)# failover ipsec pre-shared-key a3rynsun
  ```

  The *key* can be up to 128 characters in length. Identify the same key on both units. The key is used by IKEv2 to establish the tunnels.

  If you use a master passphrase (see Configure the Master Passphrase, on page 744), then the key is encrypted in the configuration. If you are copying from the configuration (for example, from **more system:running-config** output), specify that the key is encrypted by using the **8** keyword. **0** is used by default, specifying an unencrypted password.

  The **failover ipsec pre-shared-key** shows as ***** in **show running-config** output; this obscured key is not copyable.

  If you do not configure failover and state link encryption, failover communication, including any passwords or keys in the configuration that are sent during command replication, will be in clear text.

  You cannot use both IPsec encryption and the legacy **failover key** encryption. If you configure both methods, IPsec is used. However, if you use the master passphrase, you must first remove the failover key using the **no failover key** command before you configure IPsec encryption.

  Failover LAN-to-LAN tunnels do not count against the IPsec (Other VPN) license.

- (Optional) Encrypt failover communication on the failover and state links:

  failover key [**0** | **8**] {**hex** *key* | *shared_secret*}

  Example:

  ```
  ciscoasa(config)# failover key johncr1cht0n
  ```

  Use a *shared_secret* from 1 to 63 characters or a 32-character **hex** *key*. For the *shared_secret*, you can use any combination of numbers, letters, or punctuation. The shared secret or hex key is used to generate the encryption key. Identify the same key on both units.

  If you use a master passphrase (see Configure the Master Passphrase, on page 744), then the shared secret or hex key is encrypted in the configuration. If you are copying from the configuration (for example, from **more system:running-config** output), specify that the shared secret or hex key is encrypted by using the **8** keyword. **0** is used by default, specifying an unencrypted password.

The **failover key** shared secret shows as ***** in **show running-config** output; this obscured key is not copyable.

If you do not configure failover and state link encryption, failover communication, including any passwords or keys in the configuration that are sent during command replication, will be in clear text.

Step 9 Enable failover:

failover

Step 10 Save the system configuration to flash memory:

write memory

Examples

The following example configures the failover parameters for the primary unit:

```
failover lan unit primary
failover lan interface folink gigabitethernet0/3

failover interface ip folink 172.27.48.0 255.255.255.254 standby 172.27.48.1
interface gigabitethernet 0/3
  no shutdown
failover link folink gigabitethernet0/3
failover ipsec pre-shared-key a3rynsun
failover
```

Configure the Secondary Unit for Active/Standby Failover

The only configuration required on the secondary unit is for the failover link. The secondary unit requires these commands to communicate initially with the primary unit. After the primary unit sends its configuration to the secondary unit, the only permanent difference between the two configurations is the **failover lan unit** command, which identifies each unit as primary or secondary.

Before you begin

- You will not be able to enable failover if any interfaces are configured for DHCP.
- Do not configure a **nameif** for the failover and state links.
- For multiple context mode, complete this procedure in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.

Procedure

Step 1 Re-enter the exact same commands as on the primary unit *except* for the **failover lan unit primary** command. You can optionally replace it with the **failover lan unit secondary** command, but it is not necessary because **secondary** is the default setting. See Configure the Primary Unit for Active/Standby Failover, on page 284.

For example:

```
ciscoasa(config)# failover lan interface folink gigabitethernet0/3
INFO: Non-failover interface config is cleared on GigabitEthernet0/3 and its sub-interfaces
ciscoasa(config)# failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
ciscoasa(config)# interface gigabitethernet 0/3
ciscoasa(config-ifc)# no shutdown
ciscoasa(config)# failover ipsec pre-shared-key a3rynsun
ciscoasa(config)# failover
```

Step 2 After the failover configuration syncs, save the configuration to flash memory:

```
ciscoasa(config)# write memory
```

Configure Active/Active Failover

This section tells how to configure Active/Active failover.

Configure the Primary Unit for Active/Active Failover

Follow the steps in this section to configure the primary unit in an Active/Active failover configuration. These steps provide the minimum configuration needed to enable failover on the primary unit.

Before you begin

- Enable multiple context mode according to Enable or Disable Multiple Context Mode, on page 224.
- We recommend that you configure standby IP addresses for all interfaces except for the failover and state links according to Routed and Transparent Mode Interfaces, on page 669. If you use a 31-bit subnet mask for point-to-point connections, do not configure a standby IP address. You will not be able to enable failover if any interfaces are configured for DHCP.
- Do not configure a **nameif** for the failover and state links.
- Complete this procedure in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.

Procedure

Step 1 Designate this unit as the primary unit:

failover lan unit primary

Step 2 Specify the interface to be used as the failover link:

failover lan interface *if_name interface_id*

Example:

```
ciscoasa(config)# failover lan interface folink gigabitethernet0/3
```

This interface cannot be used for any other purpose (except, optionally, the state link).

The *if_name* argument assigns a name to the interface.

The *interface_id* argument can be a physical interface, subinterface, or EtherChannel interface ID. For the Firepower 4100/9300, you can use any data-type interface.

Step 3 Assign the active and standby IP addresses to the failover link:

standby failover interface ip *if_name* {*ip_address mask* | *ipv6_address*/*prefix* } **standby** *ip_address*

Example:

```
ciscoasa(config)# failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
```

Or:

```
ciscoasa(config)# failover interface ip folink 2001:a0a:b00::a0a:b70/64 standby
2001:a0a:b00::a0a:b71
```

This address should be on an unused subnet. This subnet can be 31-bits (255.255.255.254) with only two IP addresses. 169.254.1.0/24 and fd00:0:0:*::/64 are internally used subnets, and you cannot use them for the failover or state links.

The standby IP address must be in the same subnet as the active IP address.

Step 4 Enable the failover link:

interface *failover_interface_id*

no shutdown

Example:

```
ciscoasa(config)# interface gigabitethernet 0/3
ciscoasa(config-if)# no shutdown
```

Step 5 (Optional) If you want to use a separate interface for the state link, specify the interface.

failover link *if_name interface_id*

Example:

```
ciscoasa(config)# failover link statelink gigabitethernet0/4
```

We recommend that you use a separate state link from the failover link. If you do not specify a separate interface, then the failover link is used for the statelink.

The *if_name* argument assigns a name to the interface.

The *interface_id* argument can be a physical interface, subinterface, or EtherChannel interface ID.

Step 6 If you specified a separate state link, assign the active and standby IP addresses to the state link:

This address should be on an unused subnet, different from the failover link. This subnet can be 31-bits (255.255.255.254) with only two IP addresses. 169.254.1.0/24 and fd00:0:0:*::/64 are internally used subnets, and you cannot use them for the failover or state links.

The standby IP address must be in the same subnet as the active IP address.

Skip this step if you are sharing the state link.

failover interface ip state *if_name* {*ip_address mask* | *ipv6_address*/*prefix*} **standby** *ip_address*

Example:

```
ciscoasa(config)# failover interface ip statelink 172.27.49.1 255.255.255.0 standby
172.27.49.2
```

Or:

```
ciscoasa(config)# failover interface ip statelink 2001:a0a:b00:a::a0a:b70/64 standby
2001:a0a:b00:a::a0a:b71
```

Step 7 If you specified a separate state link, enable the state link:

interface *state_interface_id*

no shutdown

Example:

```
ciscoasa(config)# interface gigabitethernet 0/4
ciscoasa(config-if)# no shutdown
```

Skip this step if you are sharing the state link.

Step 8 (Optional) Do one of the following to encrypt communications on the failover and state links:

- (Preferred) Establish IPsec LAN-to-LAN tunnels on the failover and state links between the units to encrypt all failover communications:

  failover ipsec pre-shared-key [**0** | **8**] *key*

  ```
  ciscoasa(config)# failover ipsec pre-shared-key a3rynsun
  ```

  The *key* can be up to 128 characters in length. Identify the same key on both units. The key is used by IKEv2 to establish the tunnels.

  If you use a master passphrase (see Configure the Master Passphrase, on page 744), then the key is encrypted in the configuration. If you are copying from the configuration (for example, from **more system:running-config** output), specify that the key is encrypted by using the **8** keyword. **0** is used by default, specifying an unencrypted password.

  The **failover ipsec pre-shared-key** shows as ***** in **show running-config** output; this obscured key is not copyable.

  If you do not configure failover and state link encryption, failover communication, including any passwords or keys in the configuration that are sent during command replication, will be in clear text.

You cannot use both IPsec encryption and the legacy **failover key** encryption. If you configure both methods, IPsec is used. However, if you use the master passphrase, you must first remove the failover key using the **no failover key** command before you configure IPsec encryption.

Failover LAN-to-LAN tunnels do not count against the IPsec (Other VPN) license.

- (Optional) Encrypt failover communication on the failover and state links:

  failover key [**0** | **8**] {**hex** *key* | *shared_secret*}

```
ciscoasa(config)# failover key johncr1cht0n
```

  Use a *shared_secret*, from 1 to 63 characters, or a 32-character **hex** *key*.

  For the *shared_secret*, you can use any combination of numbers, letters, or punctuation. The shared secret or hex key is used to generate the encryption key. Identify the same key on both units.

  If you use a master passphrase (see Configure the Master Passphrase, on page 744), then the shared secret or hex key is encrypted in the configuration. If you are copying from the configuration (for example, from **more system:running-config** output), specify that the shared secret or hex key is encrypted by using the **8** keyword. **0** is used by default, specifying an unencrypted password.

  The **failover key** shared secret shows as ***** in **show running-config** output; this obscured key is not copyable.

  If you do not configure failover and state link encryption, failover communication, including any passwords or keys in the configuration that are sent during command replication, will be in clear text.

Step 9 Create failover group 1:

failover group 1

primary

preempt [*delay*]

Example:

```
ciscoasa(config-fover-group)# failover group 1
ciscoasa(config-fover-group)# primary
ciscoasa(config-fover-group)# preempt 1200
```

Typically, you assign group 1 to the primary unit, and group 2 to the secondary unit. Both failover groups become active on the unit that boots first (even if it seems like they boot simultaneously, one unit becomes active first), despite the primary or secondary setting for the group. The **preempt** command causes the failover group to become active on the designated unit automatically when that unit becomes available.

You can enter an optional *delay* value, which specifies the number of seconds the failover group remains active on the current unit before automatically becoming active on the designated unit. Valid values are from 1 to 1200.

If Stateful Failover is enabled, the preemption is delayed until the connections are replicated from the unit on which the failover group is currently active.

If you manually fail over, the **preempt** command is ignored.

Step 10 Create failover group 2 and assign it to the secondary unit:

failover group 2

secondary

preempt [*delay*]

Example:

```
ciscoasa(config-fover-group)# failover group 2
ciscoasa(config-fover-group)# secondary
ciscoasa(config-fover-group)# preempt 1200
```

Step 11 Enter the context configuration mode for a given context, and assign the context to a failover group:

context *name*

join-failover-group {**1** | **2**}

Example:

```
ciscoasa(config)# context Eng
ciscoasa(config-ctx)# join-failover-group 2
```

Repeat this command for each context.

Any unassigned contexts are automatically assigned to failover group 1. The admin context is always a member of failover group 1; you cannot assign it to group 2.

Step 12 Enable failover:

failover

Step 13 Save the system configuration to flash memory:

write memory

Examples

The following example configures the failover parameters for the primary unit:

```
failover lan unit primary
failover lan interface folink gigabitethernet0/3

failover interface ip folink 172.27.48.0 255.255.255.254 standby 172.27.48.1
interface gigabitethernet 0/3
  no shutdown
failover link statelink gigabitethernet0/4

failover interface ip statelink 172.27.48.2 255.255.255.254 standby 172.27.48.3
interface gigabitethernet 0/4
  no shutdown
failover group 1
  primary
  preempt
failover group 2
  secondary
  preempt
```

```
context admin
  join-failover-group 1
failover ipsec pre-shared-key a3rynsun
failover
```

Configure the Secondary Unit for Active/Active Failover

The only configuration required on the secondary unit is for the failover link. The secondary unit requires these commands to communicate initially with the primary unit. After the primary unit sends its configuration to the secondary unit, the only permanent difference between the two configurations is the **failover lan unit** command, which identifies each unit as primary or secondary.

Before you begin

- Enable multiple context mode according to Enable or Disable Multiple Context Mode, on page 224.
- You will not be able to enable failover if any interfaces are configured for DHCP.
- Do not configure a **nameif** for the failover and state links.
- Complete this procedure in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.

Procedure

Step 1 Re-enter the exact same commands as on the primary unit *except* for the **failover lan unit primary** command. You can optionally replace it with the **failover lan unit secondary** command, but it is not necessary because **secondary** is the default setting. You also do not need to enter the **failover group** and **join-failover-group** commands, as they are replicated from the primary unit. See Configure the Primary Unit for Active/Active Failover, on page 288.

For example:

```
ciscoasa(config)# failover lan interface folink gigabitethernet0/3
INFO: Non-failover interface config is cleared on GigabitEthernet0/3 and its sub-interfaces
ciscoasa(config)# failover interface ip folink 172.27.48.1 255.255.255.0 standby 172.27.48.2
ciscoasa(config)# interface gigabitethernet 0/3
no shutdown
ciscoasa(config)# failover link statelink gigabitethernet0/4
INFO: Non-failover interface config is cleared on GigabitEthernet0/4 and its sub-interfaces
ciscoasa(config)# failover interface ip statelink 172.27.49.1 255.255.255.0 standby
172.27.49.2
ciscoasa(config)# interface gigabitethernet 0/4
no shutdown
ciscoasa(config)# failover ipsec pre-shared-key a3rynsun
ciscoasa(config)# failover
```

Step 2 After the failover configuration syncs from the primary unit, save the configuration to flash memory:

ciscoasa(config)# **write memory**

Step 3 If necessary, force failover group 2 to be active on the secondary unit:

failover active group 2

Configure Optional Failover Parameters

You can customize failover settings as desired.

Configure Failover Criteria and Other Settings

See Defaults for Failover, on page 283 for the default settings for many parameters that you can change in this section. For Active/Active mode, you set most criteria per failover group.

Before you begin

- Configure these settings in the system execution space in multiple context mode.
- For Bidirectional Forwarding Detection (BFD) for unit health monitoring, see the following limitations:
  - Firepower 9300 and 4100 only.
  - Active/Standby only.
  - Routed mode only

Procedure

Step 1 Change the unit poll and hold times:

failover polltime [**unit**] [**msec**] *poll_time* [**holdtime** [**msec**] *time*]

Example:

```
ciscoasa(config)# failover polltime unit msec 200 holdtime msec 800
```

The **polltime** range is between 1 and 15 seconds or between 200 and 999 milliseconds. The **holdtime** range is between 1and 45 seconds or between 800 and 999 milliseconds.You cannot enter a holdtime value that is less than 3 times the unit poll time. With a faster poll time, the ASA can detect failure and trigger failover faster. However, faster detection can cause unnecessary switchovers when the network is temporarily congested.

If a unit does not hear hello packet on the failover communication interface for one polling period, additional testing occurs through the remaining interfaces. If there is still no response from the peer unit during the hold time, the unit is considered failed and, if the failed unit is the active unit, the standby unit takes over as the active unit.

In Active/Active mode, you set this rate for the system; you cannot set this rate per failover group.

Step 2 Configure BFD for unit health monitoring.

The regular unit monitoring can cause false alarms when CPU usage is high. The BFD method is distributed, so high CPU does not affect its operation.

a) Define a BFD template to be used for failover health detection:

bfd-template single-hop *template_name*

bfd interval min-tx *milliseconds***min-rx** *milliseconds* **multiplier** *multiplier_value*

Example:

```
ciscoasa(config)# bfd template single-hop failover-temp
ciscoasa(config-bfd)# bfd interval min-tx 50 min-rx 50 multiplier 3
```

The **min-tx** specifies the rate at which BFD control packets are sent to the failover peer. The range is 50 to 999 milliseconds. The **min-rx** specifies the rate at which BFD control packets are expected to be received from the failover peer. The range is 50 to 999 milliseconds. The **multiplier** specifies the number of consecutive BFD control packets that must be missed from a failover peer before BFD declares that the peer is unavailable. The range is 3 to 50.

You can also configure echo and authentication for this template; see Create the BFD Template, on page 889.

b) Enable BFD for health monitoring:

failover health-check bfd *template_name*

Example:

```
ciscoasa(config)# failover health-check bfd failover-temp
```

Step 3 Change the interface link state poll time:

failover polltime link-state msec *poll_time*

Example:

```
ciscoasa(config)# failover polltime link-state msec 300
```

The range is between 300 and 799 milliseconds. By default, each ASA in a failover pair checks the link state of its interfaces every 500 msec. You can customize the polltime; for example, if you set the polltime to 300 msec, the ASA can detect an interface failure and trigger failover faster.

In Active/Active mode, you set this rate for the system; you cannot set this rate per failover group.

Step 4 Set the session replication rate in connections per second:

failover replication rate *conns*

Example:

```
ciscoasa(config)# failover replication rate 20000
```

The minimum and maximum rate is determined by your model. The default is the maximum rate. In Active/Active mode, you set this rate for the system; you cannot set this rate per failover group.

Step 5 Disable the ability to make any configuration changes directly on the standby unit or context:

failover standby config-lock

By default, configurations on the standby unit/context are allowed with a warning message.

Step 6 (Active/Active mode only) Specify the failover group you want to customize:

failover group {**1** | **2**}

Example:

```
ciscoasa(config)# failover group 1
ciscoasa(config-fover-group)#
```

Step 7 Enable HTTP state replication:

- For Active/Standby mode:

  failover replication http

- For Active/Active mode:

  replication http

To allow HTTP connections to be included in the state information replication, you need to enable HTTP replication. We recommend enabling HTTP state replication.

Note
Because of a delay when deleting HTTP flows from the standby unit when using failover, the **show conn count** output might show different numbers on the active unit vs. the standby unit; if you wait several seconds and re-issue the command, you will see the same count on both units.

Step 8 Set the threshold for failover when interfaces fail:

- For Active/Standby mode:

  failover interface-policy *num* [%]

  Example:

  ```
  ciscoasa (config)# failover interface-policy 20%
  ```

- For Active/Active mode:

  interface-policy *num* [%]

  Example:

  ```
  ciscoasa(config-fover-group)# interface-policy 20%
  ```

By default, one interface failure causes failover.

When specifying a specific number of interfaces, the *num* argument can be from 1 to 1025.

When specifying a percentage of interfaces, the *num* argument can be from 1 to 100.

Step 9 Change the interface poll and hold times:

- For Active/Standby mode:

  failover polltime interface [**msec**] *polltime* [**holdtime** *time*]

Example:

```
ciscoasa(config)# failover polltime interface msec 500 holdtime 5
```

- For Active/Active mode:

  polltime interface [**msec**] *polltime* [**holdtime***time*]

  Example:

```
ciscoasa(config-fover-group)# polltime interface msec 500 holdtime 5
```

- *polltime*—Sets how long to wait between sending a hello packet to the peer. Valid values for the polltime are from 1 to 15 seconds or, if the optional **msec** keyword is used, from 500 to 999 milliseconds. The default is 5 seconds.
- **holdtime***time*—Sets the time (as a calculation) between the last-received hello message from the peer unit and the commencement of interface tests to determine the health of the interface. It also sets the duration of each interface test as holdtime/16. Valid values are from 5 to 75 seconds. The default is 5 times the polltime. You cannot enter a holdtime value that is less than five times the polltime.

  To calculate the time before starting interface tests (y):

  a. x = (*holdtime*/*polltime*)/2, rounded to the nearest integer. (.4 and down rounds down; .5 and up rounds up.)

  b. y = x**polltime*

  For example, if you use the default holdtime of 25 and polltime of 5, then y = 15 seconds.

Step 10 Configure the virtual MAC address for an interface:

- For Active/Standby mode:

  failover mac address *phy_if active_mac standby_mac*

  Example:

```
ciscoasa(config)# failover mac address gigabitethernet0/2 00a0.c969.87c8 00a0.c918.95d8
```

- For Active/Active mode:

  mac address *phy_if active_mac standby_mac*

  Example:

```
ciscoasa(config-fover-group)# mac address gigabitethernet0/2 00a0.c969.87c8 00a0.c918.95d8
```

The *phy_if* argument is the physical name of the interface, such as gigabitethernet0/1.

The *active_mac* and *standby_mac* arguments are MAC addresses in H.H.H format, where H is a 16-bit hexadecimal digit. For example, the MAC address 00-0C-F1-42-4C-DE would be entered as 000C.F142.4CDE.

The *active_mac* address is associated with the active IP address for the interface, and the *standby_mac* is associated with the standby IP address for the interface.

You can also set the MAC address using other commands or methods, but we recommend using only one method. If you set the MAC address using multiple methods, the MAC address used depends on many variables, and might not be predictable.

Use the **show interface** command to display the MAC address used by an interface.

Step 11 (Active/Active mode only) Repeat this procedure for the other failover group.

Configure Interface Monitoring

By default, monitoring is enabled on all physical interfaces, or for the Firepower 1010, all VLAN interfaces. Firepower 1010 switch ports are not elegible for interface monitoring.

You might want to exclude interfaces attached to less critical networks from affecting your failover policy.

You can monitor up to 1025 interfaces on a unit (across all contexts in multiple context mode).

Before you begin

In multiple context mode, configure interfaces within each context.

Procedure

Enable or disable health monitoring for an interface:

[**no**] **monitor-interface** {*if_name*}

Example:

```
ciscoasa(config)# monitor-interface inside
ciscoasa(config)# no monitor-interface eng1
```

Configure Support for Asymmetrically Routed Packets (Active/Active Mode)

When running in Active/Active failover, a unit might receive a return packet for a connection that originated through its peer unit. Because the ASA that receives the packet does not have any connection information for the packet, the packet is dropped. This drop most commonly occurs when the two ASAs in an Active/Active failover pair are connected to different service providers and the outbound connection does not use a NAT address.

You can prevent the return packets from being dropped by allowing asymmetrically routed packets. To do so, you assign the similar interfaces on each ASA to the same ASR group. For example, both ASAs connect to the inside network on the inside interface, but connect to separate ISPs on the outside interface. On the primary unit, assign the active context outside interface to ASR group 1; on the secondary unit, assign the active context outside interface to the same ASR group 1. When the primary unit outside interface receives

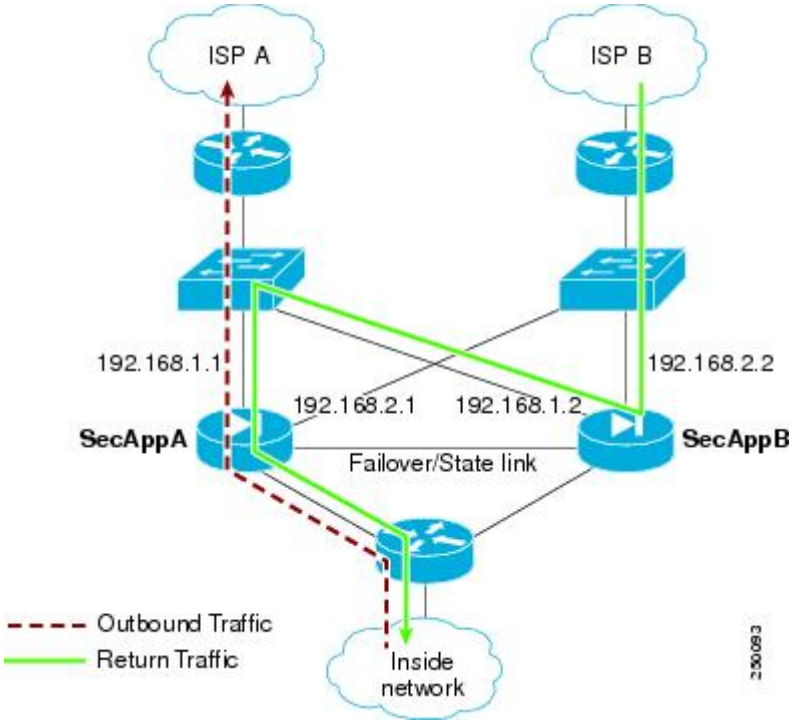
a packet for which it has no session information, it checks the session information for the other interfaces in standby contexts that are in the same group; in this case, ASR group 1. If it does not find a match, the packet is dropped. If it finds a match, then one of the following actions occurs:

- If the incoming traffic originated on a peer unit, some or all of the layer 2 header is rewritten and the packet is redirected to the other unit. This redirection continues as long as the session is active.
- If the incoming traffic originated on a different interface on the same unit, some or all of the layer 2 header is rewritten and the packet is reinjected into the stream.

Note This feature does not provide asymmetric routing; it restores asymmetrically routed packets to the correct interface.

The following figure shows an example of an asymmetrically routed packet.

Figure 45: ASR Example



1. An outbound session passes through the ASA with the active SecAppA context. It exits interface outside ISP-A (192.168.1.1).
2. Because of asymmetric routing configured somewhere upstream, the return traffic comes back through the interface outsideISP-B (192.168.2.2) on the ASA with the active SecAppB context.
3. Normally the return traffic would be dropped because there is no session information for the traffic on interface 192.168.2.2. However, the interface is configured as part of ASR group 1. The unit looks for the session on any other interface configured with the same ASR group ID.

4. The session information is found on interface outsideISP-A (192.168.1.2), which is in the standby state on the unit with SecAppB. Stateful Failover replicated the session information from SecAppA to SecAppB.
5. Instead of being dropped, the layer 2 header is rewritten with information for interface 192.168.1.1 and the traffic is redirected out of the interface 192.168.1.2, where it can then return through the interface on the unit from which it originated (192.168.1.1 on SecAppA). This forwarding continues as needed until the session ends.

Before you begin

- Stateful Failover—Passes state information for sessions on interfaces in the active failover group to the standby failover group.
- Replication HTTP—HTTP session state information is not passed to the standby failover group, and therefore is not present on the standby interface. For the ASA to be able to re-route asymmetrically routed HTTP packets, you need to replicate the HTTP state information.
- Perform this procedure within each active context on the primary and secondary units.
- You cannot configure both ASR groups and traffic zones within a context. If you configure a zone in a context, none of the context interfaces can be part of an ASR group.

Procedure

Step 1 On the primary unit, specify the interface for which you want to allow asymmetrically routed packets:

interface *phy_if*

Example:

```
primary/admin(config)# interface gigabitethernet 0/0
```

Step 2 Set the ASR group number for the interface:

asr-group *num*

Example:

```
primary/admin(config-ifc)# asr-group 1
```

Valid values for *num* range from 1 to 32.

Step 3 On the secondary unit, specify the similar interface for which you want to allow asymmetrically routed packets:

interface *phy_if*

Example:

```
secondary/ctx1(config)# interface gigabitethernet 0/1
```

Step 4 Set the ASR group number for the interface to match the primary unit interface:

asr-group *num*

Example:

```
secondary/ctx1(config-ifc)# asr-group 1
```

Examples

The two units have the following configuration (configurations show only the relevant commands). The device labeled SecAppA in the diagram is the primary unit in the failover pair.

Primary Unit System Configuration

```
interface GigabitEthernet0/1
  description LAN/STATE Failover Interface
interface GigabitEthernet0/2
  no shutdown
interface GigabitEthernet0/3
  no shutdown
interface GigabitEthernet0/4
  no shutdown
interface GigabitEthernet0/5
  no shutdown
failover
failover lan unit primary
failover lan interface folink GigabitEthernet0/1
failover link folink
failover interface ip folink 10.0.4.1 255.255.255.0 standby 10.0.4.11
failover group 1
  primary
failover group 2
  secondary
admin-context SecAppA
context admin
  allocate-interface GigabitEthernet0/2
  allocate-interface GigabitEthernet0/3
  config-url flash:/admin.cfg
  join-failover-group 1
context SecAppB
  allocate-interface GigabitEthernet0/4
  allocate-interface GigabitEthernet0/5
  config-url flash:/ctx1.cfg
  join-failover-group 2
```

SecAppA Context Configuration

```
interface GigabitEthernet0/2
  nameif outsideISP-A
  security-level 0
  ip address 192.168.1.1 255.255.255.0 standby 192.168.1.2
  asr-group 1
interface GigabitEthernet0/3
  nameif inside
  security-level 100
  ip address 10.1.0.1 255.255.255.0 standby 10.1.0.11
monitor-interface outside
```

SecAppB Context Configuration

```
interface GigabitEthernet0/4
  nameif outsideISP-B
  security-level 0
  ip address 192.168.2.2 255.255.255.0 standby 192.168.2.1
  asr-group 1
interface GigabitEthernet0/5
  nameif inside
  security-level 100
  ip address 10.2.20.1 255.255.255.0 standby 10.2.20.11
```

Manage Failover

This section describes how to manage Failover units after you enable Failover, including how to change the Failover setup and how to force failover from one unit to another.

Force Failover

To force the standby unit to become active, perform the following procedure.

Before you begin

In multiple context mode, perform this procedure in the System execution space.

Procedure

Step 1 Force a failover when entered on the *standby* unit. The standby unit becomes the active unit.

If you specify the **group** *group_id*, then this command forces a failover when entered on the *standby* unit for the specified Active/Active failover group. The standby unit becomes the active unit for the failover group.

- For Active/Standby mode on the standby unit:

  failover active

- For Active/Active mode on the standby unit:

  failover active [**group** *group_id*]

  Example:

  ```
  standby# failover active group 1
  ```

Step 2 Force a failover when entered on the *active* unit. The active unit becomes the standby unit.

If you specify the **group** *group_id*, then this command forces a failover when entered on the *active* unit for the specified failover group. The active unit becomes the standby unit for the failover group.

- For Active/Standby mode on the active unit:

  no failover active

- For Active/Active mode on the active unit:

  no failover active [**group** *group_id*]

  Example:

```
active# no failover active group 1
```

Disable Failover

Disabling failover on one or both units causes the active and standby state of each unit to be maintained until you reload. For an Active/Active failover pair, the failover groups remain in the active state on whichever unit they are active, no matter which unit they are configured to prefer.

See the following characteristics when you disable failover:

- The standby unit/context remains in standby mode so that both units do not start passing traffic (this is called a pseudo-standby state).
- The standby unit/context continues to use its standby IP addresses even though it is no longer connected to an active unit/context.
- The standby unit/context continues to listen for a connection on the failover link. If failover is re-enabled on the active unit/context, then the standby unit/context resumes ordinary standby status after re-synchronizing the rest of its configuration.
- Do not enable failover manually on the standby unit to make it active; instead see Force Failover, on page 302. If you enable failover on the standby unit, you will see a MAC address conflict that can disrupt IPv6 traffic.
- To truly disable failover, save the no failover configuration to the startup configuration, and then reload.

Before you begin

In multiple context mode, perform this procedure in the system execution space.

Procedure

Step 1 Disable failover:

no failover

Step 2 To completely disable failover, save the configuration and reload:

write memory

reload

Restore a Failed Unit

To restore a failed unit to an unfailed state, perform the following procedure.

Before you begin

In multiple context mode, perform this procedure in the System execution space.

Procedure

Step 1 Restore a failed unit to an unfailed state:

- For Active/Standby mode:

  failover reset

- For Active/Active mode:

  failover reset [**group** *group_id*]

  Example:

```
ciscoasa(config)# failover reset group 1
```

Restoring a failed unit to an unfailed state does not automatically make it active; restored units remain in the standby state until made active by failover (forced or natural). An exception is a failover group (Active/Active mode only) configured with failover preemption. If previously active, a failover group becomes active if it is configured with preemption and if the unit on which it failed is the preferred unit.

If you specify the **group** *group_id*, this command restores a failed Active/Active failover group to an unfailed state.

Step 2 (Active/Active mode only) To reset failover at the failover group level:

a) In the System choose **Monitoring > Failover > Failover Group** #, where # is the number of the failover group you want to control.
b) Click **Reset Failover**.

Re-Sync the Configuration

If you enter the write standby command on the active unit, the standby unit clears its running configuration (except for the failover commands used to communicate with the active unit), and the active unit sends its entire configuration to the standby unit.

For multiple context mode, when you enter the **write standby** command in the system execution space, all contexts are replicated. If you enter the **write standby** command within a context, the command replicates only the context configuration.

Replicated commands are stored in the running configuration.

Note If the maximum interface limit is reached, the number of VPN tunnels on the active and standby units may be inconsistent. In this situation run the**write standby**command on the active unit to sync the configuration.

Test the Failover Functionality

To test failover functionality, perform the following procedure.

Procedure

Step 1 Test that your active unit is passing traffic as expected by using FTP (for example) to send a file between hosts on different interfaces.

Step 2 Force a failover by entering the following command on the active unit:

Active/Standby mode:

ciscoasa(config)# **no failover active**

Active/Active mode:

ciscoasa(config)# **no failover active group** *group_id*

Step 3 Use FTP to send another file between the same two hosts.

Step 4 If the test was not successful, enter the **show failover** command to check the failover status.

Step 5 When you are finished, you can restore the unit to active status by enter the following command on the newly active unit:

Active/Standby mode:

ciscoasa(config)# **no failover active**

Active/Active mode:

ciscoasa(config)# **failover active group** *group_id*

Note
When an ASA interface goes down, for failover it is still considered to be a unit issue. If the ASA detects that an interface is down, failover occurs immediately, without waiting for the interface holdtime. The interface holdtime is only useful when the ASA considers its status to be OK, although it is not receiving hello packets from the peer. To simulate interface holdtime, shut down the VLAN on the switch to prevent peers from receiving hello packets from each other.

Remote Command Execution

Remote command execution lets you send commands entered at the command line to a specific failover peer.

Send a Command

Because configuration commands are replicated from the active unit or context to the standby unit or context, you can use the **failover exec** command to enter configuration commands on the correct unit, no matter which unit you are logged in to. For example, if you are logged in to the standby unit, you can use the **failover exec active** command to send configuration changes to the active unit. Those changes are then replicated to the standby unit. Do not use the **failover exec** command to send configuration commands to the standby unit or context; those configuration changes are not replicated to the active unit and the two configurations will no longer be synchronized.

Output from configuration, exec, and **show** commands is displayed in the current terminal session, so you can use the **failover exec** command to issue **show** commands on a peer unit and view the results in the current terminal.

You must have sufficient privileges to execute a command on the local unit to execute the command on the peer unit.

Procedure

Step 1 If you are in multiple context mode, use the **changeto context***name* command to change to the context you want to configure. You cannot change contexts on the failover peer with the **failover exec** command.

Step 2 Use the following command to send commands to he specified failover unit:

ciscoasa(config)# **failover exec** {**active** | **mate** | **standby**}

Use the **active** or **standby** keyword to cause the command to be executed on the specified unit, even if that unit is the current unit. Use the **mate** keyword to cause the command to be executed on the failover peer.

Commands that cause a command mode change do not change the prompt for the current session. You must use the **show failover exec** command to display the command mode the command is executed in. See Change Command Modes, page 7-53 for more information.

Change Command Modes

The **failover exec** command maintains a command mode state that is separate from the command mode of your terminal session. By default, the **failover exec** command mode starts in global configuration mode for the specified device. You can change that command mode by sending the appropriate command (such as the **interface** command) using the **failover exec** command. The session prompt does not change when you change modes using **failover exec**.

For example, if you are logged into global configuration mode of the active unit of a failover pair, and you use the **failover exec active** command to change to interface configuration mode, the terminal prompt remains in global configuration mode, but commands entered using **failover exec** are entered in interface configuration mode.

The following examples show the difference between the terminal session mode and the **failover exec** command mode. In the example, the administrator changes the **failover exec** mode on the active unit to interface configuration mode for the interface GigabitEthernet0/1. After that, all commands entered using **failover exec active** are sent to interface configuration mode for interface GigabitEthernet0/1. The administrator then uses failover exec active to assign an IP address to that interface. Although the prompt indicates global configuration mode, the **failover exec active** mode is in interface configuration mode.

```
ciscoasa(config)# failover exec active interface GigabitEthernet0/1
ciscoasa(config)# failover exec active ip address 192.168.1.1 255.255.255.0 standby
192.168.1.2
ciscoasa(config)# router rip
ciscoasa(config-router)#
```

Changing commands modes for your current session to the device does not affect the command mode used by the **failover exec** command. For example, if you are in interface configuration mode on the active unit, and you have not changed the **failover exec** command mode, the following command would be executed in global configuration mode. The result would be that your session to the device remains in interface configuration mode, while commands entered using **failover exec active** are sent to router configuration mode for the specified routing process.

```
ciscoasa(config-if)# failover exec active router ospf 100
ciscoasa(config-if)#
```

Use the **show failover exec** command to display the command mode on the specified device in which commands sent with the **failover exec** command are executed. The **show failover exec** command takes the same keywords as the **failover exec** command: **active**, **mate**, or **standby**. The **failover exec** mode for each device is tracked separately.

For example, the following is sample output from the **show failover exec** command entered on the standby unit:

```
ciscoasa(config)# failover exec active interface GigabitEthernet0/1
ciscoasa(config)# sh failover exec active
Active unit Failover EXEC is at interface sub-command mode

ciscoasa(config)# sh failover exec standby
Standby unit Failover EXEC is at config mode

ciscoasa(config)# sh failover exec mate
Active unit Failover EXEC is at interface sub-command mode
```

Security Considerations

The **failover exec** command uses the failover link to send commands to and receive the output of the command execution from the peer unit. You should enable encryption on the failover link to prevent eavesdropping or man-in-the-middle attacks.

Limitations of Remote Command Execution

When you use remote commands, you face the following limitations:

- If you upgrade one unit using the zero-downtime upgrade procedure and not the other, both units must be running software that supports the **failover exec** command.
- Command completion and context help is not available for the commands in the *cmd_string* argument.
- In multiple context mode, you can only send commands to the peer context on the peer unit. To send commands to a different context, you must first change to that context on the unit to which you are logged in.
- You cannot use the following commands with the **failover exec** command:

- **changeto**
- **debug** (**undebug**)

- If the standby unit is in the failed state, it can still receive commands from the **failover exec** command if the failure is due to a service card failure; otherwise, the remote command execution will fail.
- You cannot use the **failover exec** command to switch from privileged EXEC mode to global configuration mode on the failover peer. For example, if the current unit is in privileged EXEC mode, and you enter **failover exec mate configure terminal**, the **show failover exec mate** output will show that the failover exec session is in global configuration mode. However, entering configuration commands for the peer unit using **failover exec** will fail until you enter global configuration mode on the current unit.
- You cannot enter recursive failover exec commands, such as the **failover exec mate failover exec mate** command.
- Commands that require user input or confirmation must use the **noconfirm** option. For example, to reload the mate, enter:

  failover exec mate reload noconfirm

Monitoring Failover

Monitoring Failover is a system capability that enables you to view the Failover status.

Failover Messages

When a failover occurs, both ASAs send out system messages.

Failover Syslog Messages

The ASA issues a number of syslog messages related to failover at priority level 2, which indicates a critical condition. To view these messages, see the syslog messages guide. The ranges of message IDs associated with failover are: 101xxx, 102xxx, 103xxx, 104xxx, 105xxx, 210xxx, 311xxx, 709xxx, 727xxx. For example, 105032 and 105043 indicate a problem with the failover link.



Note During failover, the ASA logically shuts down and then brings up interfaces, generating syslog messages 411001 and 411002. This is normal activity.

Failover Debug Messages

To see debug messages, enter the **debug fover** command. See the command reference for more information.



Note Because debugging output is assigned high priority in the CPU process, it can drastically affect system performance. For this reason, use the **debug fover** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC.

SNMP Failover Traps

To receive SNMP syslog traps for failover, configure the SNMP agent to send SNMP traps to SNMP management stations, define a syslog host, and compile the Cisco syslog MIB into your SNMP management station.

Monitoring Failover Status

To monitor failover status, enter one of the following commands:

- **show failover**

  Displays information about the failover state of the unit.

- **show failover group**

  Displays information about the failover state of the failover group. The information displayed is similar to that of the **show failover** command but limited to the specified group.

- **show monitor-interface**

  Displays information about the monitored interface.

- **show running-config failover**

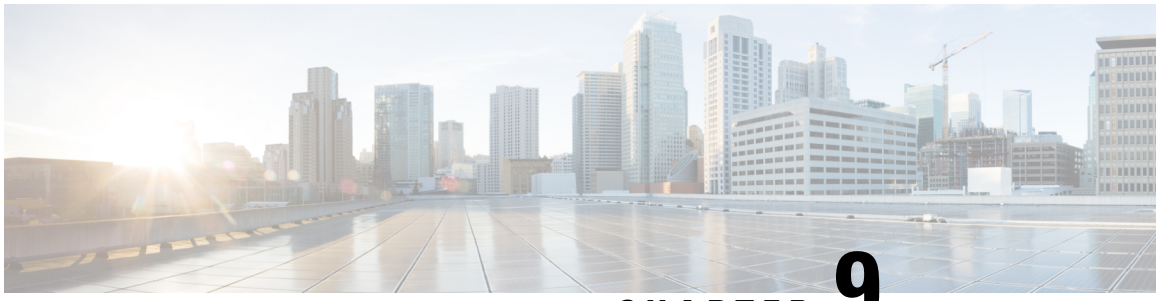
  Displays the failover commands in the running configuration.

History for Failover

Feature Name	Releases	Feature Information
Active/Standby failover	7.0(1)	This feature was introduced.
Active/Active failover	7.0(1)	This feature was introduced.
Support for a hex value for the failover key	7.0(4)	You can now specify a hex value for failover link encryption. We modified the following command: **failover key hex**.
Support for the master passphrase for the failover key	8.3(1)	The failover key now supports the master passphrase, which encrypts the shared key in the running and startup configuration. If you are copying the shared secret from one ASA to another, for example from the **more system:running-config** command, you can successfully copy and paste the encrypted shared key. **Note** The **failover key** shared secret shows as ***** in **show running-config** output; this obscured key is not copyable. We modified the following command: **failover key** [**0** | **8**].
IPv6 support for failover added.	8.2(2)	We modified the following commands: **failover interface ip**, **show failover**, **ipv6 address**, **show monitor-interface**.

Feature Name	Releases	Feature Information
Change to failover group unit preference during "simultaneous" bootup.	9.0(1)	Earlier software versions allowed “simultaneous” boot up so that the failover groups did not require the **preempt** command to become active on the preferred unit. However, this functionality has now changed so that both failover groups become active on the first unit to boot up.
Support for IPsec LAN-to-LAN tunnels to encrypt failover and state link communications	9.1(2)	Instead of using the proprietary encryption for the failover key (the **failover key** command), you can now use an IPsec LAN-to-LAN tunnel for failover and state link encryption. **Note** Failover LAN-to-LAN tunnels do not count against the IPsec (Other VPN) license. We introduced or modified the following commands: **failover ipsec pre-shared-key**, **show vpn-sessiondb**.
Disable health monitoring of a hardware module	9.3(1)	By default, the ASA monitors the health of an installed hardware module such as the ASA FirePOWER module. If you do not want a hardware module failure to trigger failover, you can disable module monitoring. We modified the following command: **monitor-interface service-module**
Lock configuration changes on the standby unit or standby context in a failover pair	9.3(2)	You can now lock configuration changes on the standby unit (Active/Standby failover) or the standby context (Active/Active failover) so you cannot make changes on the standby unit outside normal configuration syncing. We introduced the following command: **failover standby config-lock**
Enable use of the Management 1/1 interface as the failover link on the ASA 5506H	9.5(1)	On the ASA 5506H only, you can now configure the Management 1/1 interface as the failover link. This feature lets you use all other interfaces on the device as data interfaces. Note that if you use this feature, you cannot use the ASA Firepower module, which requires the Management 1/1 interface to remain as a regular management interface. We modified the following commands: **failover lan interface, failover link**
Carrier Grade NAT enhancements now supported in failover and ASA clustering	9.5(2)	For carrier-grade or large-scale PAT, you can allocate a block of ports for each host, rather than have NAT allocate one port translation at a time (see RFC 6888). This feature is now supported in failover and ASA cluster deployments. We modified the following command: **show local-host**

Feature Name	Releases	Feature Information
Improved sync time for dynamic ACLs from AnyConnect Client when using Active/Standby failover	9.6(2)	When you use AnyConnect Client on a failover pair, then the sync time for the associated dynamic ACLs (dACLs) to the standby unit is now improved. Previously, with large dACLs, the sync time could take hours during which time the standby unit is busy syncing instead of providing high availability backup. We did not modify any commands.
Stateful failover for AnyConnect Client connections in multiple context mode	9.6(2)	Stateful failover is now supported for AnyConnect Client connections in multiple context mode. We did not modify any commands.
Interface link state monitoring polling for failover now configurable for faster detection	9.7(1)	By default, each ASA in a failover pair checks the link state of its interfaces every 500 msec. You can now configure the polling interval, between 300 msec and 799 msec; for example, if you set the polltime to 300 msec, the ASA can detect an interface failure and trigger failover faster. We introduced the following command: **failover polltime link-state**
Bidirectional Forwarding Detection (BFD) support for Active/Standby failover health monitoring on the Firepower 9300 and 4100	9.7(1)	You can enable Bidirectional Forwarding Detection (BFD) for the failover health check between two units of an Active/Standby pair on the Firepower 9300 and 4100. Using BFD for the health check is more reliable than the default health check method and uses less CPU. We introduced the following command: **failover health-check bfd**
Disable failover delay	9.15(1)	When you use bridge groups or IPv6 DAD, when a failover occurs the new active unit waits up to 3000 ms for the standby unit to finish networking tasks and transition to the standby state. Then the active unit can start passing traffic. To avoid this delay, you can disable the waiting time, and the active unit will start passing traffic before the standby unit transitions. New/Modified commands: **failover wait-disable**



CHAPTER 9

Failover for High Availability in the Public Cloud

This chapter describes how to configure Active/Backup failover to accomplish high availability of the ASAv in a public cloud environment, such as Microsoft Azure.

- About Failover in the Public Cloud, on page 313
- Licensing for Failover in the Public Cloud, on page 317
- Defaults for Failover in the Public Cloud, on page 317
- About ASAv High Availability in Microsoft Azure, on page 318
- Configure Active/Backup Failover, on page 320
- Configure Optional Failover Parameters, on page 322
- Enable Active/Backup Failover, on page 327
- Manage Failover in the Public Cloud, on page 329
- Monitor Failover in the Public Cloud, on page 331
- History for Failover in the Public Cloud, on page 332

About Failover in the Public Cloud

To ensure redundancy, you can deploy the ASAv in a public cloud environment in an Active/Backup high availability (HA) configuration. HA in the public cloud implements a stateless Active/Backup solution that allows for a failure of the active ASAv to trigger an automatic failover of the system to the backup ASAv.

The following list describes the primary components in the HA public cloud solution:

- **Active ASAv**—The ASAv in the HA pair that is set up to handle the firewall traffic for the HA peers.
- **Backup ASAv**—The ASAv in the HA pair that is not handling firewall traffic and takes over as the active ASAv in the event of an active ASAv failure. It is referred to as a Backup rather than a Standby because it is does not take on the identify of its peer in the event of a failover.
- **HA Agent**—A lightweight process that runs on the ASAv and determines the HA role (active/backup) of an ASAv, detects failures of its HA peer, and performs actions based on its HA role.

On the physical ASA and the non-public cloud virtual ASA, the system handles failover conditions using gratuitous ARP requests where the backup ASA sends out a gratuitous ARP indicating it is now associated with the active IP and MAC addresses. Most public cloud environments do not allow broadcast traffic of this nature. For this reason, an HA configuration in the public cloud requires ongoing connections be restarted when failover happens.

The health of the active unit is monitored by the backup unit to determine if specific failover conditions are met. If those conditions are met, failover occurs. The failover time can vary from a few seconds to over a minute depending on the responsiveness of the public cloud infrastructure.

About Active/Backup Failover

In Active/Backup failover, one unit is the active unit. It passes traffic. The backup unit does not actively pass traffic or exchange any configuration information with the active unit. Active/Backup failover lets you use a backup ASAv device to take over the functionality of a failed unit. When the active unit fails, it changes to the backup state while the backup unit changes to the active state.

Primary/Secondary Roles and Active/Backup Status

When setting up Active/Backup failover, you configure one unit to be primary and the other as secondary. At this point, the two units act as two separate devices for device and policy configuration, as well as for events, dashboards, reports, and health monitoring.

The main differences between the two units in a failover pair are related to which unit is active and which unit is backup, namely which unit actively passes traffic. Although both units are capable of passing traffic, only the primary unit responds to Load Balancer probes and programs any configured routes to use it as a route destination. The backup unit's primary function is to monitor the health of the primary unit. The primary unit always becomes the active unit if both units start up at the same time (and are of equal operational health).

Failover Connection

The backup ASAv monitors the health of the active ASAv using a failover connection established over TCP:

- The active ASAv acts as a connection server by opening a *listen port*.
- The backup ASAv connects to the active ASAv using *connect port*.
- Typically the *listen port* and the *connect port* are the same, unless your configuration requires some type of network address translation between the ASAv units.

The state of the failover connection detects the failure of the active ASAv. When the backup ASAv sees the failover connection come down, it considers the active ASAv as *failed*. Similarly, if the backup ASAv does not receive a response to a keepalive message sent to the active unit, it considers the active ASAv as *failed*

Related Topics

Polling and Hello Messages

The backup ASAv sends Hello messages over the failover connection to the active ASAv and expects a Hello Response in return. Message timing uses a polling interval, the time period between the receipt of a Hello Response by the backup ASAv unit and the sending of the next Hello message. The receipt of the response is enforced by a receive timeout, called the hold time. If the receipt of the Hello Response times out, the active ASAv is considered to have failed.

The polling and hold time intervals are configurable parameters; see Configure Failover Criteria and Other Settings, on page 322.

Active Unit Determination at Startup

The active unit is determined by the following:

- If a unit boots and detects a peer already running as active, it becomes the backup unit.
- If a unit boots and does not detect a peer, it becomes the active unit.
- If both units boot simultaneously, then the primary unit becomes the active unit, and the secondary unit becomes the backup unit.

Failover Events

In Active/Backup failover, failover occurs on a unit basis. The following table shows the failover action for each failure event. For each failure event, the table shows the failover policy (failover or no failover), the action taken by the active unit, the action taken by the backup unit, and any special notes about the failover condition and actions.

Table 17: Failover Events

Failure Event	Policy	Active Action	Backup Action	Notes
Backup unit sees a failover connection close	Failover	n/a	Become active Mark active as failed	This is the standard failover use case.
Active unit sees a failover connection close	No failover	Mark backup as failed	n/a	Failover to an inactive unit should never occur.
Active unit sees a TCP timeout on failover link	No failover	Mark backup as failed	No action	Failover should not occur if the active unit is not getting a reponse from the backup unit.
Backup unit sees a TCP timeout on failover link	Failover	n/a	Become active Mark active as failed Try to send failover command to active unit	The backup unit assumes that the active unit is unable to continue operation and takes over. In case the active unit is still up, but fails to send a response in time, the backup unit sends the failover command to the active unit.
Active Authentication failed	No failover	No action	No action	Because the backup unit is changing the route tables, it is the only unit that needs to be authenticated to Azure. It does not matter if the active unit is authenticated to Azure or not.
Backup Authentication failed	No failover	Mark backup as unauthenticated	No action	Failover cannot happen if the backup unit is not authenticated to Azure.

Failure Event	Policy	Active Action	Backup Action	Notes
Active unit initiates intentional failover	Failover	Become backup	Become active	The active unit initiates failover by closing the Failover Link Connection. The backup unit sees the connection close and becomes the active unit.
Backup unit initiates intentional failover	Failover	Become backup	Become active	The backup unit initiates failover by sending a failover message to the active unit. When the active unit sees the message, it closes the connection and becomes the backup unit. The backup unit sees the connection close and becomes the active unit.
Formerly active unit recovers	No failover	Become backup	Mark mate as backup	Failover should not occur unless absolutely necessary.
Active unit sees failover message from backup unit	Failover	Become backup	Become active	Can occur if a manual failover was initiated by a user; or the backup unit saw the TCP timeout, but the active unit is able to receive messages from the backup unit.

Guidelines and Limitations

This section includes the guidelines and limitations for this feature.

ASAv Failover for High Availability in the Public Cloud

To ensure redundancy, you can deploy the ASAv in a public cloud environment in an Active/Backup high availability (HA) configuration.

- Supported only on the Microsoft Azure public cloud; when configuring the ASAv VM, the maximum supported number of vCPUs is 16; and the maximum supported memory is 64GB RAM. See the ASAv Getting Started Guide for comprehensive list of supported instances.
- Implements a stateless Active/Backup solution that allows for a failure of the active ASAv to trigger an automatic failover of the system to the backup ASAv.

Limitations

- Failover is on the order of seconds rather than milliseconds.
- The HA role determination and the ability to participate as an HA unit depends on TCP connectivity between HA peers and between an HA unit and the Azure infrastructure. There are several situations where an ASAv will not be able participate as an HA unit:
  - The inability to establish a failover connection to its HA peer.

- The inability to retrieve an authentication token from Azure.
- The inability to authenticate with Azure.

- There is no synching of the configuration from the Active unit to the Backup unit. Each unit must be configured individually with similar configurations for handling failover traffic.
- Failover route-table limitations

  With respect to route-tables for HA in the public cloud:

  - You can configure a maximum of 16 route-tables.
  - Within a route-table, you can configure a maximum of 64 routes.

  In each case the system alerts you when you have reached the limit, with the recommendation to remove a route-table or route and retry.
- No ASDM support.
- No IPSec Remote Access VPN support.



Note See the Cisco Adaptive Security Virtual Appliance (ASAv) Quick Start Guide for information about supported VPN topologies in the public cloud.

- ASAv VM instances must be in the same availability set. If you are a current ASAv user in Azure, you will not be able to upgrade to HA from an existing deployment. You have to delete your instance and deploy the ASAv 4 NIC HA offering from the Azure Marketplace.

Licensing for Failover in the Public Cloud

The ASAv uses Cisco Smart Software Licensing. A smart license is required for regular operation. Each ASAv must be licensed independently with an ASAv platform license. Until you install a license, throughput is limited to 100 Kbps so you can perform preliminary connectivity tests. See the Cisco ASA Series Feature Licenses page to find precise licensing requirements for the ASAv.

Defaults for Failover in the Public Cloud

By default, the failover policy consists of the following:

- Stateless failover only.
- Each unit must be configured individually with similar configurations for handling failover traffic.
- The failover TCP control port number is 44442.
- The Azure Load Balancer health probe port number is 44441.
- The unit poll time is 5 seconds.
- The unit hold time is 15 seconds.

- The ASAv responds to health probes on the primary interface (Management 0/0).
- The ASAv authentication with Azure Service Principal is performed on the primary interface (Management 0/0).

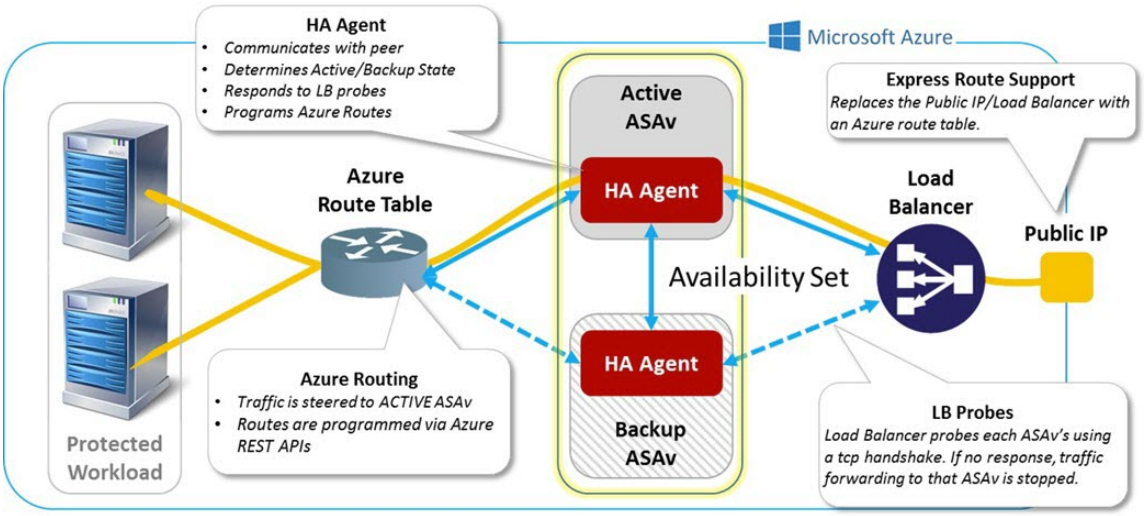
Note See Configure Optional Failover Parameters, on page 322 for options to change the failover port number, health probe port number, poll times, and primary interface.

About ASAv High Availability in Microsoft Azure

The following figure shows a high-level view of an ASAv HA deployment in Azure. A protected workload sits behind two ASAv instances in an Active/Backup failover configuration. An Azure Load Balancer probes both of the ASAv units using a three-way TCP handshake. The active ASAv completes the three way handshake indicating that it is healthy, while the backup ASAv intentionally does not respond. By not responding to the Load Balancer, the backup ASAv appears unhealthy to the Load Balancer, which in turn does not send traffic to it.

On failover, the active ASAv stops responding to the Load Balancer probes and the backup ASAv starts responding, causing all new connections to be sent to the backup ASAv. The backup ASAv sends API requests to the Azure Fabric to modify the route table, redirecting traffic from the active unit to the backup unit. At this point, the backup ASAv becomes the active unit and the active unit becomes the backup unit or is offline, depending on the reason for the failover.

Figure 46: ASAv HA Deployment in Azure



To be able to automatically make API calls to modify Azure route tables, the ASAv HA units need to have Azure Active Directory credentials. Azure employs the concept of a Service Principal which, in simple terms, is a service account. A Service Principal allows you to provision an account with only enough permissions and scope to run a task within a predefined set of Azure resources.

There are two steps to enable your ASAv HA deployment to manage your Azure subscription using a Service Principal:

1. Create an Azure Active Directory application and Service Principal; see About the Azure Service Principal, on page 319.
2. Configure the ASAv instances to authenticate with Azure using a Service Principal; see Configure Authentication Credentials for an Azure Service Principal, on page 324.

Related Topics

See the Azure documentation for more informaion about the Load Balancer.

About the Azure Service Principal

When you have an application that needs to access or modify Azure resources, such as route tables, you must set up an Azure Active Directory (AD) application and assign the required permissions to it. This approach is preferable to running the application under your own credentials because:

- You can assign permissions to the application identity that are different than your own permissions. Typically, these permissions are restricted to exactly what the application needs to do.
- You do not have to change the application's credentials if your responsibilities change.
- You can use a certificate to automate authentication when executing an unattended script.

When you register an Azure AD application in the Azure portal, two objects are created in your Azure AD tenant: an application object, and a service principal object.

- **Application object**—An Azure AD application is defined by its one and only application object, which resides in the Azure AD tenant where the application was registered, known as the application's "home" tenant.
- **Service principal object**—The service principal object defines the policy and permissions for an application's use in a specific tenant, providing the basis for a security principal to represent the application at run-time.

Azure provides instructions on how to create an Azure AD application and service principal in the *Azure Resource Manager Documentation*. See the following topics for complete instructions:

- Use portal to create an Azure Active Directory application and service principal that can access resources
- Use Azure PowerShell to create a service principal to access resource

Note After you set up the service principal, obtain the **Directory ID**, **Application ID**, and **Secret key**. These are required to configure Azure authentication credentials; see Configure Authentication Credentials for an Azure Service Principal, on page 324.

Configuration Requirements for ASAv High Availability in Azure

To deploy a configuration similar to the one described in Figure 46: ASAv HA Deployment in Azure, on page 318 you need the following :

- Azure Authentication information (see About the Azure Service Principal, on page 319):

- Directory ID
- Application ID
- Secret key

- Azure route information (see Configure Azure Route Tables, on page 325):
  - Azure Subscription ID
  - Route table resource group
  - Table names
  - Address prefix
  - Next hop address

- ASA configuration (see Configure Active/Backup Failover, on page 320, Defaults for Failover in the Public Cloud, on page 317):
  - Active/Backup IP addresses
  - HA Agent communication port
  - Load Balancer probe port
  - Polling intervals



Note Configure basic failover settings on both the primary and secondary units. There is no synching of configuration from the primary unit to the secondary unit. Each unit must be configured individually with similar configurations for handling failover traffic.

Configure Active/Backup Failover

To configure Active/Backup failover, configure basic failover settings on both the primary and secondary units. There is no synching of configuration from the primary unit to the secondary unit. Each unit must be configured individually with similar configurations for handling failover traffic.

Before you begin

- Deploy your ASAv HA pair in an Azure Availability Set.
- Have your Azure environment information available, including your Azure Subscription ID and Azure authentication credentials for the Service Principal.

Configure the Primary Unit for Active/Backup Failover

Follow the steps in this section to configure the primary in an Active/Backup failover configuration. These steps provide the minimum configuration needed to enable failover on the primary unit.

Before you begin

- Configure these settings in the system execution space in single context mode.

Example

The following example shows how to configure the failover parameters for the primary/active unit:

```
ciscoasa(config)# failover cloud unit primary
ciscoasa(config)# failover cloud peer ip 10.4.3.5 port 4444
ciscoasa(config)#
```

The peer IP address that you enter here must be the IP address of the management interface on the ASA virtual HA peer.

What to do next

Configure additional parameters as needed:

- Configure the backup unit; see Configure the Secondary Unit for Active/Backup Failover, on page 321.
- Configure Azure authentication; see Configure Authentication Credentials for an Azure Service Principal, on page 324.
- Configure Azure route information; see Configure Azure Route Tables, on page 325.
- Review additional parameters; see Configure Failover Criteria and Other Settings, on page 322.

Configure the Secondary Unit for Active/Backup Failover

Follow the steps in this section to configure the secondary unit in an Active/Backup failover configuration. These steps provide the minimum configuration needed to enable failover to the secondary unit.

Before you begin

- Configure these settings in the system execution space in single context mode.

Procedure

Step 1 Designate this unit as the backup unit:

failover cloud unit secondary

Step 2 Assign the active IP address to the failover link:

failover cloud peer ip *ip-address* [**port** *port-number*]

This IP address is used to establish a TCP failover control connection to the HA peer. The port is used when attempting to open a failover connection to the HA peer, which may already be the active unit. Configuring the port here may be needed if NAT is in place between the HA peers. In most cases you will not need to configure the port.

The peer IP address that you enter here must be the IP address of the management interface on the ASA virtual HA peer.

Example

The following example shows how to configure the failover parameters for the secondary/backup unit:

```
failover cloud unit secondary
failover cloud peer ip 10.4.3.4 port 4444
```

What to do next

Configure additional parameters as needed:

- Configure Azure authentication; see Configure Authentication Credentials for an Azure Service Principal, on page 324.
- Configure Azure route information; see Configure Azure Route Tables, on page 325.
- Review additional parameters; see Configure Failover Criteria and Other Settings, on page 322.

Configure Optional Failover Parameters

You can customize failover settings as necessary.

Configure Failover Criteria and Other Settings

See Defaults for Failover in the Public Cloud, on page 317 for the default settings for many parameters that you can change in this section.

Before you begin

- Configure these settings in the system execution space in single context mode.
- Configure these settings on both the primary and secondary units. There is no synching of configuration from the primary unit to the secondary unit.

Procedure

Step 1 Specify the TCP port to be used for communication with the HA peer:

failover cloud port control *port-number*

Example:

```
ciscoasa(config)# failover cloud port control 4444
```

The *port-number* argument assigns a number for the TCP port used for peer-to-peer communication.

This configures the failover connection TCP port on which to accept connections when in the active unit role. This is the port opened on the active ASAv to which the backup ASAv connects.

Note
We recommend that you keep the default value of 44442, which is the default for both HA peers. If you change the default value for one HA peer, the best practice is to make the same change to the other HA unit.

Step 2 Change the unit poll and hold times:

failover polltime *poll_time* [**holdtime** *time*]

Example:

```
ciscoasa(config)# failover polltime 10 holdtime 30
```

The **polltime** range is between 1 and 15 seconds. The hold time determines how long it takes from the time a hello packet is missed to when the unit is marked as failed. The **holdtime** range is between 3 and 60 seconds. You cannot enter a holdtime value that is less than 3 times the unit poll time. With a faster poll time, the ASA can detect failure and trigger failover faster. However, faster detection can cause unnecessary switchovers when the network is temporarily congested.

Step 3 Specify the TCP port used for Azure Load Balancer health probes:

failover cloud port probe *port-number*

Example:

```
ciscoasa(config)# failover cloud port probe 4443
```

If your deployment uses an Azure Load Balancer, the active ASAv must respond to TCP probes from the load balancer so that incoming connections are directed to the active unit.

Step 4 Specify a secondary interface for Azure Load Balancer health probes:

failover cloud port probe *port-number* **interface** *if-name*

Example:

```
ciscoasa(config)# failover cloud port probe 4443 interface inside
```

The TCP probes used in Cloud HA have a source IP address of 168.63.129.16. This address is Azure's virtual public IP address. This address is the source address of Azure DHCP packets and is the address of the DNS name server in Azure.

By default, the ASAv responds to probes by which 168.63.129.16 is reachable, according to the ASA route tables. This ends up being the primary interface (Management0/0) because of the presence of the default route.

To support load balancers on interfaces other than Management0/0, you configure another interface for the port probe. You also need to configure two static routes: one for the primary interface, and one for the interface configured for load balancer probes.

Step 5 Add static routes for the primary interface and the interface configured for load balancer probes:

route *if-name dest_ip mask gateway_ip* **[distance]**

Example:

```
ciscoasa(config)# route outside 168.63.129.16 255.255.255.255 10.22.0.1 1
ciscoasa(config)# route inside 168.63.129.16 255.255.255.255 10.22.1.1 2
```

The *distance* argument is the administrative distance for the route. The default is **1** if you do not specify a value. Administrative distance is a parameter used to compare routes among different routing protocols. When multiple routes exist to the same destination (168.63.129.16), then the administrative distance for the route determines priority.

The static route for the primary interface (outside) with the the administrative distance of 1 establishes the primary interface as the preferred interface for packets destined to 168.63.129.16, but also allows the interface configured for load balancer probes to send packets to 168.63.129.16.

Note
The mechanism for responding to probes is to create a TCP socket on an interface. Cloud HA uses the route lookup for 168.63.129.16 to decide which interface to create the socket on. This ends up being the primary interface because of the presence of the default route. Without the static route for the interface configured for probes, the ASA would not respond to the TCP packets sent by the load balancer.

Configure Authentication Credentials for an Azure Service Principal

You can enable your ASAv HA peers to access or modify Azure resources, such as route tables, using an Azure Service Principal. You must set up an Azure Active Directory (AD) application and assign the required permissions to it. The following commands allow the ASAv to authenticate with Azure using a Service Principal. See the ASAv Quick Start Guide's Azure chapter for more information about the Azure Service Principal.

Before you begin

- Configure these settings in the system execution space in single context mode.
- Configure these settings on both the primary and secondary units. There is no synching of configuration from the primary unit to the secondary unit.

Procedure

Step 1 Configure the Azure Subscription ID for the Azure Service Principal:

failover cloud subscription-id *subscription-id*

Example:

```
(config)# failover cloud subscription-id ab2fe6b2-c2bd-44
```

The Azure Subscription ID is needed to modify Azure route tables, for example, when the Cloud HA user wants to direct internal routes to the active unit.

Step 2 Configure Azure Service Principal credential information:

failover cloud authentication {**application-id** | **directory-id** | **key**}

To alter Azure route tables during a failover, you need to obtain an *access key* from the Azure infrastructure before you can access route tables. You obtain the *access key* using a application ID, a directory ID, and a secret key for the Azure Service Principal controlling the HA pair.

Step 3 Configure the Azure Service Principal's application ID:

failover cloud authentication application-id *appl-id*

Example:

```
(config)# failover cloud authentication application-id dfa92ce2-fea4-67b3-ad2a-6931704e4201
```

You need this application ID when you request an access key from the Azure infrastructure.

Step 4 Configure the Azure Service Principal's directory ID:

failover cloud authentication directory-id *dir-id*

Example:

```
(config)# failover cloud authentication directory-id 227b0f8f-684d-48fa-9803-c08138b77ae9
```

You need this directory ID when you request an access key from the Azure infrastructure.

Step 5 Configure the Azure Service Principal's secret key ID:

failover cloud authentication key *secret-key* [**encrypt**]

Example:

```
(config)# failover cloud authentication key 5yOhH593dtD/O8gzAlWgulrkWz5dH02d2STk3LDbI4c=
```

You need this secret key when requesting an access key from the Azure infrastructure. If the **encrypt** keyword is present the secret key is encrypted in the **running-config**.

Configure Azure Route Tables

The route table configuration consists of information about Azure user-defined routes that need to be updated when the ASAv assumes the active role. On failover, you want to direct internal routes to the active unit, which uses the configured route table information to automatically direct the routes to itself.

Note You need to configure any Azure route table information on both the active and backup units.

Before you begin

- Configure these settings in the system execution space in single context mode.
- Configure these settings on both the primary and secondary units. There is no synching of configuration from the primary unit to the secondary unit.
- Have your Azure environment information available, including your Azure Subscription ID and Azure authentication credentials for the Service Principal.

Procedure

Step 1 Configure an Azure route table that requires updating during a failover:

failover cloud route-table *table-name* [**subscription-id** *sub-id*]

Example:

```
ciscoasa(config)# failover cloud route-table inside-rt
```

(Optional) To update user-defined routes in more than one Azure subscription, include the **subscription-id** parameter.

Example:

```
ciscoasa(config)# failover cloud route-table inside-rt subscription-id cd5fe6b4-d2ed-45
```

The **subscription-id** parameter at the **route-table** command level overrides the Azure Subscription ID specified at the global level. If you enter the **route-table** command without specifying an Azure Subscription ID, the global **subscription-id** parameter is used. See Configure Authentication Credentials for an Azure Service Principal, on page 324 for information about the Azure Subscription ID .

Note
When you enter the **route-table** command the ASAv switches to **cfg-fover-cloud-rt** mode.

Step 2 Configure an Azure Resource Group for a route table:

rg *resource-group*

Example:

```
ciscoasa(cfg-fover-cloud-rt)# rg east-rg
```

You need a resource group for the route table update requests in Azure.

Step 3 Configure a route that requires updating during a failover:

route name *route-name* **prefix** *address-prefix* **nexthop** *ip-address*

Example:

```
ciscoasa(cfg-fover-cloud-rt)# route route-to-outside prefix 10.4.2.0/24 nexthop 10.4.1.4
```

The address prefix is configured as an IP address prefix, a slash (‘/’) and a numerical netmask. For example *192.120.0.0/16*.

Example

Full configuration example:

```
ciscoasa(config)# failover cloud route-table inside-rt
ciscoasa(cfg-fover-cloud-rt)# rg east-rg
ciscoasa(cfg-fover-cloud-rt)# route route-to-outside prefix 10.4.2.0/24 nexthop 10.4.1.4

ciscoasa(config)# failover cloud route-table outside-rt
ciscoasa(cfg-fover-cloud-rt)# rg east-rg
ciscoasa(cfg-fover-cloud-rt)# route route-to-inside prefix 10.4.1.0/24 nexthop 10.4.2.4
```

Enable Active/Backup Failover

You enable Active/Backup failover after you configure settings on both the primary and secondary units. There is no synching of configuration from the primary unit to the secondary unit. Each unit must be configured individually with similar configurations for handling failover traffic.

Enable the Primary Unit for Active/Backup Failover

Follow the steps in this section to enable the primary in an Active/Backup failover configuration.

Before you begin

- Configure these settings in the system execution space in single context mode.

Procedure

Step 1 Enable failover:

ciscoasa(config)# failover

Step 2 Save the system configuration to flash memory:

ciscoasa(config)# write memory

Example

The following example shows a complete configuration for the primary unit:

```
ciscoasa(config)# failover cloud unit primary
ciscoasa(config)# failover cloud peer ip 10.4.3.4

ciscoasa(config)# failover cloud authentication application-id dfa92ce2-fea4-67b3-ad2a-693170
ciscoasa(config)# failover cloud authentication directory-id 227b0f8f-684d-48fa-9803-c08138
ciscoasa(config)# failover cloud authentication key 5yOhH593dtD/O8gzAWguH02d2STk3LDbI4c=
ciscoasa(config)# failover cloud authentication subscription-id ab2fe6b2-c2bd-44

ciscoasa(config)# failover cloud route-table inside-rt
ciscoasa(cfg-fover-cloud-rt)# rg east-rg
ciscoasa(cfg-fover-cloud-rt)# route route-to-outside prefix 10.4.2.0/24 nexthop 10.4.1.4

ciscoasa(config)# failover cloud route-table outside-rt
ciscoasa(cfg-fover-cloud-rt)# rg east-rg
ciscoasa(cfg-fover-cloud-rt)# route route-to-inside prefix 10.4.1.0/24 nexthop 10.4.2.4

ciscoasa(config)# failover
ciscoasa(config)# write memory
```

What to do next

Enable the secondary unit.

Enable the Secondary Unit for Active/Backup Failover

Follow the steps in this section to enable the secondary in an Active/Backup failover configuration.

Before you begin

- Configure these settings in the system execution space in single context mode.

Procedure

Step 1 Enable failover:

ciscoasa(config)# **failover**

Step 2 Save the system configuration to flash memory:

ciscoasa(config)# **write memory**

Example

The following example shows a complete configuration for the secondary unit:

```
ciscoasa(config)# failover cloud unit secondary
ciscoasa(config)# failover cloud peer ip 10.4.3.5

ciscoasa(config)# failover cloud authentication application-id dfa92ce2-fea4-67b3-ad2a-693170
```

```
ciscoasa(config)# failover cloud authentication directory-id 227b0f8f-684d-48fa-9803-c08138
ciscoasa(config)# failover cloud authentication key 5yOhH593dtD/O8gzAWguH02d2STk3LDbI4c=
ciscoasa(config)# failover cloud authentication subscription-id ab2fe6b2-c2bd-44

ciscoasa(config)# failover cloud route-table inside-rt
ciscoasa(cfg-fover-cloud-rt)# rg east-rg
ciscoasa(cfg-fover-cloud-rt)# route route-to-outside prefix 10.4.2.0/24 nexthop 10.4.1.4

ciscoasa(config)# failover cloud route-table outside-rt
ciscoasa(cfg-fover-cloud-rt)# rg east-rg
ciscoasa(cfg-fover-cloud-rt)# route route-to-inside prefix 10.4.1.0/24 nexthop 10.4.2.4

ciscoasa(config)# failover
ciscoasa(config)# write memory
```

Manage Failover in the Public Cloud

This section describes how to manage Failover units in the Cloud after you enable failover, including how to change to force failover from one unit to another.

Force Failover

To force the standby unit to become active, perform the following command.

Before you begin

Use this command in the system execution space in single context mode.

Procedure

Step 1 Force a failover when entered on the *standby* unit:

failover active

Example:

```
ciscoasa# failover active
```

The standby unit becomes the active unit.

Step 2 Force a failover when entered on the *active* unit:

no failover active

Example:

```
ciscoasa# no failover active
```

The active unit becomes the standby unit.

Update Routes

If the state of the routes in Azure is inconsistent with the ASAv in the *active* role, you can force route updates on the ASAv using the following EXEC command:

Before you begin

Use this command in the system execution space in single context mode.

Procedure

Update the routes on the *active* unit:

failover cloud update routes

Example:

```
ciscoasa# failover cloud update routes
Beginning route-table updates
Routes changed
```

This command is only valid on the ASAv in the *active* role. If authentication fails the command output will be `Route changes failed`.

Validate Azure Authentication

For a successful ASAv HA deployment in Azure, the Service Principal configuration must be complete and accurate. Without proper Azure authorization, the ASAv units will be unable to access resources to handle failover and to perform route updates. You can test your failover configuration to detect errors related to the following elements of your Azure Service Principal:

- Directory ID
- Application ID
- Authentication Key

Before you begin

Use this command in the system execution space in single context mode.

Procedure

Test the Azure authentication elements in your ASAv HA configuration:

test failover cloud authentication

Example:

```
ciscoasa(config)# test failover cloud authentication
Checking authentication to cloud provider
Authentication Succeeded
```

If authentication fails the command output will be `Authentication Failed`.

If the Directory ID or Application ID is not configured properly, Azure will not recognize the resource addressed in the REST request to obtain an authentication token. The event history for this condition entry will read:

```
Error Connection - Unexpected status in response to access token request: Bad Request
```

If the Directory ID or Application ID are correct, but the authentication key is not configured properly, Azure will not grant permission to generate the authentication token. The event history for this condition entry will read:

```
Error Connection - Unexpected status in response to access token request: Unauthorized
```

Monitor Failover in the Public Cloud

This section explains how you monitor the failover status.

Failover Status

To monitor failover status, enter one of the following commands:

- **show failover**

  Displays information about the failover state of the unit. Values for configuration elements that have not been configured will display *not configured*.

  Route update information is presented only for the active unit.

- **show failover history**

  Displays failover event history with a timestamp, severity level, event type, and event text.

Failover Messages

Failover Syslog Messages

The ASA issues a number of syslog messages related to failover at priority level 2, which indicates a critical condition. To view these messages, see the syslog messages guide. Syslog messages are in the ranges of 1045xx and 1055xx.

Note During failover, the ASA logically shuts down and then brings up interfaces, generating syslog messages. This is normal activity.

The following are sample syslogs generated during a switchover:

```
%ASA-3-105509: (Primary) Error sending Hello message to peer unit 10.22.3.5, error: Unknown
 error
%ASA-1-104500: (Primary) Switching to ACTIVE - switch reason: Unable to send message to
Active unit
%ASA-5-105522: (Primary) Updating route-table wc-rt-inside
%ASA-5-105523: (Primary) Updated route-table wc-rt-inside
%ASA-5-105522: (Primary) Updating route-table wc-rt-outside
%ASA-5-105523: (Primary) Updated route-table wc-rt-outside
%ASA-5-105542: (Primary) Enabling load balancer probe responses
%ASA-5-105503: (Primary) Internal state changed from Backup to Active no peer
%ASA-5-105520: (Primary) Responding to Azure Load Balancer probes
```

Each syslog related to a Public Cloud deployment is prefaced with the unit role: (Primary) or (Secondary).

Failover Debug Messages

To see debug messages, enter the **debug fover** command. See the command reference for more information.



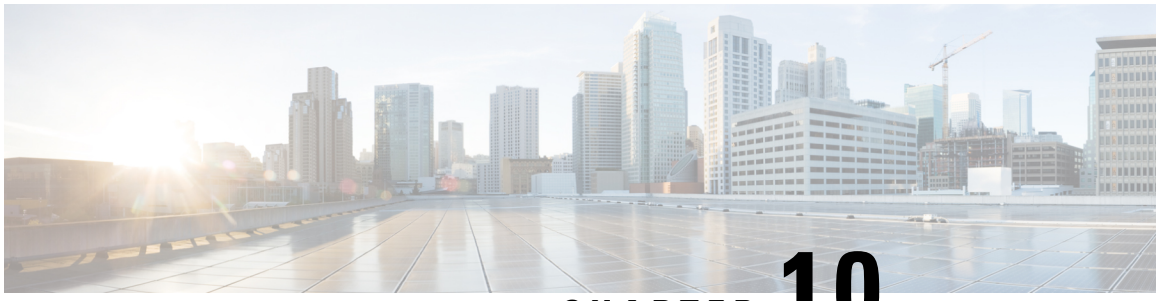
Note Because debugging output is assigned high priority in the CPU process, it can drastically affect system performance. For this reason, use the **debug fover** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC.

SNMP Failover Traps

To receive SNMP syslog traps for failover, configure the SNMP agent to send SNMP traps to SNMP management stations, define a syslog host, and compile the Cisco syslog MIB into your SNMP management station.

History for Failover in the Public Cloud

Feature Name	Releases	Feature Information
Active/Backup failover on Microsoft Azure	9.8(200)	This feature was introduced.



CHAPTER 10

ASA Cluster for the Secure Firewall 3100

Clustering lets you group multiple ASAs together as a single logical device. A cluster provides all the convenience of a single device (management, integration into a network) while achieving the increased throughput and redundancy of multiple devices.

Note Some features are not supported when using clustering. See Unsupported Features with Clustering, on page 408.

- About ASA Clustering, on page 333
- Licenses for ASA Clustering, on page 337
- Requirements and Prerequisites for ASA Clustering, on page 338
- Guidelines for Clustering, on page 340
- Configure ASA Clustering, on page 345
- Manage Cluster Nodes, on page 378
- Monitoring the ASA Cluster, on page 383
- Examples for ASA Clustering, on page 395
- Reference for Clustering, on page 408
- History for ASA Clustering for the Secure Firewall 3100, on page 422

About ASA Clustering

This section describes the clustering architecture and how it works.

How a cluster fits into your network

The cluster consists of multiple firewalls acting as a single unit. To act as a cluster, the firewalls need the following infrastructure:

- Isolated, high-speed backplane network for intra-cluster communication, known as the *cluster control link*.
- Management access to each firewall for configuration and monitoring.

Infrastructure requirements

When you place the cluster in your network, the upstream and downstream routers need to be able to load-balance the data coming to and from the cluster using Spanned EtherChannels. Interfaces on multiple members of the cluster are grouped into a single EtherChannel; the EtherChannel performs load balancing between units.

Cluster Members

Cluster members work together to accomplish the sharing of the security policy and traffic flows. This section describes the nature of each member role.

Bootstrap Configuration

On each device, you configure a minimal bootstrap configuration including the cluster name, cluster control link interface, and other cluster settings. The first node on which you enable clustering typically becomes the *control* node. When you enable clustering on subsequent nodes, they join the cluster as *data* nodes.

Control and data node roles

One member of the cluster is the control node. If multiple cluster nodes come online at the same time, the control node is determined by the priority setting in the bootstrap configuration; the priority is set between 1 and 100, where 1 is the highest priority. All other members are data nodes. Typically, when you first create a cluster, the first node you add becomes the control node simply because it is the only node in the cluster so far.

You must perform all configuration (aside from the bootstrap configuration) on the control node only; the configuration is then replicated to the data nodes. In the case of physical assets, such as interfaces, the configuration of the control node is mirrored on all data nodes. For example, if you configure Ethernet 1/2 as the inside interface and Ethernet 1/1 as the outside interface, then these interfaces are also used on the data nodes as inside and outside interfaces.

Some features do not scale in a cluster, and the control node handles all traffic for those features.

Cluster interfaces

You can configure data interfaces as Spanned EtherChannels. Refer Cluster interfaces, on page 345 for more information.

Interface type restrictions

Note Individual interfaces are not supported, with the exception of a management interface.

Cluster control links

A cluster control link is a dedicated hardware interface that each unit must allocate for cluster communication. For more information, refer Cluster control link, on page 346.

Configuration replication

All nodes in the cluster share a single configuration. You can only make configuration changes on the control node (with the exception of the bootstrap configuration), and changes are automatically synced to all other nodes in the cluster.

ASA Cluster Management

One of the benefits of using ASA clustering is the ease of management. This section describes how to manage the cluster.

Management Network

We recommend connecting all units to a single management network. This network is separate from the cluster control link.

Management Interface

For the management interface, we recommend using one of the dedicated management interfaces. You can configure the management interfaces as Individual interfaces (for both routed and transparent modes) or as a Spanned EtherChannel interface.

We recommend using Individual interfaces for management, even if you use Spanned EtherChannels for your data interfaces. Individual interfaces let you connect directly to each unit if necessary, while a Spanned EtherChannel interface only allows remote connection to the current control unit.

Note You cannot enable dynamic routing for the management interface. You must use a static route.

For an Individual interface, the Main cluster IP address is a fixed address for the cluster that always belongs to the current control unit. For each interface, you also configure a range of addresses so that each unit, including the current control unit, can use a Local address from the range. The Main cluster IP address provides consistent management access to an address; when a control unit changes, the Main cluster IP address moves to the new control unit, so management of the cluster continues seamlessly. The Local IP address is used for routing, and is also useful for troubleshooting.

For example, you can manage the cluster by connecting to the Main cluster IP address, which is always attached to the current control unit. To manage an individual member, you can connect to the Local IP address.

Note To-the-box traffic needs to be directed to the node's management IP address; to-the-box traffic is not forwarded over the cluster control link to any other node.

For outbound management traffic such as TFTP or syslog, each unit, including the control unit, uses the Local IP address to connect to the server.

For a Spanned EtherChannel interface, you can only configure one IP address, and that IP address is always attached to the control unit. You cannot connect directly to a data unit using the EtherChannel interface; we recommend configuring the management interface as an Individual interface so that you can connect to each unit. Note that you can use a device-local EtherChannel for management.

Control Unit Management Vs. Data Unit Management

All management and monitoring can take place on the control node. From the control node, you can check runtime statistics, resource usage, or other monitoring information of all nodes. You can also issue a command to all nodes in the cluster, and replicate the console messages from data nodes to the control node.

You can monitor data nodes directly if desired. Although also available from the control node, you can perform file management on data nodes (including backing up the configuration and updating images). The following functions are not available from the control node:

- Monitoring per-node cluster-specific statistics.
- Syslog monitoring per node (except for syslogs sent to the console when console replication is enabled).
- SNMP
- NetFlow

Crypto Key Replication

When you create a crypto key on the control node, the key is replicated to all data nodes. If you have an SSH session to the Main cluster IP address, you will be disconnected if the control node fails. The new control node uses the same key for SSH connections, so that you do not need to update the cached SSH host key when you reconnect to the new control node.

ASDM Connection Certificate IP Address Mismatch

By default, a self-signed certificate is used for the ASDM connection based on the Local IP address. If you connect to the Main cluster IP address using ASDM, then a warning message about a mismatched IP address might appear because the certificate uses the Local IP address, and not the Main cluster IP address. You can ignore the message and establish the ASDM connection. However, to avoid this type of warning, you can enroll a certificate that contains the Main cluster IP address and all the Local IP addresses from the IP address pool. You can then use this certificate for each cluster member. See https://www.cisco.com/c/en/us/td/docs/security/asdm/identity-cert/cert-install.html for more information.

Inter-Site Clustering

For inter-site installations, you can take advantage of ASA clustering as long as you follow the recommended guidelines.

You can configure each cluster chassis to belong to a separate site ID.

Site IDs work with site-specific MAC addresses and IP addresses. Packets egressing the cluster use a site-specific MAC address and IP address, while packets received by the cluster use a global MAC address and IP address. This feature prevents the switches from learning the same global MAC address from both sites on two different ports, which causes MAC flapping; instead, they only learn the site MAC address. Site-specific MAC addresses and IP address are supported for routed mode using Spanned EtherChannels only.

Site IDs are also used to enable flow mobility using LISP inspection, director localization to improve performance and reduce round-trip time latency for inter-site clustering for data centers, and site redundancy for connections where a backup owner of a traffic flow is always at a different site from the owner.

See the following sections for more information about inter-site clustering:

- Sizing the Data Center Interconnect—Requirements and Prerequisites for ASA Clustering, on page 338
- Inter-Site Guidelines—Guidelines for Clustering, on page 340
- Configure Cluster Flow Mobility—Configure Cluster Flow Mobility, on page 373
- Enable Director Localization—Enable Director Localization, on page 371
- Enable Site Redundancy—Enable Director Localization, on page 371
- Inter-Site Examples—Examples for Inter-Site Clustering, on page 405

Licenses for ASA Clustering

Smart Software Manager Regular and On-Prem

Each unit requires the Standard license (enabled by default) and the same encryption license. We recommend licensing each unit with the licensing server *before* you enable clustering to avoid licensing mismatch issues, and when using the Strong Encryption license, issues with cluster control link encryption.

The clustering feature itself does not require any licenses. There is no extra cost for the Context license on data units.

The Strong Encryption license is automatically enabled for qualified customers when you apply the registration token. For the optional Strong Encryption (3DES/AES) feature license enabled in the ASA configuration, see below.

In the ASA license configuration, the Standard license is always enabled by default on all units. You can only configure smart licensing on the control unit. The configuration is replicated to the data units, but for some licenses, they do not use the configuration; it remains in a cached state, and only the control unit requests the license. The licenses are aggregated into a single cluster license that is shared by the cluster units, and this aggregated license is also cached on the data units to be used if one of them becomes the control unit in the future. Each license type is managed as follows:

- Standard—Each unit requests a Standard license from the server.
- Context—Only the control unit requests the Context license from the server. The Standard license includes 2 contexts by default and is present on all cluster members. The value from each unit's Standard license plus the value of the Context license on the control unit are combined up to the platform limit in an aggregated cluster license. For example:
  - You have 6 Secure Firewall 3100s in the cluster. The Standard license includes 2 contexts; for 6 units, these licenses add up to 12 contexts. You configure an additional 20-Context license on the control unit. Therefore, the aggregated cluster license includes 32 contexts. Because the platform limit for one chassis is 100, the combined license allows a maximum of 100 contexts; the 32 contexts are within the limit. Therefore, you can configure up to 32 contexts on the control unit; each data unit will also have 32 contexts through configuration replication.
  - You have 3 Secure Firewall 3100s in the cluster. The Standard license includes 2 contexts; for 3 units, these licenses add up to 6 contexts. You configure an additional 100-Context license on the control unit. Therefore, the aggregated cluster license includes 106 contexts. Because the platform limit for one unit is 100, the combined license allows a maximum of 100 contexts; the 106 contexts are over the limit. Therefore, you can only configure up to 100 contexts on the control unit; each data unit will also have 100 contexts through configuration replication. In this case, you should only configure the control unit Context license to be 94 contexts.

- Strong Encryption (3DES)—If your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. Only the control unit requests this license, and all units can use it due to license aggregation.

If a new control unit is elected, the new control unit continues to use the aggregated license. It also uses the cached license configuration to re-request the control unit license. When the old control unit rejoins the cluster as a data unit, it releases the control unit license entitlement. Before the data unit releases the license, the control unit's license might be in a non-compliant state if there are no available licenses in the account. The retained license is valid for 30 days, but if it is still non-compliant after the grace period, you will not be able to make configuration changes to features requiring special licenses; operation is otherwise unaffected. The new active unit sends an entitlement authorization renewal request every 35 seconds until the license is compliant. You should refrain from making configuration changes until the license requests are completely processed. If a unit leaves the cluster, the cached control configuration is removed, while the per-unit entitlements are retained. In particular, you would need to re-request the Context license on non-cluster units.

Permanent License Reservation

For permanent license reservation, you must purchase separate licenses for each chassis and enable the licenses *before* you configure clustering.

Requirements and Prerequisites for ASA Clustering

Model Requirements

- Secure Firewall 3100—Maximum 8 nodes

ASA Hardware and Software Requirements

All units in a cluster:

- Must be the same model with the same DRAM. You do not have to have the same amount of flash memory.
- Must run the identical software except at the time of an image upgrade. Hitless upgrade is supported. Mismatched software versions can lead to poor performance, so be sure to upgrade all nodes in the same maintenance window.
- Must be in the same security context mode, single or multiple.
- (Single context mode) Must be in the same firewall mode, routed or transparent.
- New cluster members must use the same SSL encryption setting (the **ssl encryption** command) as the control unit for initial cluster control link communication before configuration replication.

Switch Requirements

- Be sure to complete the switch configuration before you configure clustering on the ASAs.
- For a list of supported switches, see Cisco ASA Compatibility.

ASA Requirements

- Provide each unit with a unique IP address before you join them to the management network.
  - See the Getting Started chapter for more information about connecting to the ASA and setting the management IP address.
  - Except for the IP address used by the control unit (typically the first unit you add to the cluster), these management IP addresses are for temporary use only.
  - After a data unit joins the cluster, its management interface configuration is replaced by the one replicated from the control unit.

Sizing the Data Center Interconnect for Inter-Site Clustering

You should reserve bandwidth on the data center interconnect (DCI) for cluster control link traffic equivalent to the following calculation:

$$\frac{\text{\# of cluster members per site}}{2} \times \text{cluster control link size per member}$$

If the number of members differs at each site, use the larger number for your calculation. The minimum bandwidth for the DCI should not be less than the size of the cluster control link for one member.

For example:

- For 4 members at 2 sites:
  - 4 cluster members total
  - 2 members at each site
  - 5 Gbps cluster control link per member

  Reserved DCI bandwidth = 5 Gbps (2/2 x 5 Gbps).
- For 6 members at 3 sites, the size increases:
  - 6 cluster members total
  - 3 members at site 1, 2 members at site 2, and 1 member at site 3
  - 10 Gbps cluster control link per member

  Reserved DCI bandwidth = 15 Gbps (3/2 x 10 Gbps).
- For 2 members at 2 sites:
  - 2 cluster members total
  - 1 member at each site
  - 10 Gbps cluster control link per member

  Reserved DCI bandwidth = 10 Gbps (1/2 x 10 Gbps = 5 Gbps; but the minimum bandwidth should not be less than the size of the cluster control link (10 Gbps)).

Other Requirements

We recommend using a terminal server to access all cluster member unit console ports. For initial setup, and ongoing management (for example, when a unit goes down), a terminal server is useful for remote management.

Guidelines for Clustering

Context Mode

The mode must match on each member unit.

Firewall Mode

For single mode, the firewall mode must match on all units.

Failover

Failover is not supported with clustering.

IPv6

The cluster control link is only supported using IPv4.

Switches

- Make sure connected switches match the MTU for both cluster data interfaces and the cluster control link interface. You should configure the cluster control link interface MTU to be at least 100 bytes higher than the data interface MTU, so make sure to configure the cluster control link connecting switch appropriately. Because the cluster control link traffic includes data packet forwarding, the cluster control link needs to accommodate the entire size of a data packet plus cluster traffic overhead. In addition, we do not recommend setting the cluster control link MTU between 2561 and 8362; due to block pool handling, this MTU size is not optimal for system operation.
- For Cisco IOS XR systems, if you want to set a non-default MTU, set the IOS XR interface MTU to be 14 bytes higher than the cluster device MTU. Otherwise, OSPF adjacency peering attempts may fail unless the **mtu-ignore** option is used. Note that the cluster device MTU should match the IOS XR *IPv4* MTU. This adjustment is not required for Cisco Catalyst and Cisco Nexus switches.
- On the switch(es) for the cluster control link interfaces, you can optionally enable Spanning Tree PortFast on the switch ports connected to the cluster unit to speed up the join process for new units.
- On the switch, we recommend that you use one of the following EtherChannel load-balancing algorithms: **source-dest-ip** or **src-dst-mixed-ip-port** (see the Cisco Nexus OS and Cisco IOS-XE **port-channel load-balance** command). Do not use a **vlan** keyword in the load-balance algorithm because it can cause unevenly distributed traffic to the devices in a cluster. *Do not* change the load-balancing algorithm from the default on the cluster device.
- If you change the load-balancing algorithm of the EtherChannel on the switch, the EtherChannel interface on the switch temporarily stops forwarding traffic, and the Spanning Tree Protocol restarts. There will be a delay before traffic starts flowing again.

- Switches on the cluster control link path should not verify the L4 checksum. Redirected traffic over the cluster control link does not have a correct L4 checksum. Switches that verify the L4 checksum could cause traffic to be dropped.
- Port-channel bundling downtime should not exceed the configured keepalive interval.
- On Supervisor 2T EtherChannels, the default hash distribution algorithm is adaptive. To avoid asymmetric traffic in a VSS design, change the hash algorithm on the port-channel connected to the cluster device to fixed:

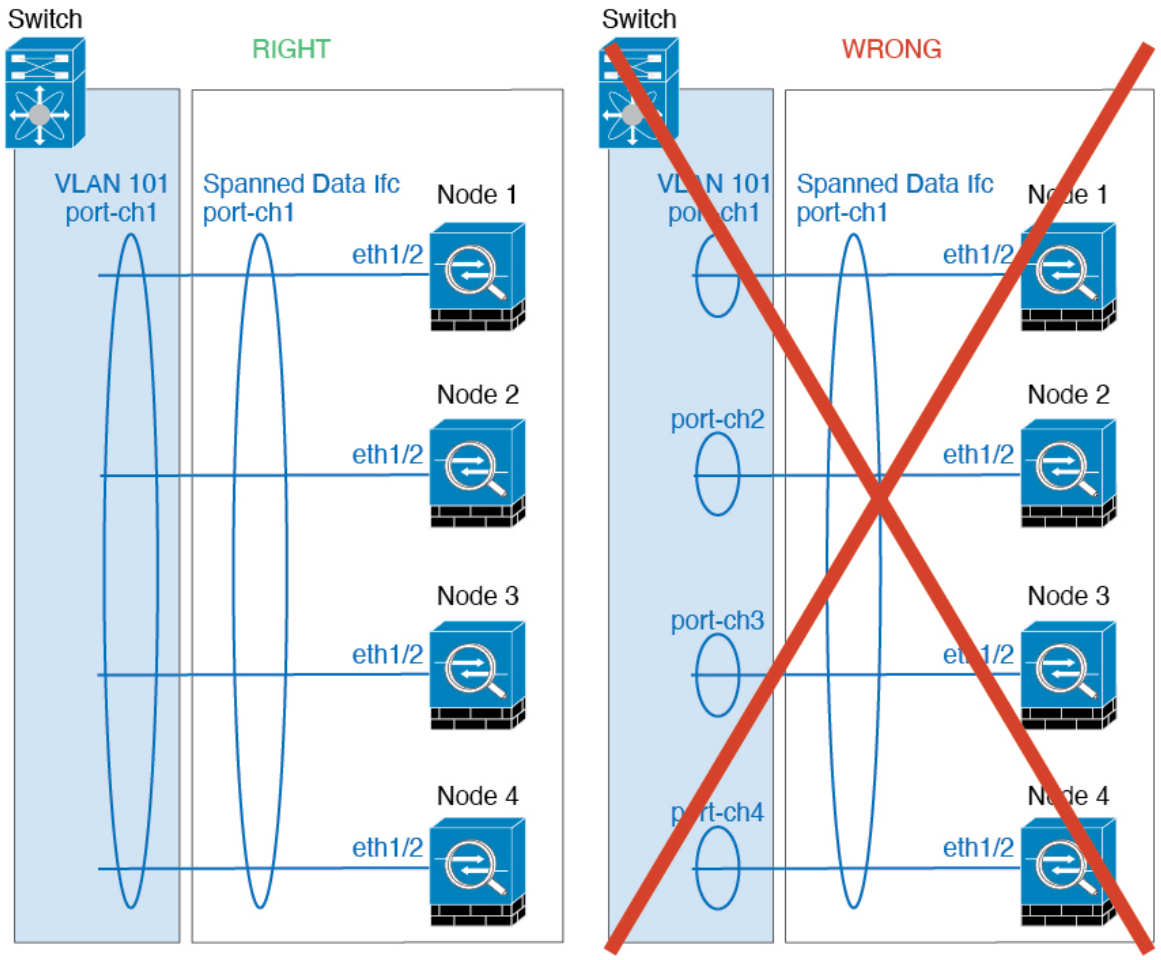
  router(config)# **port-channel** *id* **hash-distribution fixed**

  Do not change the algorithm globally; you may want to take advantage of the adaptive algorithm for the VSS peer link.

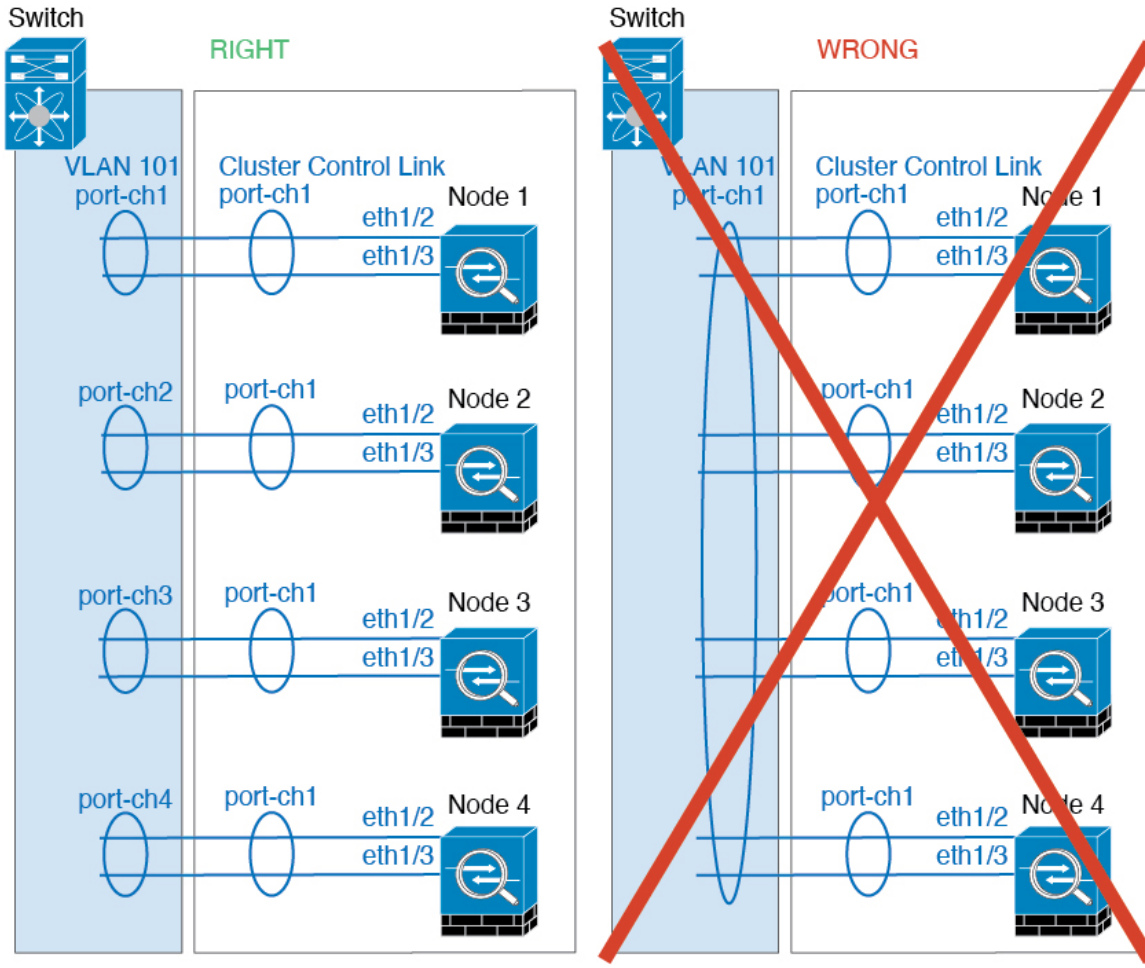
- You should disable the LACP Graceful Convergence feature on all cluster-facing EtherChannel interfaces for Cisco Nexus switches.

EtherChannels

- In Catalyst 3750-X Cisco IOS software versions earlier than 15.1(1)S2, the cluster unit did not support connecting an EtherChannel to a switch stack. With default switch settings, if the cluster unit EtherChannel is connected cross stack, and if the control unit switch is powered down, then the EtherChannel connected to the remaining switch will not come up. To improve compatibility, set the **stack-mac persistent timer** command to a large enough value to account for reload time; for example, 8 minutes or 0 for indefinite. Or, you can upgrade to more a more stable switch software version, such as 15.1(1)S2.
- Spanned vs. Device-Local EtherChannel Configuration—Be sure to configure the switch appropriately for Spanned EtherChannels vs. Device-local EtherChannels.
  - Spanned EtherChannels—For cluster unit *Spanned* EtherChannels, which span across all members of the cluster, the interfaces are combined into a single EtherChannel on the switch. Make sure each interface is in the same channel group on the switch.



- Device-local EtherChannels—For cluster unit *Device-local* EtherChannels including any EtherChannels configured for the cluster control link, be sure to configure discrete EtherChannels on the switch; do not combine multiple cluster unit EtherChannels into one EtherChannel on the switch.



Inter-Site Guidelines

See the following guidelines for inter-site clustering:

- The cluster control link latency must be less than 20 ms round-trip time (RTT).
- The cluster control link must be reliable, with no out-of-order or dropped packets; for example, you should use a dedicated link.
- Do not configure connection rebalancing; you do not want connections rebalanced to cluster members at a different site.
- The ASA does not encrypt forwarded data traffic on the cluster control link because it is a dedicated link, even when used on a Data Center Interconnect (DCI). If you use Overlay Transport Virtualization (OTV), or are otherwise extending the cluster control link outside of the local administrative domain, you can configure encryption on your border routers such as 802.1AE MacSec over OTV.
- The cluster implementation does not differentiate between members at multiple sites for incoming connections; therefore, connection roles for a given connection may span across sites. This is expected behavior. However, if you enable director localization, the local director role is always chosen from the same site as the connection owner (according to site ID). Also, the local director chooses a new owner

at the same site if the original owner fails (Note: if the traffic is asymmetric across sites, and there is continuous traffic from the remote site after the original owner fails, then a node from the remote site might become the new owner if it receives a data packet within the re-hosting window.).

- For director localization, the following traffic types do not support localization: NAT or PAT traffic; SCTP-inspected traffic; Fragmentation owner query.
- For UDP long-lived flows in a North-South deployment, routing loops can occur if nodes at the original flow owner site fail and then come back up, after which the flow is directed back to the original site. If the new owner at the other site doesn't have a route to the destination, it will route the flow back to the internet, causing a loop. In this case, use the **clear conn** command on the new owner to force the flow to be reestablished.
- For transparent mode, if the cluster is placed between a pair of inside and outside routers (AKA North-South insertion), you must ensure that both inside routers share a MAC address, and also that both outside routers share a MAC address. When a cluster member at site 1 forwards a connection to a member at site 2, the destination MAC address is preserved. The packet will only reach the router at site 2 if the MAC address is the same as the router at site 1.
- For transparent mode, if the cluster is placed between data networks and the gateway router at each site for firewalling between internal networks (AKA East-West insertion), then each gateway router should use a First Hop Redundancy Protocol (FHRP) such as HSRP to provide identical virtual IP and MAC address destinations at each site. The data VLANs are extended across the sites using Overlay Transport Virtualization (OTV), or something similar. You need to create filters to prevent traffic that is destined to the local gateway router from being sent over the DCI to the other site. If the gateway router becomes unreachable at one site, you need to remove any filters so traffic can successfully reach the other site's gateway.
- For transparent mode, if the cluster is connected to an HSRP router, you must add the router HSRP MAC address as a static MAC address table entry on the ASA (see Add a Static MAC Address for Bridge Groups, on page 829). When adjacent routers use HSRP, traffic destined to the HSRP IP address will be sent to the HSRP MAC Address, but return traffic will be sourced from the MAC address of a particular router's interface in the HSRP pair. Therefore, the ASA MAC address table is typically only updated when the ASA ARP table entry for the HSRP IP address expires, and the ASA sends an ARP request and receives a reply. Because the ASA's ARP table entries expire after 14400 seconds by default, but the MAC address table entry expires after 300 seconds by default, a static MAC address entry is required to avoid MAC address table expiration traffic drops.
- For routed mode using Spanned EtherChannel, configure site-specific MAC addresses. Extend the data VLANs across the sites using OTV, or something similar. You need to create filters to prevent traffic that is destined to the global MAC address from being sent over the DCI to the other site. If the cluster becomes unreachable at one site, you need to remove any filters so traffic can successfully reach the other site's cluster nodes. Dynamic routing is not supported when an inter-site cluster acts as the first hop router for an extended segment.

Additional Guidelines

- When significant topology changes occur (such as adding or removing an EtherChannel interface, enabling or disabling an interface on the firewall or the switch, adding an additional switch to form a VSS, vPC, StackWise, or StackWise Virtual) you should disable the health check feature and also disable interface monitoring for the disabled interfaces. When the topology change is complete, and the configuration change is synced to all units, you can re-enable the interface health check feature.

- When adding a unit to an existing cluster, or when reloading a unit, there will be a temporary, limited packet/connection drop; this is expected behavior. In some cases, the dropped packets can hang your connection; for example, dropping a FIN/ACK packet for an FTP connection will make the FTP client hang. In this case, you need to reestablish the FTP connection.
- If you use a Windows 2003 server connected to a Spanned EtherChannel, when the syslog server port is down and the server does not throttle ICMP error messages, then large numbers of ICMP messages are sent back to the cluster. These messages can result in some units of the cluster experiencing high CPU, which can affect performance. We recommend that you throttle ICMP error messages.
- It takes time to replicate changes to all the units in a cluster. If you make a large change, for example, adding an access control rule that uses object groups (which, when deployed, are broken out into multiple rules), the time needed to complete the change can exceed the timeout for the cluster units to respond with a success message. If this happens, you might see a "failed to replicate command" message. You can ignore the message.

Defaults for Clustering

- When using Spanned EtherChannels, the cLACP system ID is auto-generated and the system priority is 1 by default.
- The cluster health check feature is enabled by default with the holdtime of 3 seconds. Interface health monitoring is enabled on all interfaces by default.
- The cluster auto-rejoin feature for a failed cluster control link is unlimited attempts every 5 minutes.
- The cluster auto-rejoin feature for a failed data interface is 3 attempts every 5 minutes, with the increasing interval set to 2.
- Connection rebalancing is disabled by default. If you enable connection rebalancing, the default time between load information exchanges is 5 seconds.
- Connection replication delay of 5 seconds is enabled by default for HTTP traffic.

Configure ASA Clustering

To configure clustering, perform the following tasks.



Note To enable or disable clustering, you must use a console connection (for CLI) or an ASDM connection.

Cable the Units and Configure Interfaces

Before configuring clustering, cable the cluster control link network, management network, and data networks. Then configure your interfaces.

Cluster interfaces

You can configure data interfaces as Spanned EtherChannels.

Each unit must also dedicate at least one hardware interface as the cluster control link.

Cluster control link

A cluster control link is a dedicated hardware interface that enables communication between cluster units and supports cluster operation coordination.

Each unit must dedicate at least one hardware interface as the cluster control link. We recommend using an EtherChannel for the cluster control link if available.

Cluster control link traffic

Cluster control link traffic includes both control and data traffic.

Control traffic includes:

- Control node election.
- Configuration replication.
- Health monitoring.

Data traffic includes:

- State replication.
- Connection ownership queries and data packet forwarding.

Cluster Control Link Interfaces and Network

You can use any data interface(s) for the cluster control link, with the following exceptions:

- You cannot use a VLAN subinterface as the cluster control link.
- You cannot use a Management *x*/*x* interface as the cluster control link, either alone or as an EtherChannel.

You can use an EtherChannel.

Each cluster control link has an IP address on the same subnet. This subnet should be isolated from all other traffic, and should include only the ASA cluster control link interfaces.

For a 2-member cluster, do not directly-connect the cluster control link from one ASA to the other ASA. If you directly connect the interfaces, then when one unit fails, the cluster control link fails, and thus the remaining healthy unit fails. If you connect the cluster control link through a switch, then the cluster control link remains up for the healthy unit.

Size the cluster control link

Cluster control link traffic is comprised mainly of state update and forwarded packets. The amount of traffic at any given time on the cluster control link varies. The amount of forwarded traffic depends on the load-balancing efficacy or whether there is a lot of traffic for centralized features.

For example:

- NAT results in poor load balancing of connections, and the need to rebalance all returning traffic to the correct units.
- AAA for network access is a centralized feature, so all traffic is forwarded to the control unit.

- When membership changes, the cluster needs to rebalance a large number of connections, thus temporarily using a large amount of cluster control link bandwidth.

A higher-bandwidth cluster control link helps the cluster to converge faster when there are membership changes and prevents throughput bottlenecks.



Note If your cluster has large amounts of asymmetric (rebalanced) traffic, then you should increase the cluster control link size.

Cluster control link MTU ping testing

Use Cluster control link MTU ping testing to

- verify MTU compatibility between cluster nodes during the join process
- use ping-based fragmentation tests to confirm the network path supports proper packet handling, and
- allow nodes to join even if tests fail while providing diagnostic messages for troubleshooting MTU mismatches.

When a node joins the cluster, the control node send a ping with a payload size of twice the MTU. This process tests the network's ability to handle packet fragmentation because the underlying IP layer will fragment packets that exceed the MTU limit.

A successful ping confirms that the network path supports proper fragmentation and that the cluster control link can reliably process traffic at the configured MTU size.

If the ping fails, view these messages:

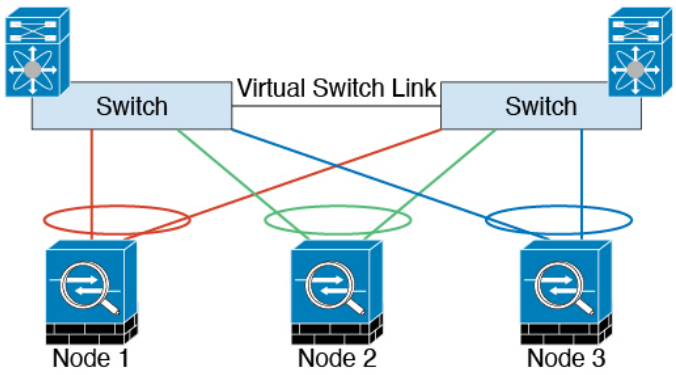
- **show cluster history**—`Event: CCL MTU test to unit name failed`
- Console WARNING—`WARNING: Unit name is not reachable in CCL jumbo frame ICMP test, please check cluster interface and switch MTU configuration`

Even if the ping fails, the node is allowed to join the cluster. In this case, you need to resolve the MTU mismatch as soon as possible.

Cluster control link redundancy

We recommend using an EtherChannel for the cluster control link, so that you can pass traffic on multiple links in the EtherChannel while still achieving redundancy.

The diagram shows how to use an EtherChannel as a cluster control link in a Virtual Switching System (VSS), Virtual Port Channel (vPC), StackWise, or StackWise Virtual environment. All links in the EtherChannel are active. When the switch is part of a redundant system, then you can connect firewall interfaces within the same EtherChannel to separate switches in the redundant system. The switch interfaces are members of the same EtherChannel port-channel interface, because the separate switches act like a single switch. Note that this EtherChannel is device-local, not a Spanned EtherChannel.



Cluster control link reliability

To ensure cluster control link functionality, be sure the round-trip time (RTT) between units is less than 20 ms. This maximum latency enhances compatibility with cluster members installed at different geographical sites. To check your latency, perform a ping on the cluster control link between units.

The cluster control link must be reliable, with no out-of-order or dropped packets; for example, for inter-site deployment, you should use a dedicated link.

Cluster Control Link Failure

If the cluster control link line protocol goes down for a unit, then clustering is disabled; data interfaces are shut down. After you fix the cluster control link, you must manually rejoin the cluster by re-enabling clustering.

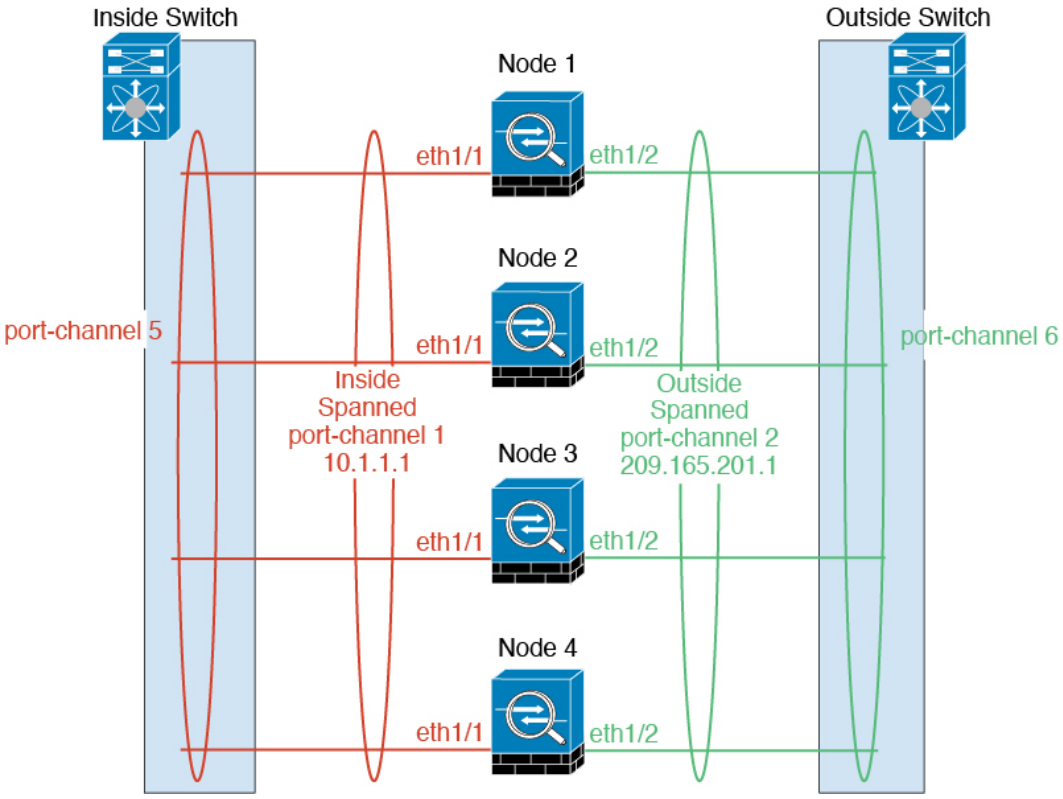
Note When the ASA becomes inactive, all data interfaces are shut down; only the management-only interface can send and receive traffic. The management interface remains up using the IP address the unit received from the cluster IP pool. However if you reload, and the unit is still inactive in the cluster, the management interface is not accessible (because it then uses the Main IP address, which is the same as the control unit). You must use the console port for any further configuration.

Spanned EtherChannels

You can group one or more interfaces per chassis into an EtherChannel that spans all chassis in the cluster. The EtherChannel aggregates the traffic across all the available active interfaces in the channel.

A Spanned EtherChannel can be configured in both routed and transparent firewall modes. In routed mode, the EtherChannel is configured as a routed interface with a single IP address. In transparent mode, the IP address is assigned to the BVI, not to the bridge group member interface.

The EtherChannel inherently provides load balancing as part of basic operation.



Guidelines for maximum throughput

To achieve maximum throughput, follow these guidelines:

- Use a symmetric load-balancing hash algorithm, meaning that packets from both directions will have the same hash and will be sent to the same ASA in the Spanned EtherChannel. We recommend using the source and destination IP address (the default) or the source and destination port as the hashing algorithm.
- Use the same type of line cards when connecting the ASAs to the switch so that hashing algorithms applied to all packets are the same.

Load balancing

The EtherChannel link is selected using a proprietary hash algorithm, based on source or destination IP addresses and TCP and UDP port numbers.

Note On the switch, we recommend that you use one of the following algorithms: **source-dest-IP** or **source-dest-IP-port** (see the Cisco Nexus OS or Cisco IOS **port-channel load-balance** command). Do not use a **vlan** keyword in the load-balance algorithm because it can cause unevenly distributed traffic to the nodes in a cluster.

The number of links in the EtherChannel affects load balancing.

Symmetric load balancing is not always possible. If you configure NAT, then forward and return packets will have different IP addresses and/or ports. Return traffic will be sent to a different unit based on the hash, and the cluster will have to redirect most returning traffic to the correct unit.

EtherChannel redundancy

The EtherChannel has built-in redundancy. It monitors the line protocol status of all links. If one link fails, traffic is re-balanced between remaining links. If all links in the EtherChannel fail on a particular unit, but other units are still active, then the unit is removed from the cluster.

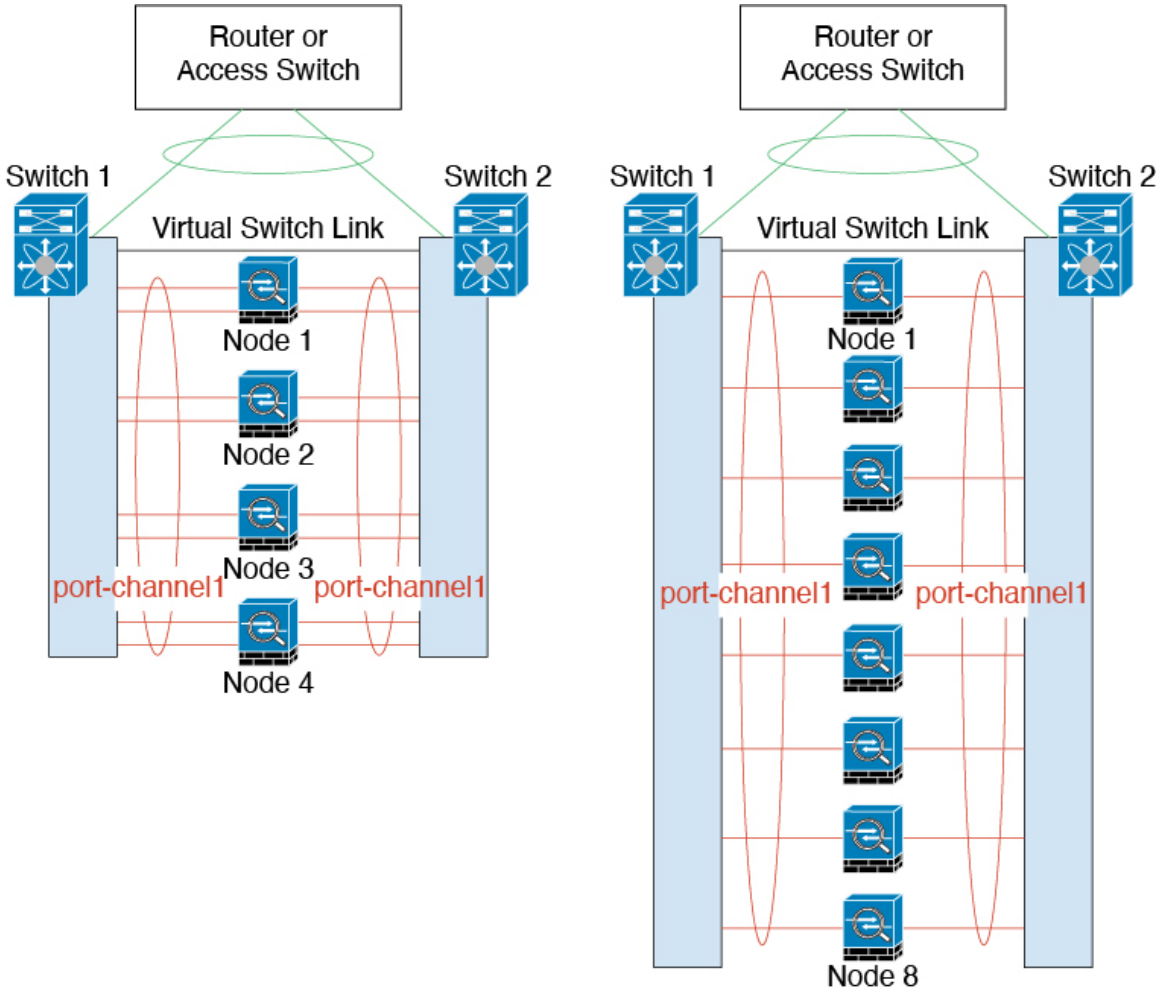
Connecting to a redundant switch system

You can include multiple interfaces per ASA in the Spanned EtherChannel. Multiple interfaces per ASA are especially useful for connecting to both switches in a VSS, vPC, StackWise, or StackWise Virtual system.

Depending on your switches, you can configure up to 32 active links in the spanned EtherChannel. This feature requires both switches in the vPC to support EtherChannels with 16 active links each (for example the Cisco Nexus 7000 with F2-Series 10 Gigabit Ethernet Module).

For switches that support 8 active links in the EtherChannel, you can configure up to 16 active links in the spanned EtherChannel when connecting to two switches in a redundant system.

This figure shows a 16-active-link spanned EtherChannel in a 4-node cluster and an 8-node cluster.



Cable the Cluster Units and Configure Upstream and Downstream Equipment

Before configuring clustering, cable the cluster control link network, management network, and data networks.

Procedure

Step 1 Cable the cluster control link network, management network, and data networks.

Note
At a minimum, an active cluster control link network is required before you configure the nodes to join the cluster.

Step 2 You should also configure the upstream and downstream equipment. For example, if you use EtherChannels, then you should configure the upstream and downstream equipment for the EtherChannels.

Configure the Cluster Interface Mode on Each Unit

Before you enable clustering, you need to convert the firewall to use Spanned EtherChannels. Because clustering limits the types of interfaces you can use, this process lets you check your existing configuration for incompatible interfaces and then prevents you from configuring any unsupported interfaces.

Before you begin

- You must set the mode separately on each ASA that you want to add to the cluster.
- You can always configure the management-only interface as an Individual interface (recommended). The management interface can be an Individual interface even in transparent firewall mode.
- If you configure the management interface as an Individual interface, you cannot enable dynamic routing for the management interface. You must use a static route.

Procedure

Step 1 Show any incompatible configuration so that you can force the interface mode and fix your configuration later; the mode is not changed with this command:

cluster interface-mode spanned check-details

Example:

```
ciscoasa(config)# cluster interface-mode spanned check-details
```

Step 2 Set the interface mode for clustering:

cluster interface-mode spanned force

Example:

```
ciscoasa(config)# cluster interface-mode spanned force
```

There is no default setting; you must explicitly choose the mode. If you have not set the mode, you cannot enable clustering.

The **force** option changes the mode without checking your configuration for incompatible settings. You need to manually fix any configuration issues after you change the mode. Because any interface configuration can only be fixed after you set the mode, we recommend using the **force** option so that you can at least start from the existing configuration. You can re-run the **check-details** option after you set the mode for more guidance.

Without the **force** option, if there is any incompatible configuration, you are prompted to clear your configuration and reload, thus requiring you to connect to the console port to reconfigure your management access. If your configuration is compatible (rare), the mode is changed and the configuration is preserved. If you do not want to clear your configuration, you can exit the command by typing **n**.

To remove the interface mode, enter the **no cluster interface-mode** command.

Configure Interfaces on the Control Node

You must modify any interface that is currently configured with an IP address to be cluster-ready before you enable clustering. For other interfaces, you can configure them before or after you enable clustering; we recommend pre-configuring all of your interfaces so that the complete configuration is synced to new cluster members.

This section describes how to configure interfaces to be compatible with clustering.

Configure the Management Interface as an Individual Interface

Individual interfaces are normal routed interfaces, each with their own IP address taken from a pool of IP addresses. The Main cluster IP address is a fixed address for the cluster that always belongs to the control node.

We recommend configuring the management interface as an Individual interface. Individual management interfaces let you connect directly to each unit if necessary, while a Spanned EtherChannel interface only allows connection to the control node.

Before you begin

- For multiple context mode, perform this procedure in each context. If you are not already in the context configuration mode, enter the **changeto context** *name* command.
- (Optional) Configure the interface as a device-local EtherChannel interface, and/or configure subinterfaces.
  - For an EtherChannel, this EtherChannel is local to the unit, and is not a Spanned EtherChannel.

Procedure

Step 1 Configure a pool of Local IP addresses (IPv4 and/or IPv6), one of which will be assigned to each cluster unit for the interface:

(IPv4)

ip local pool *poolname first-address — last-address* [**mask** *mask*]

(IPv6)

ipv6 local pool *poolname ipv6-address*/*prefix-length number_of_addresses*

Example:

```
ciscoasa(config)# ip local pool ins 192.168.1.2-192.168.1.9
ciscoasa(config-if)# ipv6 local pool insipv6 2001:DB8:45:1002/64 8
```

Include at least as many addresses as there are units in the cluster. If you plan to expand the cluster, include additional addresses. The Main cluster IP address that belongs to the current primary unit is *not* a part of this pool; be sure to reserve an IP address on the same network for the Main cluster IP address.

You cannot determine the exact Local address assigned to each unit in advance; to see the address used on each unit, enter the **show ip**[**v6**] **local pool** *poolname* command. Each cluster member is assigned a member ID when it joins the cluster. The ID determines the Local IP used from the pool.

Step 2 Enter interface configuration mode:

interface *interface_id*

Example:

```
ciscoasa(config)# interface management 1/1
```

Step 3 Set the interface to management-only mode so that it does not pass through traffic:

management-only

By default, Management type interfaces are configured as management-only. In transparent mode, this command is always enabled for a Management type interface.

Step 4 Name the interface:

nameif *name*

Example:

```
ciscoasa(config-if)# nameif management
```

The *name* is a text string up to 48 characters, and is not case-sensitive. You can change the name by reentering this command with a new value.

Step 5 Set the Main cluster IP address and identify the cluster pool:

(IPv4)

ip address *ip_address* [*mask*] **cluster-pool** *poolname*

(IPv6)

ipv6 address *ipv6-address*/*prefix-length* **cluster-pool** *poolname*

Example:

```
ciscoasa(config-if)# ip address 192.168.1.1 255.255.255.0 cluster-pool ins
ciscoasa(config-if)# ipv6 address 2001:DB8:45:1002::99/64 cluster-pool insipv6
```

This IP address must be on the same network as the cluster pool addresses, but not be part of the pool. You can configure an IPv4 and/or an IPv6 address.

DHCP, PPPoE, and IPv6 autoconfiguration are not supported; you must manually configure the IP addresses. Manually configuring the link-local address is also not supported.

Step 6 Set the security level, where *number* is an integer between 0 (lowest) and 100 (highest):

security-level *number*

Example:

```
ciscoasa(config-if)# security-level 100
```

Step 7 Enable the interface:

no shutdown

Examples

The following example configures the Ethernet 1/3 and Ethernet 1/4 interfaces as a device-local EtherChannel, and then configures the EtherChannel as an Individual interface:

```
ip local pool mgmt 10.1.1.2-10.1.1.9
ipv6 local pool mgmtipv6 2001:DB8:45:1002/64 8

interface ethernet 1/3
channel-group 1 mode active
no shutdown

interface ethernet 1/4
channel-group 1 mode active
no shutdown

interface port-channel 1
nameif management
ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
ipv6 address 2001:DB8:45:1002::99/64 cluster-pool mgmtipv6
security-level 100
management-only
```

Configure Spanned EtherChannels

A Spanned EtherChannel spans all ASAs in the cluster, and provides load balancing as part of the EtherChannel operation.

Before you begin

- You must be in Spanned EtherChannel interface mode.
- For multiple context mode, start this procedure in the system execution space. If you are not already in the System configuration mode, enter the **changeto system** command.
- For transparent mode, configure the bridge group. See Configure the Bridge Virtual Interface (BVI), on page 677.
- When using Spanned EtherChannels, the port-channel interface will not come up until clustering is fully enabled. This requirement prevents traffic from being forwarded to a unit that is not an active unit in the cluster.

Procedure

Step 1 Specify the interface you want to add to the channel group:

interface *physical_interface*

Example:

```
ciscoasa(config)# interface ethernet 1/1
```

The *physical_interface* ID includes the type, slot, and port number as type slot/port. This first interface in the channel group determines the type and speed for all other interfaces in the group.

Step 2 Assign this interface to an EtherChannel:

channel-group *channel_id* **mode active**

Example:

```
ciscoasa(config-if)# channel-group 1 mode active
```

The *channel_id* is between 1 and 48. If the port-channel interface for this channel ID does not yet exist in the configuration, one will be added automatically:

interface port-channel *channel_id*

Only **active** mode is supported for Spanned EtherChannels.

Step 3 Enable the interface:

no shutdown

Step 4 (Optional) Add additional interfaces to the EtherChannel by repeating the process.

Example:

```
ciscoasa(config)# interface ethernet 1/2
ciscoasa(config-if)# channel-group 1 mode active
ciscoasa(config-if)# no shutdown
```

Multiple interfaces in the EtherChannel per unit are useful for connecting to switches in a VSS, vPC, StackWise, or StackWise Virtual.

Step 5 Specify the port-channel interface:

interface port-channel *channel_id*

Example:

```
ciscoasa(config)# interface port-channel 1
```

This interface was created automatically when you added an interface to the channel group.

Step 6 (Optional) If you are creating VLAN subinterfaces on this EtherChannel, do so now.

Example:

```
ciscoasa(config)# interface port-channel 1.10
ciscoasa(config-if)# vlan 10
```

The rest of this procedure applies to the subinterfaces.

Step 7 (Multiple Context Mode) Allocate the interface to a context. Then enter:

changeto context *name*
interface port-channel *channel_id*

Example:

```
ciscoasa(config)# context admin
ciscoasa(config)# allocate-interface port-channel1
ciscoasa(config)# changeto context admin
ciscoasa(config-if)# interface port-channel 1
```

For multiple context mode, the rest of the interface configuration occurs within each context.

Step 8 Name the interface:

nameif *name*

Example:

```
ciscoasa(config-if)# nameif inside
```

The *name* is a text string up to 48 characters, and is not case-sensitive. You can change the name by reentering this command with a new value.

Step 9 Perform one of the following, depending on the firewall mode.

- Routed Mode—Set the IPv4 and/or IPv6 address:

  (IPv4)

  ip address *ip_address* [*mask*]

  (IPv6)

  ipv6 address *ipv6-prefix*/*prefix-length*

  Example:

  ```
  ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0
  ciscoasa(config-if)# ipv6 address 2001:DB8::1001/32
  ```

  DHCP, PPPoE, and IPv6 autoconfig are not supported. For point-to-point connections, you can specify a 31-bit subnet mask (255.255.255.254). In this case, no IP addresses are reserved for the network or broadcast addresses. Manually configuring the link-local address is also not supported.

- Transparent Mode—Assign the interface to a bridge group:

  bridge-group *number*

  Example:

  ```
  ciscoasa(config-if)# bridge-group 1
  ```

  Where *number* is an integer between 1 and 100. You can assign up to 64 interfaces to a bridge group. You cannot assign the same interface to more than one bridge group. Note that the BVI configuration includes the IP address.

Step 10 Set the security level:

security-level *number*

Example:

```
ciscoasa(config-if)# security-level 50
```

Where *number* is an integer between 0 (lowest) and 100 (highest).

Step 11 Configure a unique, manual global MAC address for a Spanned EtherChannel to avoid potential network connectivity problems:

mac-address *mac_address*

Example:

```
ciscoasa(config-if)# mac-address 000C.F142.4CDE
```

You must configure a unique MAC address not currently in use on your network. With a manually-configured MAC address, the MAC address stays with the current control unit. If you do not configure a MAC address, then if the control unit changes, the new control unit uses a new MAC address for the interface, which can cause a temporary network outage.

In multiple context mode, if you share an interface between contexts, you should instead enable auto-generation of MAC addresses so you do not need to set the MAC address manually. Note that you must manually configure the MAC address using this command for *non-shared* interfaces.

The *mac_address* is in H.H.H format, where H is a 16-bit hexadecimal digit. For example, the MAC address 00-0C-F1-42-4C-DE is entered as 000C.F142.4CDE.

The first two bytes of a manual MAC address cannot be A2 if you also want to use auto-generated MAC addresses.

Step 12 (Routed mode) For inter-site clustering, configure a site-specific MAC address and IP address for each site:

mac-address *mac_address* **site-id** *number* **site-ip** *ip_address*

Example:

```
ciscoasa(config-if)# mac-address aaaa.1111.1234
ciscoasa(config-if)# mac-address aaaa.1111.aaaa site-id 1 site-ip 10.9.9.1
ciscoasa(config-if)# mac-address aaaa.1111.bbbb site-id 2 site-ip 10.9.9.2
ciscoasa(config-if)# mac-address aaaa.1111.cccc site-id 3 site-ip 10.9.9.3
ciscoasa(config-if)# mac-address aaaa.1111.dddd site-id 4 site-ip 10.9.9.4
```

The site-specific IP addresses must be on the same subnet as the global IP address. The site-specific MAC address and IP address used by a unit depends on the site ID you specify in each unit's bootstrap configuration.

Create the Bootstrap Configuration

Each node in the cluster requires a bootstrap configuration to join the cluster.

Configure the Control Node Bootstrap Settings

Each node in the cluster requires a bootstrap configuration to join the cluster. Typically, the first node you configure to join the cluster will be the control node. After you enable clustering, after an election period, the cluster elects a control node. With only one node in the cluster initially, that node will become the control node. Subsequent nodes that you add to the cluster will be data nodes.

Before you begin

- Back up your configurations in case you later want to leave the cluster, and need to restore your configuration.
- For multiple context mode, complete these procedures in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.
- You must use the console port to enable or disable clustering. You cannot use Telnet or SSH.
- When you add a node to a running cluster, you may see temporary, limited packet/connection drops; this is expected behavior.
- Pre-determine the size of the cluster control link. See Size the cluster control link, on page 346.

Procedure

Step 1 Enable the cluster control link interface before you join the cluster.

You will later identify this interface as the cluster control link when you enable clustering.

We recommend that you combine multiple cluster control link interfaces into an EtherChannel if you have enough interfaces. The EtherChannel is local to the ASA, and is not a Spanned EtherChannel.

The cluster control link interface configuration is not replicated from the control node to data nodes; however, you must use the same configuration on each node. Because this configuration is not replicated, you must configure the cluster control link interfaces separately on each node.

- You cannot use a VLAN subinterface as the cluster control link.
- You cannot use a Management *x*/*x* interface as the cluster control link, either alone or as an EtherChannel.

a) Enter interface configuration mode:

interface *interface_id*

Example:

```
ciscoasa(config)# interface ethernet 1/6
```

b) (Optional, for an EtherChannel) Assign this physical interface to an EtherChannel:

channel-group *channel_id* **mode on**

Example:

```
ciscoasa(config-if)# channel-group 1 mode on
```

The *channel_id* is between 1 and 48. If the port-channel interface for this channel ID does not yet exist in the configuration, one will be added automatically:

interface port-channel *channel_id*

We recommend using the On mode for cluster control link member interfaces to reduce unnecessary traffic on the cluster control link. The cluster control link does not need the overhead of LACP traffic because it is an isolated, stable network. **Note:** We recommend setting *data* EtherChannels to Active mode.

c) Enable the interface:

no shutdown

You only need to enable the interface; do not configure a name for the interface, or any other parameters.

d) (For an EtherChannel) Repeat for each additional interface you want to add to the EtherChannel:

Example:

```
ciscoasa(config)# interface ethernet 1/7
ciscoasa(config-if)# channel-group 1 mode on
ciscoasa(config-if)# no shutdown
```

Step 2 Specify the maximum transmission node for the cluster control link interface to be at least 100 bytes higher than the highest MTU of the data interfaces.

mtu cluster *bytes*

Example:

```
ciscoasa(config)# mtu cluster 9198
```

Set the MTU between 1400 and 9198 bytes, but not between 2561 and 8362. Due to block pool handling, this MTU size is not optimal for system operation. The default MTU is 1500 bytes. We suggest setting the cluster control link MTU to the maximum. Because the cluster control link traffic includes data packet forwarding, the cluster control link needs to accommodate the entire size of a data packet plus cluster traffic overhead.

For example, because the maximum MTU is 9198 bytes, then the highest data interface MTU can be 9098, while the cluster control link can be set to 9198.

This command is a global configuration command, but is also part of the bootstrap configuration that is not replicated between nodes.

Step 3 Name the cluster and enter cluster configuration mode:

cluster group *name*

Example:

```
ciscoasa(config)# cluster group pod1
```

The name must be an ASCII string from 1 to 38 characters. You can only configure one cluster group per node. All members of the cluster must use the same name.

Step 4 Name this member of the cluster:

local-unit *unit_name*

Use a unique ASCII string from 1 to 38 characters. Each node must have a unique name. A node with a duplicated name will be not be allowed in the cluster.

Example:

```
ciscoasa(cfg-cluster)# local-unit node1
```

Step 5 Specify the cluster control link interface, preferably an EtherChannel:

cluster-interface *interface_id* **ip** *ip_address mask*

Example:

```
ciscoasa(cfg-cluster)# cluster-interface port-channel2 ip 192.168.1.1 255.255.255.0
INFO: Non-cluster interface config is cleared on Port-Channel2
```

Subinterfaces and Management interfaces are not allowed.

Specify an IPv4 address for the IP address; IPv6 is not supported for this interface. This interface cannot have a **nameif** configured.

For each node, specify a different IP address on the same network.

Step 6 If you use inter-site clustering, set the site ID for this node so it uses a site-specific MAC address:

site-id *number*

Example:

```
ciscoasa(cfg-cluster)# site-id 1
```

The *number* is between 1 and 8.

Step 7 Set the priority of this node for control node elections:

priority *priority_number*

Example:

```
ciscoasa(cfg-cluster)# priority 1
```

The priority is between 1 and 100, where 1 is the highest priority.

Step 8 (Optional) Set an authentication key for control traffic on the cluster control link:

key *shared_secret*

Example:

```
ciscoasa(cfg-cluster)# key chuntheunavoidable
```

The shared secret is an ASCII string from 1 to 63 characters. The shared secret is used to generate the key. This command does not affect datapath traffic, including connection state update and forwarded packets, which are always sent in the clear.

Step 9 (Optional) Manually specify the cLACP system ID and system priority:

clacp system-mac {*mac_address* | **auto**} [**system-priority** *number*]

Example:

```
ciscoasa(cfg-cluster)# clacp system-mac 000a.0000.aaaa
```

When using Spanned EtherChannels, the ASA uses cLACP to negotiate the EtherChannel with the neighbor switch. ASAs in a cluster collaborate in cLACP negotiation so that they appear as a single (virtual) device to the switch. One parameter in cLACP negotiation is a system ID, which is in the format of a MAC address. All ASAs in the cluster use the same system ID: auto-generated by the control node (the default) and replicated to all secondaries; or manually specified in this command in the form *H.H.H*, where H is a 16-bit hexadecimal digit. (For example, the MAC address 00-0A-00-00-AA-AA is entered as 000A.0000.AAAA.) You might want to manually configure the MAC address for troubleshooting purposes, for example, so that you can use an easily identified MAC address. Typically, you would use the auto-generated MAC address.

The system priority, between 1 and 65535, is used to decide which node is in charge of making a bundling decision. By default, the ASA uses priority 1, which is the highest priority. The priority needs to be higher than the priority on the switch.

This command is not part of the bootstrap configuration, and is replicated from the control node to the data nodes. However, you cannot change this value after you enable clustering.

Step 10 Enable clustering:

enable [**noconfirm**]

Example:

```
ciscoasa(cfg-cluster)# enable
INFO: Clustering is not compatible with following commands:
policy-map global_policy
 class inspection_default
 inspect skinny
policy-map global_policy
 class inspection_default
  inspect sip
 Would you like to remove these commands? [Y]es/[N]o:Y

INFO: Removing incompatible commands from running configuration...
Cryptochecksum (changed): f16b7fc2 a742727e e40bc0b0 cd169999
INFO: Done
```

When you enter the **enable** command, the ASA scans the running configuration for incompatible commands for features that are not supported with clustering, including commands that may be present in the default configuration. You are prompted to delete the incompatible commands. If you respond **No**, then clustering is not enabled. Use the **noconfirm** keyword to bypass the confirmation and delete incompatible commands automatically.

For the first node enabled, a control node election occurs. Because the first node should be the only member of the cluster so far, it will become the control node. Do not perform any configuration changes during this period.

To disable clustering, enter the **no enable** command.

Note

If you disable clustering, all data interfaces are shut down, and only the management-only interface is active.

Examples

The following example configures a management interface, configures a device-local EtherChannel for the cluster control link, and then enables clustering for the ASA called "node1," which will become the control node because it is added to the cluster first:

```
ip local pool mgmt 10.1.1.2-10.1.1.9
ipv6 local pool mgmtipv6 2001:DB8::1002/32 8
interface management 1/1
   nameif management
   ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
   ipv6 address 2001:DB8::1001/32 cluster-pool mgmtipv6
   security-level 100
   management-only
   no shutdown

interface ethernet 1/6
   channel-group 1 mode on
   no shutdown

interface ethernet 1/7
   channel-group 1 mode on
   no shutdown

cluster group pod1
   local-unit node1
   cluster-interface port-channel1 ip 192.168.1.1 255.255.255.0
   priority 1
   key chuntheunavoidable
   enable noconfirm
```

Configure Data Node Bootstrap Settings

Perform the following procedure to configure the data nodes.

Before you begin

- You must use the console port to enable or disable clustering. You cannot use Telnet or SSH.
- Back up your configurations in case you later want to leave the cluster, and need to restore your configuration.
- For multiple context mode, complete this procedure in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.
- If you have any interfaces in your configuration that have not been configured for clustering (for example, the default configuration Management 1/1 interface), you can join the cluster as a data node (with no possibility of becoming the control node in a current election).
- When you add a node to a running cluster, you may see temporary, limited packet/connection drops; this is expected behavior.

Procedure

Step 1 Configure the same cluster control link interface as you configured for the control node.

Example:

```
ciscoasa(config)# interface ethernet 1/6
ciscoasa(config-if)# channel-group 1 mode on
ciscoasa(config-if)# no shutdown
ciscoasa(config)# interface ethernet 1/7
ciscoasa(config-if)# channel-group 1 mode on
ciscoasa(config-if)# no shutdown
```

Step 2 Specify the same MTU that you configured for the control node:

Example:

```
ciscoasa(config)# mtu cluster 9198
```

Step 3 Identify the same cluster name that you configured for the control node:

Example:

```
ciscoasa(config)# cluster group pod1
```

Step 4 Name this member of the cluster with a unique string:

local-unit *unit_name*

Example:

```
ciscoasa(cfg-cluster)# local-unit node2
```

Specify an ASCII string from 1 to 38 characters.

Each node must have a unique name. A node with a duplicated name will be not be allowed in the cluster.

Step 5 Specify the same cluster control link interface that you configured for the control node, but specify a different IP address on the same network for each node:

cluster-interface *interface_id* **ip** *ip_address mask*

Example:

```
ciscoasa(cfg-cluster)# cluster-interface port-channel2 ip 192.168.1.2 255.255.255.0
INFO: Non-cluster interface config is cleared on Port-Channel2
```

Specify an IPv4 address for the IP address; IPv6 is not supported for this interface. This interface cannot have a **nameif** configured.

Step 6 If you use inter-site clustering, set the site ID for this node so it uses a site-specific MAC address:

site-id *number*

Example:

```
ciscoasa(cfg-cluster)# site-id 1
```

The **number** is between 1 and 8.

Step 7 Set the priority of this node for control node elections, typically to a higher value than the control node:

priority *priority_number*

Example:

```
ciscoasa(cfg-cluster)# priority 2
```

Set the priority between 1 and 100, where 1 is the highest priority.

Step 8 Set the same authentication key that you set for the control node:

Example:

```
ciscoasa(cfg-cluster)# key chuntheunavoidable
```

Step 9 Enable clustering:

enable as-slave

You can avoid any configuration incompatibilities (primarily the existence of any interfaces not yet configured for clustering) by using the **enable as-slave** command. This command ensures the data node joins the cluster with no possibility of becoming the control node in any current election. Its configuration is overwritten with the one synced from the control node.

To disable clustering, enter the **no enable** command.

Note
If you disable clustering, all data interfaces are shut down, and only the management interface is active.

Examples

The following example includes the configuration for a data node, node2:

```
interface ethernet 1/6

channel-group 1 mode on
no shutdown

interface ethernet 1/7

channel-group 1 mode on
no shutdown

cluster group pod1

local-unit node2
cluster-interface port-channel1 ip 192.168.1.2 255.255.255.0
```

```
priority 2
key chuntheunavoidable
enable as-slave
```

Customize the Clustering Operation

You can customize clustering health monitoring, TCP connection replication delay, flow mobility and other optimizations.

Perform these procedures on the control node.

Configure Basic ASA Cluster Parameters

You can customize cluster settings on the control node.

Before you begin

- For multiple context mode, complete this procedure in the system execution space on the control node. To change from the context to the system execution space, enter the **changeto system** command.

Procedure

Step 1 Enter cluster configuration mode:

cluster group *name*

Step 2 (Optional) Enable console replication from data nodes to the control node:

console-replicate

This feature is disabled by default. The ASA prints out some messages directly to the console for certain critical events. If you enable console replication, data nodes send the console messages to the control node so that you only need to monitor one console port for the cluster.

Step 3 Set the minimum trace level for clustering events:

trace-level *level*

Set the minimum level as desired:

- **critical**—Critical events (severity=1)
- **warning**—Warnings (severity=2)
- **informational**—Informational events (severity=3)
- **debug**—Debugging events (severity=4)

Configure Health Monitoring and Auto-Rejoin Settings

This procedure configures node and interface health monitoring.

You might want to disable health monitoring of non-essential interfaces, for example, the management interface. You can monitor any port-channel ID, redundant ID, or single physical interface ID. Health monitoring is not performed on VLAN subinterfaces or virtual interfaces such as VNIs or BVIs. You cannot configure monitoring for the cluster control link; it is always monitored.

Procedure

Step 1 Enter cluster configuration mode.

cluster group *name*

Example:

```
ciscoasa(config)# cluster group test
ciscoasa(cfg-cluster)#
```

Step 2 Customize the cluster node health check feature.

health-check [**holdtime** *timeout*] [**vss-enabled**]

To determine node health, the ASA cluster nodes send heartbeat messages on the cluster control link to other nodes. If a node does not receive any heartbeat messages from a peer node within the holdtime period, the peer node is considered unresponsive or dead.

- **holdtime** *timeout*—Determines the amount of time between node heartbeat status messages, between .3 and 45 seconds; The default is 3 seconds.
- **vss-enabled**—Floods the heartbeat messages on all EtherChannel interfaces in the cluster control link to ensure that at least one of the switches can receive them. If you configure the cluster control link as an EtherChannel (recommended), and it is connected to a VSS, vPC, StackWise, or StackWise Virtual pair, then you might need to enable the **vss-enabled** option. For some switches, when one node in the redundant system is shutting down or booting up, EtherChannel member interfaces connected to that switch may appear to be Up to the ASA, but they are not passing traffic on the switch side. The ASA can be erroneously removed from the cluster if you set the ASA holdtime timeout to a low value (such as .8 seconds), and the ASA sends keepalive messages on one of these EtherChannel interfaces.

When any topology changes occur (such as adding or removing a data interface, enabling or disabling an interface on the ASA or the switch, or adding an additional switch to form a VSS, vPC, StackWise, or StackWise Virtual) you should disable the health check feature and also disable interface monitoring for the disabled interfaces (**no health-check monitor-interface**). When the topology change is complete, and the configuration change is synced to all nodes, you can re-enable the health check feature.

Example:

```
ciscoasa(cfg-cluster)# health-check holdtime 5
```

Step 3 Disable the interface health check on an interface.

no health-check monitor-interface *interface_id*

The interface health check monitors for link failures. If all physical ports for a given logical interface fail on a particular node, but there are active ports under the same logical interface on other nodes, then the node is removed from the cluster. The amount of time before the ASA removes a member from the cluster depends

on the type of interface and whether the node is an established member or is joining the cluster. Health check is enabled by default for all interfaces. You can disable it per interface using the **no** form of this command. You might want to disable health monitoring of non-essential interfaces, for example, the management interface.

- *interface_id*—Disables monitoring of any port-channel ID, redundant ID, or single physical interface ID. Health monitoring is not performed on VLAN subinterfaces or virtual interfaces such as VNIs or BVIs. You cannot configure monitoring for the cluster control link; it is always monitored.

When any topology changes occur (such as adding or removing a data interface, enabling or disabling an interface on the ASA or the switch, or adding an additional switch to form a VSS, vPC, StackWise, or StackWise Virtual) you should disable the health check feature (**no health-check**) and also disable interface monitoring for the disabled interfaces. When the topology change is complete, and the configuration change is synced to all nodes, you can re-enable the health check feature.

Example:

```
ciscoasa(cfg-cluster)# no health-check monitor-interface management1/1
```

Step 4 Customize the auto-rejoin cluster settings after a health check failure.

health-check {**data-interface** | **cluster-interface** | **system**} **auto-rejoin** [**unlimited** | *auto_rejoin_max*] *auto_rejoin_interval auto_rejoin_interval_variation*

- **system**—Specifies the auto-rejoin settings for internal errors. Internal failures include: application sync timeout; inconsistent application statuses; and so on.
- **unlimited**—(Default for the **cluster-interface**) Does not limit the number of rejoin attempts.
- *auto-rejoin-max*—Sets the number of rejoin attempts, between 0 and 65535. **0** disables auto-rejoining. The default for the **data-interface** and **system** is 3.
- *auto_rejoin_interval*—Defines the interval duration in minutes between rejoin attempts, between 2 and 60. The default value is 5 minutes. The maximum total time that the node attempts to rejoin the cluster is limited to 14400 minutes (10 days) from the time of last failure.
- *auto_rejoin_interval_variation*—Defines if the interval duration increases. Set the value between 1 and 3: **1** (no change); **2** (2 x the previous duration), or **3** (3 x the previous duration). For example, if you set the interval duration to 5 minutes, and set the variation to 2, then the first attempt is after 5 minutes; the 2nd attempt is 10 minutes (2 x 5); the 3rd attempt 20 minutes (2 x 10), and so on. The default value is **1** for the cluster-interface and **2** for the data-interface and system.

Example:

```
ciscoasa(cfg-cluster)# health-check data-interface auto-rejoin 10 3 3
```

Step 5 Configure the debounce time before the ASA considers an interface to be failed and the node is removed from the cluster.

health-check monitor-interface debounce-time *ms*

Example:

```
ciscoasa(cfg-cluster)# health-check monitor-interface debounce-time 300
```

Set the debounce time between 300 and 9000 ms. The default is 500 ms. Lower values allow for faster detection of interface failures. Note that configuring a lower debounce time increases the chances of false-positives. When an interface status update occurs, the ASA waits the number of milliseconds specified before marking the interface as failed and the node is removed from the cluster. In the case of an EtherChannel that transitions from a down state to an up state (for example, the switch reloaded, or the switch enabled an EtherChannel), a longer debounce time can prevent the interface from appearing to be failed on a cluster node just because another cluster node was faster at bundling the ports.

Step 6 (Optional) Configure traffic load monitoring.

load-monitor [**frequency** *seconds*] [**intervals** *intervals*]

- **frequency** *seconds*—Sets the time in seconds between monitoring messages, between 10 and 360 seconds. The default is 20 seconds.
- **intervals** *intervals*—Sets the number of intervals for which the ASA maintains data, between 1 and 60. The default is 30.

You can monitor the traffic load for cluster members, including total connection count, CPU and memory usage, and buffer drops. If the load is too high, you can choose to manually disable clustering on the node if the remaining nodes can handle the load, or adjust the load balancing on the external switch. This feature is enabled by default. For example, for inter-chassis clustering on the Firepower 9300 with 3 security modules in each chassis, if 2 security modules in a chassis leave the cluster, then the same amount of traffic to the chassis will be sent to the remaining module and potentially overwhelm it. You can periodically monitor the traffic load. If the load is too high, you can choose to manually disable clustering on the node.

Use the **show cluster info load-monitor** command to view the traffic load.

Example:

```
ciscoasa(cfg-cluster)# load-monitor frequency 50 intervals 25
ciscoasa(cfg-cluster)# show cluster info load-monitor
ID  Unit Name
0  B
1  A_1
Information from all units with 50 second interval:
Unit     Connections    Buffer Drops   Memory Used   CPU Used
Average from last 1 interval:
   0           0            0            14              25
   1           0            0            16              20
Average from last 25 interval:
   0           0            0            12              28
   1           0            0            13              27
```

Example

The following example configures the health-check holdtime to .3 seconds; enables VSS; disables monitoring on the Ethernet 1/2 interface, which is used for management; sets the auto-rejoin for data interfaces to 4 attempts starting at 2 minutes, increasing the duration by 3 x the previous interval; and sets the auto-rejoin for the cluster control link to 6 attempts every 2 minutes.

```
ciscoasa(config)# cluster group test
ciscoasa(cfg-cluster)# health-check holdtime .3 vss-enabled
ciscoasa(cfg-cluster)# no health-check monitor-interface ethernet1/2
```

```
ciscoasa(cfg-cluster)# health-check data-interface auto-rejoin 4 2 3
ciscoasa(cfg-cluster)# health-check cluster-interface auto-rejoin 6 2 1
```

Configure Connection Rebalancing and the Cluster TCP Replication Delay

You can configure connection rebalancing. For more information, see Rebalancing New TCP Connections Across the Cluster, on page 421

Enable the cluster replication delay for TCP connections to help eliminate the “unnecessary work” related to short-lived flows by delaying the director/backup flow creation. Note that if a node fails before the director/backup flow is created, then those flows cannot be recovered. Similarly, if traffic is rebalanced to a different node before the flow is created, then the flow cannot be recovered. You should not enable the TCP replication delay for traffic on which you disable TCP randomization.

Procedure

Step 1 Enable the cluster replication delay for TCP connections:

cluster replication delay *seconds* {**http | match tcp {host** *ip_address* | *ip_address mask* | **any | any4 | any6} [{eq | lt | gt}** *port*] {**host** *ip_address* | *ip_address mask* | **any | any4 | any6} [{eq | lt | gt}** *port*]}

Example:

```
ciscoasa(config)# cluster replication delay 15 match tcp any any eq ftp
ciscoasa(config)# cluster replication delay 15 http
```

Set the *seconds* between 1 and 15. The **http** delay is enabled by default for 5 seconds.

In multiple context mode, configure this setting within the context.

Step 2 Enter cluster configuration mode:

cluster group *name*

Step 3 (Optional) Enable connection rebalancing for TCP traffic:

conn-rebalance [**frequency** *seconds*]

Example:

```
ciscoasa(cfg-cluster)# conn-rebalance frequency 60
```

This command is disabled by default. If enabled, ASAs exchange information about the connections per second periodically, and offload new connections from devices with more connections per second to less loaded devices. Existing connections are never moved. Moreover, because this command only rebalances based on connections per second, the total number of established connections on each node is not considered, and the total number of connections may not be equal. The frequency, between 1 and 360 seconds, specifies how often the load information is exchanged. The default is 5 seconds.

Once a connection is offloaded to a different node, it becomes an asymmetric connection.

Do not configure connection rebalancing for inter-site topologies; you do not want connections rebalanced to cluster members at a different site.

Configure Inter-Site Features

For inter-site clustering, you can customize your configuration to enhance redundancy and stability.

Enable Director Localization

To improve performance and reduce round-trip time (RTT) for inter-site clustering, you can enable director localization. Director localization adds two director roles: a local director and a global director.

When a cluster member receives a new-flow packet and localization applies, the receiving member uses its own site ID to select a local director from members at the same site. The global director can be at any site. Director localization optimizes same-site flow setup for common cases, but it does not guarantee that every director query or owner selection remains at the ingress site.

The cluster uses the global director instead of the local director in these conditions:

- The local-director hash and global-director hash resolve to the same member.
- The receiving member is selected as the local director.
- The flow has a NAT or PAT translation.
- The flow is SCTP traffic.
- The flow is a TCP-proxy server-certificate probe connection.
- The flow is a reassembled fragment chain.
- The local data interface is not ready.



Note Cross-site director-query output in **packet-tracer** does not by itself indicate a cluster failure. If packets are dropped, verify routing on each cluster member. Cluster members run independent routing processes, and routing tables are not synchronized over the cluster control link.

Before you begin

- Set the site ID for the cluster member in the bootstrap configuration.
- The following traffic types do not support localization: NAT or PAT traffic; SCTP-inspected traffic; Fragmentation owner query.

Procedure

Step 1 Enter cluster configuration mode.

cluster group *name*

Example:

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)#
```

Step 2 Enable director localization.

director-localization

Enable Site Redundancy

To protect flows from a site failure, you can enable site redundancy. If the connection backup owner is at the same site as the owner, then an additional backup owner will be chosen from another site to protect flows from a site failure.

Before you begin

- Set the site ID for the cluster member in the bootstrap configuration.

Procedure

Step 1 Enter cluster configuration mode.

cluster group *name*

Example:

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)#
```

Step 2 Enable site redundancy.

site-redundancy

Configure Per-Site Gratuitous ARP

The ASA now generates gratuitous ARP (GARP) packets to keep the switching infrastructure up to date: the highest priority member at each site periodically generates GARP traffic for the global MAC/IP addresses.

When using per-site MAC and IP addresses, packets sourced from the cluster use a site-specific MAC address and IP address, while packets received by the cluster use a global MAC address and IP address. If traffic is not generated from the global MAC address periodically, you could experience a MAC address timeout on your switches for the global MAC address. After a timeout, traffic destined for the global MAC address will be flooded across the entire switching infrastructure, which can cause performance and security concerns.

GARP is enabled by default when you set the site ID for each unit and the site MAC address for each Spanned EtherChannel. You can customize the GARP interval, or you can disable GARP.

Before you begin

- Set the site ID for the cluster member in the bootstrap configuration.
- Set the per-site MAC address for the Spanned EtherChannel in the control unit configuration.

Procedure

Step 1 Enter cluster configuration mode.

cluster group *name*

Example:

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)#
```

Step 2 Customize the GARP interval.

site-periodic-garp interval *seconds*

- *seconds*—Sets the time in seconds between GARP generation, between 1 and 1000000 seconds. The default is 290 seconds.

To disable GARP, enter **no site-periodic-garp interval**.

Configure Cluster Flow Mobility

You can inspect LISP traffic to enable flow mobility when a server moves between sites.

About LISP Inspection

You can inspect LISP traffic to enable flow mobility between sites.

About LISP

Data center virtual machine mobility such as VMware VMotion enables servers to migrate between data centers while maintaining connections to clients. To support such data center server mobility, routers need to be able to update the ingress route towards the server when it moves. Cisco Locator/ID Separation Protocol (LISP) architecture separates the device identity, or endpoint identifier (EID), from its location, or routing locator (RLOC), into two different numbering spaces, making server migration transparent to clients. For example, when a server moves to a new site and a client sends traffic to the server, the router redirects traffic to the new location.

LISP requires routers and servers in certain roles, such as the LISP egress tunnel router (ETR), ingress tunnel router (ITR), first hop routers, map resolver (MR), and map server (MS). When the first hop router for the server senses that the server is connected to a different router, it updates all of the other routers and databases so that the ITR connected to the client can intercept, encapsulate, and send traffic to the new server location.

ASA LISP Support

The ASA does not run LISP itself; it can, however, inspect LISP traffic for location changes and then use this information for seamless clustering operation. Without LISP integration, when a server moves to a new site, traffic comes to an ASA cluster member at the new site instead of to the original flow owner. The new ASA forwards traffic to the ASA at the old site, and then the old ASA has to send traffic back to the new site to reach the server. This traffic flow is sub-optimal and is known as "tromboning" or "hair-pinning."

With LISP integration, the ASA cluster members can inspect LISP traffic passing between the first hop router and the ETR or ITR, and can then change the flow owner to be at the new site.

LISP Guidelines

- The ASA cluster members must reside between the first hop router and the ITR or ETR for the site. The ASA cluster itself cannot be the first hop router for an extended segment.
- Only fully-distributed flows are supported; centralized flows, semi-distributed flows, or flows belonging to individual nodes are not moved to new owners. Semi-distributed flows include applications, such as SIP, where all child flows are owned by the same ASA that owns the parent flow.
- The cluster only moves Layer 3 and 4 flow states; some application data might be lost.
- For short-lived flows or non-business-critical flows, moving the owner may not be worthwhile. You can control the types of traffic that are supported with this feature when you configure the inspection policy, and should limit flow mobility to essential traffic.

ASA LISP Implementation

This feature includes several inter-related configurations (all of which are described in this chapter):

1. (Optional) Limit inspected EIDs based on the host or server IP address—The first hop router might send EID-notify messages for hosts or networks the ASA cluster is not involved with, so you can limit the EIDs to only those servers or networks relevant to your cluster. For example, if the cluster is only involved with 2 sites, but LISP is running on 3 sites, you should only include EIDs for the 2 sites involved with the cluster.
2. LISP traffic inspection—The ASA inspects LISP traffic on UDP port 4342 for the EID-notify message sent between the first hop router and the ITR or ETR. The ASA maintains an EID table that correlates the EID and the site ID. For example, you should inspect LISP traffic with a source IP address of the first hop router and a destination address of the ITR or ETR. Note that LISP traffic is not assigned a director, and LISP traffic itself does not participate in cluster state sharing.
3. Service Policy to enable flow mobility on specified traffic—You should enable flow mobility on business-critical traffic. For example, you can limit flow mobility to only HTTPS traffic, and/or to traffic to specific servers.
4. Site IDs—The ASA uses the site ID for each cluster node to determine the new owner.
5. Cluster-level configuration to enable flow mobility—You must also enable flow mobility at the cluster level. This on/off toggle lets you easily enable or disable flow mobility for a particular class of traffic or applications.

Configure LISP Inspection

You can inspect LISP traffic to enable flow mobility when a server moves between sites.

Before you begin

- Assign each cluster unit to a site ID according to Configure the Control Node Bootstrap Settings, on page 359 and Configure Data Node Bootstrap Settings, on page 363.
- LISP traffic is not included in the default-inspection-traffic class, so you must configure a separate class for LISP traffic as part of this procedure.

Procedure

Step 1 (Optional) Configure a LISP inspection map to limit inspected EIDs based on IP address, and to configure the LISP pre-shared key:

a) Create an extended ACL; only the destination IP address is matched to the EID embedded address:

access list *eid_acl_name* **extended permit ip** *source_address mask destination_address mask*

Both IPv4 and IPv6 ACLs are accepted. See the command reference for exact **access-list extended** syntax.

b) Create the LISP inspection map, and enter parameters mode:

policy-map type inspect lisp *inspect_map_name*

parameters

c) Define the allowed EIDs by identifying the ACL you created:

allowed-eid access-list *eid_acl_name*

The first hop router or ITR/ETR might send EID-notify messages for hosts or networks that the ASA cluster is not involved with, so you can limit the EIDs to only those servers or networks relevant to your cluster. For example, if the cluster is only involved with 2 sites, but LISP is running on 3 sites, you should only include EIDs for the 2 sites involved with the cluster.

d) If necessary, enter the pre-shared key:

validate-key *key*

Example:

```
ciscoasa(config)# access-list TRACKED_EID_LISP extended permit ip any 10.10.10.0 255.255.255.0
ciscoasa(config)# policy-map type inspect lisp LISP_EID_INSPECT
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# allowed-eid access-list TRACKED_EID_LISP
ciscoasa(config-pmap-p)# validate-key MadMaxShinyandChrome
```

Step 2 Configure LISP inspection for UDP traffic between the first hop router and the ITR or ETR on port 4342:

a) Configure the extended ACL to identify LISP traffic:

access list *inspect_acl_name* **extended permit udp** *source_address mask destination_address mask* **eq 4342**

You *must* specify UDP port 4342. Both IPv4 and IPv6 ACLs are accepted. See the command reference for exact **access-list extended** syntax.

b) Create a class map for the ACL:

class-map *inspect_class_name*

match access-list *inspect_acl_name*

c) Specify the policy map, the class map, enable inspection using the optional LISP inspection map, and apply the service policy to an interface (if new):

policy-map *policy_map_name*

class *inspect_class_name*

inspect lisp [*inspect_map_name*]

service-policy *policy_map_name* {**global** | **interface** *ifc_name*}

If you have an existing service policy, specify the existing policy map name. By default, the ASA includes a global policy called **global_policy**, so for a global policy, specify that name. You can also create one service policy per interface if you do not want to apply the policy globally. LISP inspection is applied to traffic bidirectionally so you do not need to apply the service policy on both the source and destination interfaces; all traffic that enters or exits the interface to which you apply the policy map is affected if the traffic matches the class map for both directions.

Example:

```
ciscoasa(config)# access-list LISP_ACL extended permit udp host 192.168.50.89 host
192.168.10.8 eq 4342
ciscoasa(config)# class-map LISP_CLASS
ciscoasa(config-cmap)# match access-list LISP_ACL
ciscoasa(config-cmap)# policy-map INSIDE_POLICY
ciscoasa(config-pmap)# class LISP_CLASS
ciscoasa(config-pmap-c)# inspect lisp LISP_EID_INSPECT
ciscoasa(config)# service-policy INSIDE_POLICY interface inside
```

The ASA inspects LISP traffic for the EID-notify message sent between the first hop router and the ITR or ETR. The ASA maintains an EID table that correlates the EID and the site ID.

Step 3 Enable Flow Mobility for a traffic class:

a) Configure the extended ACL to identify business critical traffic that you want to re-assign to the most optimal site when servers change sites:

access list *flow_acl_name* **extended permit udp** *source_address mask destination_address mask* **eq** *port*

Both IPv4 and IPv6 ACLs are accepted. See the command reference for exact **access-list extended** syntax. You should enable flow mobility on business-critical traffic. For example, you can limit flow mobility to only HTTPS traffic, and/or to traffic to specific servers.

b) Create a class map for the ACL:

class-map *flow_map_name*

match access-list *flow_acl_name*

c) Specify the same policy map on which you enabled LISP inspection, the flow class map, and enable flow mobility:

policy-map *policy_map_name*

class *flow_map_name*

cluster flow-mobility lisp

Example:

```
ciscoasa(config)# access-list IMPORTANT-FLOWS extended permit tcp any 10.10.10.0 255.255.255.0
 eq https
ciscoasa(config)# class-map IMPORTANT-FLOWS-MAP
ciscoasa(config)# match access-list IMPORTANT-FLOWS
ciscoasa(config-cmap)# policy-map INSIDE_POLICY
ciscoasa(config-pmap)# class IMPORTANT-FLOWS-MAP
ciscoasa(config-pmap-c)# cluster flow-mobility lisp
```

Step 4 Enter cluster group configuration mode, and enable flow mobility for the cluster:

cluster group *name*

flow-mobility lisp

This on/off toggle lets you easily enable or disable flow mobility.

Examples

The following example:

- Limits EIDs to those on the 10.10.10.0/24 network
- Inspects LISP traffic (UDP 4342) between a LISP router at 192.168.50.89 (on inside) and an ITR or ETR router (on another ASA interface) at 192.168.10.8
- Enables flow mobility for all inside traffic going to a server on 10.10.10.0/24 using HTTPS.
- Enables flow mobility for the cluster.

```
access-list TRACKED_EID_LISP extended permit ip any 10.10.10.0 255.255.255.0
policy-map type inspect lisp LISP_EID_INSPECT
   parameters
      allowed-eid access-list TRACKED_EID_LISP
      validate-key MadMaxShinyandChrome
!
access-list LISP_ACL extended permit udp host 192.168.50.89 host 192.168.10.8 eq 4342
class-map LISP_CLASS
   match access-list LISP_ACL
policy-map INSIDE_POLICY
   class LISP_CLASS
      inspect lisp LISP_EID_INSPECT
service-policy INSIDE_POLICY interface inside
!
access-list IMPORTANT-FLOWS extended permit tcp any 10.10.10.0 255.255.255.0 eq https
class-map IMPORTANT-FLOWS-MAP
   match access-list IMPORTANT-FLOWS
policy-map INSIDE_POLICY
   class IMPORTANT-FLOWS-MAP
      cluster flow-mobility lisp
!
cluster group cluster1
   flow-mobility lisp
```

Manage Cluster Nodes

After you deploy the cluster, you can change the configuration and manage cluster nodes.

Become an Inactive Node

To become an inactive member of the cluster, disable clustering on the node while leaving the clustering configuration intact.



Note When an ASA becomes inactive (either manually or through a health check failure), all data interfaces are shut down; only the management-only interface can send and receive traffic. To resume traffic flow, re-enable clustering; or you can remove the node altogether from the cluster. The management interface remains up using the IP address the node received from the cluster IP pool. However if you reload, and the node is still inactive in the cluster (for example, you saved the configuration with clustering disabled), then the management interface is disabled. You must use the console port for any further configuration.

Before you begin

- You must use the console port; you cannot enable or disable clustering from a remote CLI connection.
- For multiple context mode, perform this procedure in the system execution space. If you are not already in the System configuration mode, enter the **changeto system** command.

Procedure

Step 1 Enter cluster configuration mode:

cluster group *name*

Example:

```
ciscoasa(config)# cluster group pod1
```

Step 2 Disable clustering:

no enable

If this node was the control node, a new control election takes place, and a different member becomes the control node.

The cluster configuration is maintained, so that you can enable clustering again later.

Deactivate a Node

To deactivate a member other than the node you are logged into, perform the following steps.



Note When an ASA becomes inactive, all data interfaces are shut down; only the management-only interface can send and receive traffic. To resume traffic flow, re-enable clustering. The management interface remains up using the IP address the node received from the cluster IP pool. However if you reload, and the node is still inactive in the cluster (for example, if you saved the configuration with clustering disabled), the management interface is disabled. You must use the console port for any further configuration.

Before you begin

For multiple context mode, perform this procedure in the system execution space. If you are not already in the System configuration mode, enter the **changeto system** command.

Procedure

Remove the node from the cluster.

cluster remove unit *node_name*

The bootstrap configuration remains intact, as well as the last configuration synched from the control node, so that you can later re-add the node without losing your configuration. If you enter this command on a data node to remove the control node, a new control node is elected.

To view member names, enter **cluster remove unit ?**, or enter the **show cluster info** command.

Example:

```
ciscoasa(config)# cluster remove unit ?

Current active units in the cluster:
asa2

ciscoasa(config)# cluster remove unit asa2
WARNING: Clustering will be disabled on unit asa2. To bring it back
to the cluster please logon to that unit and re-enable clustering
```

Rejoin the Cluster

If a node was removed from the cluster, for example for a failed interface or if you manually deactivated a member, you must manually rejoin the cluster.

Before you begin

- You must use the console port to reenable clustering. Other interfaces are shut down.

- For multiple context mode, perform this procedure in the system execution space. If you are not already in the System configuration mode, enter the **changeto system** command.
- Make sure the failure is resolved before you try to rejoin the cluster.

Procedure

Step 1 At the console, enter cluster configuration mode:

cluster group *name*

Example:

```
ciscoasa(config)# cluster group pod1
```

Step 2 Enable clustering.

enable

Leave the Cluster

If you want to leave the cluster altogether, you need to remove the entire cluster bootstrap configuration. Because the current configuration on each node is the same (synced from the active unit), leaving the cluster also means either restoring a pre-clustering configuration from backup, or clearing your configuration and starting over to avoid IP address conflicts.

Before you begin

You must use the console port; when you remove the cluster configuration, all interfaces are shut down, including the management interface and cluster control link. Moreover, you cannot enable or disable clustering from a remote CLI connection.

Procedure

Step 1 For a data node, disable clustering:

cluster group *cluster_name*
no enable

Example:

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)# no enable
```

You cannot make configuration changes while clustering is enabled on a data node.

Step 2 Clear the cluster configuration:

clear configure cluster

The ASA shuts down all interfaces including the management interface and cluster control link.

Step 3 Disable cluster interface mode:

no cluster interface-mode

The mode is not stored in the configuration and must be reset manually.

Step 4 If you have a backup configuration, copy the backup configuration to the running configuration:

copy *backup_cfg* **running-config**

Example:

```
ciscoasa(config)# copy backup_cluster.cfg running-config

Source filename [backup_cluster.cfg]?

Destination filename [running-config]?
ciscoasa(config)#
```

Step 5 Save the configuration to startup:

write memory

Step 6 If you do not have a backup configuration, reconfigure management access. Be sure to change the interface IP addresses, and restore the correct hostname, for example.

Change the Control Node



Caution The best method to change the control node is to disable clustering on the control node, wait for a new control election, and then re-enable clustering. If you must specify the exact node you want to become the control node, use the procedure in this section. Note, however, that for centralized features, if you force a control node change using this procedure, then all connections are dropped, and you have to re-establish the connections on the new control node.

To change the control node, perform the following steps.

Before you begin

For multiple context mode, perform this procedure in the system execution space. If you are not already in the System configuration mode, enter the **changeto system** command.

Procedure

Set a new node as the control node:

cluster master unit*node_name*

Example:

```
ciscoasa(config)# cluster master unit asa2
```

You will need to reconnect to the Main cluster IP address.

To view member names, enter **cluster master unit ?** (to see all names except the current node), or enter the **show cluster info** command.

Execute a Command Cluster-Wide

To send a command to all nodes in the cluster, or to a specific node, perform the following steps. Sending a **show** command to all nodes collects all output and displays it on the console of the current node. Other commands, such as **capture** and **copy**, can also take advantage of cluster-wide execution.

Procedure

Send a command to all nodes, or if you specify the node name, a specific node:

cluster exec [**unit** *node_name*] *command*

Example:

```
ciscoasa# cluster exec show xlate
```

To view node names, enter **cluster exec unit ?** (to see all names except the current node), or enter the **show cluster info** command.

Examples

To copy the same capture file from all nodes in the cluster at the same time to a TFTP server, enter the following command on the control node:

```
ciscoasa# cluster exec copy /pcap capture: tftp://10.1.1.56/capture1.pcap
```

Multiple PCAP files, one from each node, are copied to the TFTP server. The destination capture file name is automatically attached with the node name, such as capture1_asa1.pcap, capture1_asa2.pcap, and so on. In this example, asa1 and asa2 are cluster node names.

The following sample output for the **cluster exec show port-channel** summary command shows EtherChannel information for each node in the cluster:

```
ciscoasa# cluster exec show port-channel summary
master(LOCAL):*********************************************************
```

```
 Number of channel-groups in use: 2
Group  Port-channel  Protocol  Span-cluster  Ports
------+-------------+-----------+-----------------------------------------------
1         Po1             LACP       Yes  Gi0/0(P)
2         Po2             LACP       Yes  Gi0/1(P)
 slave:********************************************************************
 Number of channel-groups in use: 2
Group  Port-channel  Protocol  Span-cluster   Ports
------+-------------+-----------+-----------------------------------------------
1         Po1             LACP       Yes   Gi0/0(P)
2         Po2             LACP       Yes   Gi0/1(P)
```

Monitoring the ASA Cluster

You can monitor and troubleshoot cluster status and connections.

Monitoring Cluster Status

See the following commands for monitoring cluster status:

- **show cluster info** [**health** [**details**]]

  With no keywords, the **show cluster info** command shows the status of all members of the cluster.

  The **show cluster info health** command shows the current health of interfaces, nodes, and the cluster overall. The **details** keyword shows the number heartbeat message failures.

  See the following output for the **show cluster info** command:

```
ciscoasa# show cluster info
Cluster stbu: On
  This is "C" in state SLAVE
      ID        : 0
      Site ID   : 1
            Version   : 9.4(1)
      Serial No.: P3000000025
      CCL IP    : 10.0.0.3
      CCL MAC   : 000b.fcf8.c192
      Last join : 17:08:59 UTC Sep 26 2011
      Last leave: N/A
Other members in the cluster:
  Unit "D" in state SLAVE
      ID        : 1
      Site ID   : 1
            Version   : 9.4(1)
      Serial No.: P3000000001
      CCL IP    : 10.0.0.4
      CCL MAC   : 000b.fcf8.c162
      Last join : 19:13:11 UTC Sep 23 2011
      Last leave: N/A
  Unit "A" in state MASTER
      ID        : 2
      Site ID   : 2
            Version   : 9.4(1)
      Serial No.: JAB0815R0JY
      CCL IP    : 10.0.0.1
      CCL MAC   : 000f.f775.541e
      Last join : 19:13:20 UTC Sep 23 2011
```

```
    Last leave: N/A
Unit "B" in state SLAVE
    ID        : 3
    Site ID   : 2
          Version   : 9.4(1)
    Serial No.: P3000000191
    CCL IP    : 10.0.0.2
    CCL MAC   : 000b.fcf8.c61e
    Last join : 19:13:50 UTC Sep 23 2011
    Last leave: 19:13:36 UTC Sep 23 2011
```

- **show cluster info auto-join**

  Shows whether the cluster node will automatically rejoin the cluster after a time delay and if the failure conditions (such as waiting for the license, chassis health check failure, and so on) are cleared. If the node is permanently disabled, or if the node is already in the cluster, then this command will not show any output.

  See the following outputs for the **show cluster info auto-join** command:

```
ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster in 253 seconds.
Quit reason: Received control message DISABLE

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Master has application down that slave has up.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Chassis-blade health check failed.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Service chain application became down.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Unit is kicked out from cluster because of Application health check failure.


ciscoasa(cfg-cluster)# show cluster info auto-join
Unit join is pending (waiting for the smart license entitlement: ent1)

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit join is pending (waiting for the smart license export control flag)
```

- **show cluster info transport** {**asp** | **cp** [**detail**]}

  Shows transport related statistics for the following:

  - **asp** —Data plane transport statistics.
  - **cp** —Control plane transport statistics.

    If you enter the **detail** keyword, you can view cluster reliable transport protocol usage so you can identify packet drop issues when the buffer is full in the control plane. See the following output for the **show cluster info transport cp detail** command:

```
ciscoasa# show cluster info transport cp detail
```

```
Member ID to name mapping:
  0 - unit-1-1   2 - unit-4-1   3 - unit-2-1

Legend:
  U    - unreliable messages
  UE   - unreliable messages error
  SN   - sequence number
  ESN  - expecting sequence number
  R    - reliable messages
  RE   - reliable messages error
  RDC  - reliable message deliveries confirmed
  RA   - reliable ack packets received
  RFR  - reliable fast retransmits
  RTR  - reliable timer-based retransmits
  RDP  - reliable message dropped
  RDPR - reliable message drops reported
  RI   - reliable message with old sequence number
  RO   - reliable message with out of order sequence number
  ROW  - reliable message with out of window sequence number
  ROB  - out of order reliable messages buffered
  RAS  - reliable ack packets sent

This unit as a sender
--------------------------
        all       0         2         3
  U     123301    3867966   3230662   3850381
  UE    0         0         0         0
  SN    1656a4ce  acb26fe   5f839f76  7b680831
  R     733840    1042168   852285    867311
  RE    0         0         0         0
  RDC   699789    934969    740874    756490
  RA    385525    281198    204021    205384
  RFR   27626     56397     0         0
  RTR   34051     107199    111411    110821
  RDP   0         0         0         0
  RDPR  0         0         0         0

This unit as a receiver of broadcast messages
----------------------------------------------
        0         2         3
  U     111847    121862    120029
  R     7503      665700    749288
  ESN   5d75b4b3  6d81d23   365ddd50
  RI    630       34278     40291
  RO    0         582       850
  ROW   0         566       850
  ROB   0         16        0
  RAS   1571      123289    142256

This unit as a receiver of unicast messages
-------------------------------------------
        0         2         3
  U     1         3308122   4370233
  R     513846    879979    1009492
  ESN   4458903a  6d841a84  7b4e7fa7
  RI    66024     108924    102114
  RO    0         0         0
  ROW   0         0         0
  ROB   0         0         0
  RAS   130258    218924    228303

Gated Tx Buffered Message Statistics
------------------------------------
    current sequence number: 0
```

```
    total:                  0
    current:                0
    high watermark:         0

    delivered:              0
    deliver failures:       0

    buffer full drops:      0
    message truncate drops:  0

    gate close ref count:    0

    num of supported clients:45

MRT Tx of broadcast messages
============================
Message high watermark: 3%
  Total messages buffered at high watermark: 5677
  [Per-client message usage at high watermark]
  ----------------------------------------------------------------
  Client name                          Total messages  Percentage
  Cluster Redirect Client                        4153         73%
  Route Cluster Client                            419          7%
  RRI Cluster Client                             1105         19%

Current MRT buffer usage: 0%
  Total messages buffered in real-time: 1
  [Per-client message usage in real-time]
  Legend:
        F - MRT messages sending when buffer is full
        L - MRT messages sending when cluster node leave
        R - MRT messages sending in Rx thread
  ------------------------------------------------------------------------------
  Client name                          Total messages  Percentage    F   L   R
  VPN Clustering HA Client                          1        100%    0   0   0

MRT Tx of unitcast messages(to member_id:0)
===========================================
Message high watermark: 31%
  Total messages buffered at high watermark: 4059
  [Per-client message usage at high watermark]
  ----------------------------------------------------------------
  Client name                          Total messages  Percentage
  Cluster Redirect Client                        3731         91%
  RRI Cluster Client                              328          8%

Current MRT buffer usage: 29%
  Total messages buffered in real-time: 3924
  [Per-client message usage in real-time]
  Legend:
        F - MRT messages sending when buffer is full
        L - MRT messages sending when cluster node leave
        R - MRT messages sending in Rx thread
  ------------------------------------------------------------------------------
  Client name                          Total messages  Percentage    F   L   R
  Cluster Redirect Client                        3607         91%    0   0   0
  RRI Cluster Client                              317          8%    0   0   0

MRT Tx of unitcast messages(to member_id:2)
===========================================
Message high watermark: 14%
  Total messages buffered at high watermark: 578
  [Per-client message usage at high watermark]
```

```
  ------------------------------------------------------------
  Client name                          Total messages  Percentage
  VPN Clustering HA Client                        578        100%

Current MRT buffer usage: 0%
  Total messages buffered in real-time: 0

MRT Tx of unitcast messages(to member_id:3)
===========================================
Message high watermark: 12%
  Total messages buffered at high watermark: 573
  [Per-client message usage at high watermark]
  ------------------------------------------------------------
  Client name                          Total messages  Percentage
  VPN Clustering HA Client                        572         99%
  Cluster VPN Unique ID Client                      1          0%

Current MRT buffer usage: 0%
  Total messages buffered in real-time: 0
```

- **show cluster history**

  Shows the cluster history, as well as error messages about why a cluster node failed to join or why a node left the cluster.

Capturing Packets Cluster-Wide

See the following command for capturing packets in a cluster:

cluster exec capture

To support cluster-wide troubleshooting, you can enable capture of cluster-specific traffic on the control node using the **cluster exec capture** command, which is then automatically enabled on all of the data nodes in the cluster.

Monitoring Cluster Resources

See the following command for monitoring cluster resources:

show cluster {**cpu** | **memory** | **resource**} [*options*]

Displays aggregated data for the entire cluster. The *options* available depends on the data type.

Monitoring Cluster Traffic

See the following commands for monitoring cluster traffic:

- **show conn [detail], cluster exec show conn**

  The **show conn** command shows whether a flow is a director, backup, or forwarder flow. Use the **cluster exec show conn** command on any node to view all connections. This command can show how traffic for a single flow arrives at different ASAs in the cluster. The throughput of the cluster is dependent on the efficiency and configuration of load balancing. This command provides an easy way to view how traffic for a connection is flowing through the cluster, and can help you understand how a load balancer might affect the performance of a flow.

The **show conn detail** command also shows which flows are subject to flow mobility.

The following is sample output for the **show conn detail** command:

```
ciscoasa/ASA2/slave# show conn detail
12 in use, 13 most used
Cluster stub connections: 0 in use, 46 most used
Flags: A - awaiting inside ACK to SYN, a - awaiting outside ACK to SYN,
       B - initial SYN from outside, b - TCP state-bypass or nailed,
       C - CTIQBE media, c - cluster centralized,
       D - DNS, d - dump, E - outside back connection, e - semi-distributed,
       F - outside FIN, f - inside FIN,
       G - group, g - MGCP, H - H.323, h - H.225.0, I - inbound data,
       i - incomplete, J - GTP, j - GTP data, K - GTP t3-response
       k - Skinny media, L - LISP triggered flow owner mobility,
       M - SMTP data, m - SIP media, n - GUP
       O - outbound data, o - offloaded,
       P - inside back connection,
       Q - Diameter, q - SQL*Net data,
       R - outside acknowledged FIN,
       R - UDP SUNRPC, r - inside acknowledged FIN, S - awaiting inside SYN,
       s - awaiting outside SYN, T - SIP, t - SIP transient, U - up,
       V - VPN orphan, W - WAAS,
       w - secondary domain backup,
       X - inspected by service module,
       x - per session, Y - director stub flow, y - backup stub flow,
       Z - Scansafe redirection, z - forwarding stub flow
ESP outside: 10.1.227.1/53744 NP Identity Ifc: 10.1.226.1/30604, , flags c, idle 0s,
uptime
1m21s, timeout 30s, bytes 7544, cluster sent/rcvd bytes 0/0, owners (0,255) Traffic
received
at interface outside Locally received: 7544 (93 byte/s) Traffic received at interface
NP
Identity Ifc Locally received: 0 (0 byte/s) UDP outside: 10.1.227.1/500 NP Identity
Ifc:
10.1.226.1/500, flags -c, idle 1m22s, uptime 1m22s, timeout 2m0s, bytes 1580, cluster
sent/rcvd bytes 0/0, cluster sent/rcvd total bytes 0/0, owners (0,255) Traffic received
 at
interface outside Locally received: 864 (10 byte/s) Traffic received at interface NP
Identity
Ifc Locally received: 716 (8 byte/s)
```

To troubleshoot the connection flow, first see connections on all nodes by entering the **cluster exec show conn** command on any node. Look for flows that have the following flags: director (Y), backup (y), and forwarder (z). The following example shows an SSH connection from 172.18.124.187:22 to 192.168.103.131:44727 on all three ASAs; ASA 1 has the z flag showing it is a forwarder for the connection, ASA3 has the Y flag showing it is the director for the connection, and ASA2 has no special flags showing it is the owner. In the outbound direction, the packets for this connection enter the inside interface on ASA2 and exit the outside interface. In the inbound direction, the packets for this connection enter the outside interface on ASA 1 and ASA3, are forwarded over the cluster control link to ASA2, and then exit the inside interface on ASA2.

```
ciscoasa/ASA1/master# cluster exec show conn
ASA1(LOCAL):**********************************************************
18 in use, 22 most used
Cluster stub connections: 0 in use, 5 most used
TCP outside  172.18.124.187:22 inside  192.168.103.131:44727, idle 0:00:00, bytes
37240828, flags z
```

```
ASA2:**********************************************************************
12 in use, 13 most used
Cluster stub connections: 0 in use, 46 most used
TCP outside  172.18.124.187:22 inside  192.168.103.131:44727, idle 0:00:00, bytes
37240828, flags UIO


ASA3:**********************************************************************
10 in use, 12 most used
Cluster stub connections: 2 in use, 29 most used
TCP outside  172.18.124.187:22 inside  192.168.103.131:44727, idle 0:00:03, bytes 0,
flags  Y
```

- **show cluster info** [**conn-distribution** | **packet-distribution** | **loadbalance** | **flow-mobility counters**]

  The **show cluster info conn-distribution** and **show cluster info packet-distribution** commands show traffic distribution across all cluster nodes. These commands can help you to evaluate and adjust the external load balancer.

  The **show cluster info loadbalance** command shows connection rebalance statistics.

  The **show cluster info flow-mobility counters** command shows EID movement and flow owner movement information. See the following output for **show cluster info flow-mobility counters**:

```
ciscoasa# show cluster info flow-mobility counters
EID movement notification received  : 4
EID movement notification processed : 4
Flow owner moving requested         : 2
```

- **show cluster info load-monitor** [**details**]

  The **show cluster info load-monitor** command shows the traffic load for cluster members for the last interval and also the average over total number of intervals configured (30 by default). Use the **details** keyword to view the value for each measure at each interval.

```
ciscoasa(cfg-cluster)# show cluster info load-monitor
ID  Unit Name
0  B
1  A_1
Information from all units with 20 second interval:
Unit     Connections    Buffer Drops   Memory Used   CPU Used
Average from last 1 interval:
   0           0            0            14              25
   1           0            0            16              20
Average from last 30 interval:
   0           0            0            12              28
   1           0            0            13              27


ciscoasa(cfg-cluster)# show cluster info load-monitor details


ID  Unit Name

 0  B

 1  A_1

Information from all units with 20 second interval
```

```
Connection count captured over 30 intervals:

Unit ID  0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

Unit ID  1

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0


Buffer drops captured over 30 intervals:

Unit ID  0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

Unit ID  1

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0


Memory usage(%) captured over 30 intervals:

Unit ID  0

        25         25         30         30         30         35

        25         25         35         30         30         30
```

```
        25          25          30          25          25          35
        30          30          30          25          25          25
        25          20          30          30          30          30
Unit ID  1
        30          25          35          25          30          30
        25          25          35          25          30          35
        30          30          35          30          30          30
        25          20          30          25          25          30
        20          30          35          30          30          35


CPU usage(%) captured over 30 intervals:
Unit ID  0
        25          25          30          30          30          35
        25          25          35          30          30          30
        25          25          30          25          25          35
        30          30          30          25          25          25
        25          20          30          30          30          30
Unit ID  1
        30          25          35          25          30          30
        25          25          35          25          30          35
        30          30          35          30          30          30
        25          20          30          25          25          30
        20          30          35          30          30          35
```

- **show cluster** {**access-list** | **conn** | **traffic** | **user-identity | xlate**} [*options*]

  Displays aggregated data for the entire cluster. The *options* available depends on the data type.

  See the following output for the **show cluster access-list** command:

```
ciscoasa# show cluster access-list
hitcnt display order: cluster-wide aggregated result, unit-A, unit-B,  unit-C, unit-D
access-list cached ACL log flows: total 0, denied 0 (deny-flow-max 4096) alert-interval
 300
access-list 101; 122 elements; name hash: 0xe7d586b5
access-list 101 line 1 extended permit tcp 192.168.143.0 255.255.255.0 any eq www
(hitcnt=0, 0, 0, 0, 0) 0x207a2b7d
access-list 101 line 2 extended permit tcp any 192.168.143.0 255.255.255.0 (hitcnt=0,
0, 0, 0, 0) 0xfe4f4947
access-list  101 line 3 extended permit tcp host 192.168.1.183 host 192.168.43.238
```

```
(hitcnt=1, 0, 0, 0, 1) 0x7b521307
access-list 101 line 4 extended permit tcp host 192.168.1.116 host 192.168.43.238
(hitcnt=0, 0, 0, 0, 0) 0x5795c069
access-list 101 line 5 extended permit tcp host 192.168.1.177 host 192.168.43.238
(hitcnt=1, 0, 0, 1, 0) 0x51bde7ee
access list 101 line 6 extended permit tcp host 192.168.1.177 host 192.168.43.13
(hitcnt=0, 0, 0, 0, 0) 0x1e68697c
access-list 101 line 7 extended permit tcp host 192.168.1.177 host 192.168.43.132
(hitcnt=2, 0, 0, 1, 1) 0xc1ce5c49
access-list 101 line 8 extended permit tcp host 192.168.1.177 host 192.168.43.192
(hitcnt=3, 0, 1, 1, 1) 0xb6f59512
access-list 101 line 9 extended permit tcp host 192.168.1.177 host 192.168.43.44
(hitcnt=0, 0, 0, 0, 0) 0xdc104200
access-list 101 line 10 extended permit tcp host 192.168.1.112 host 192.168.43.44
(hitcnt=429, 109, 107, 109, 104)
0xce4f281d
access-list 101 line 11 extended permit tcp host 192.168.1.170 host 192.168.43.238
(hitcnt=3, 1, 0, 0, 2) 0x4143a818
access-list 101 line 12 extended permit tcp host 192.168.1.170 host 192.168.43.169
(hitcnt=2, 0, 1, 0, 1) 0xb18dfea4
access-list 101 line 13 extended permit tcp host 192.168.1.170 host 192.168.43.229
(hitcnt=1, 1, 0, 0, 0) 0x21557d71
access-list 101 line 14 extended permit tcp host 192.168.1.170 host 192.168.43.106
(hitcnt=0, 0, 0, 0, 0) 0x7316e016
access-list 101 line 15 extended permit tcp host 192.168.1.170 host 192.168.43.196
(hitcnt=0, 0, 0, 0, 0) 0x013fd5b8
access-list 101 line 16 extended permit tcp host 192.168.1.170 host 192.168.43.75
(hitcnt=0, 0, 0, 0, 0) 0x2c7dba0d
```

To display the aggregated count of in-use connections for all nodes, enter:

```
ciscoasa# show cluster conn count
Usage Summary In Cluster:*****************************************
  200 in use (cluster-wide aggregated)
     cl2(LOCAL):**********************************************************
  100 in use, 100 most used

  cl1:*************************************************************
  100 in use, 100 most used
```

- **show asp cluster counter**

  This command is useful for datapath troubleshooting.

Monitoring Cluster Routing

See the following commands for cluster routing:

- **show route cluster**
- **debug route cluster**

  Shows cluster information for routing.

- **show lisp eid**

  Shows the ASA EID table showing EIDs and site IDs.

  See the following output from the **cluster exec show lisp eid** command.

```
ciscoasa# cluster exec show lisp eid
L1(LOCAL):*********************************************************
     LISP EID       Site ID
    33.44.33.105        2
    33.44.33.201        2
    11.22.11.1             4
    11.22.11.2             4
L2:*********************************************************************
     LISP EID       Site ID
    33.44.33.105        2
    33.44.33.201        2
    11.22.11.1   4
    11.22.11.2  4
```

- **show asp table classify domain inspect-lisp**

  This command is useful for troubleshooting.

Configuring Logging for Clustering

See the following command for configuring logging for clustering:

logging device-id

Each node in the cluster generates syslog messages independently. You can use the **logging device-id** command to generate syslog messages with identical or different device IDs to make messages appear to come from the same or different nodes in the cluster.

Monitoring Cluster Interfaces

See the following commands for monitoring cluster interfaces:

- **show cluster interface-mode**

  Shows the cluster interface mode.

- **show port-channel**

  Includes information about whether a port-channel is spanned.

- **show lacp cluster {system-mac | system-id}**

  Shows the cLACP system ID and priority.

- **debug lacp cluster [all | ccp | misc | protocol]**

  Shows debug messages for cLACP.

- **show interface**

  Shows the use of the site MAC address when in use:

```
ciscoasa# show interface port-channel1.3151
Interface Port-channel1.3151 "inside", is up, line protocol is up
Hardware is EtherChannel/LACP, BW 1000 Mbps, DLY 10 usec
VLAN identifier 3151
MAC address aaaa.1111.1234, MTU 1500
Site Specific MAC address aaaa.1111.aaaa
```

```
IP address 10.3.1.1, subnet mask 255.255.255.0
Traffic Statistics for "inside":
132269 packets input, 6483425 bytes
1062 packets output, 110448 bytes
98530 packets dropped
```



Note When comparing the packets on the cluster control link in ASA vs. FXOS, the FXOS packet count is higher than what is shown in ASA. In ASA, only packets destined for the cluster control link IP address are counted towards input packets. Forwarded packets, which are re-injected to the data interfaces, only get counted towards the *input* statistics of data interfaces and the *output* statistics of the cluster control link.

Debugging Clustering

See the following commands for debugging clustering:

- **debug cluster [ccp | datapath | fsm | general | hc | license | rpc | transport]**

  Shows debug messages for clustering.

- **debug cluster flow-mobility**

  Shows events related to clustering flow mobility.

- **debug lisp eid-notify-intercept**

  Shows events when the eid-notify message is intercepted.

- **show cluster info trace**

  The **show cluster info trace** command shows the debug information for further troubleshooting.

  See the following output for the **show cluster info trace** command:

```
ciscoasa# show cluster info trace
 Feb 02 14:19:47.456 [DBUG]Receive CCP message: CCP_MSG_LOAD_BALANCE
 Feb 02 14:19:47.456 [DBUG]Receive CCP message: CCP_MSG_LOAD_BALANCE
 Feb 02 14:19:47.456 [DBUG]Send CCP message to all: CCP_MSG_KEEPALIVE from 80-1 at
MASTER
```

  For example, if you see the following messages showing that two nodes with the same **local-unit** name are acting as the control node, it could mean that either two nodes have the same **local-unit** name (check your configuration), or a node is receiving its own broadcast messages (check your network).

```
ciscoasa# show cluster info trace
 May 23 07:27:23.113 [CRIT]Received datapath event 'multi masters' with parameter 1.
 May 23 07:27:23.113 [CRIT]Found both unit-9-1 and unit-9-1 as master units. Master
role retained by unit-9-1, unit-9-1 will leave then join as a Slave
 May 23 07:27:23.113 [DBUG]Send event (DISABLE, RESTART | INTERNAL-EVENT, 5000 msecs,
Detected another Master, leave and re-join as Slave) to FSM. Current state MASTER
 May 23 07:27:23.113 [INFO]State machine changed from state MASTER to DISABLED
```

Examples for ASA Clustering

These examples include all cluster-related ASA configuration for typical deployments.

Sample ASA and Switch Configuration

The following sample configurations connect the following interfaces between the ASA and the switch:

ASA Interface	Switch Interface
Ethernet 1/2	GigabitEthernet 1/0/15
Ethernet 1/3	GigabitEthernet 1/0/16
Ethernet 1/4	GigabitEthernet 1/0/17
Ethernet 1/5	GigabitEthernet 1/0/18

ASA Configuration

Interface Mode on Each Unit

```
cluster interface-mode spanned force
```

ASA1 Control Unit Bootstrap Configuration

```
interface Ethernet1/6
 channel-group 1 mode on
 no shutdown
!
interface Ethernet1/7
 channel-group 1 mode on
 no shutdown
!
interface Port-channel1
 description Clustering Interface
!
cluster group Moya
 local-unit A
 cluster-interface Port-channel1 ip 10.0.0.1 255.255.255.0
 priority 10
 key emphyri0
 enable noconfirm
```

ASA2 Data Unit Bootstrap Configuration

```
interface Ethernet1/6
 channel-group 1 mode on
 no shutdown
!
interface Ethernet1/7
```

```
 channel-group 1 mode on
 no shutdown
!
interface Port-channel1
 description Clustering Interface
!
cluster group Moya
 local-unit B
 cluster-interface Port-channel1 ip 10.0.0.2 255.255.255.0
 priority 11
 key emphyri0
 enable as-slave
```

Control Unit Interface Configuration

```
ip local pool mgmt-pool 10.53.195.231-10.53.195.232

interface Ethernet1/2
 channel-group 10 mode active
 no shutdown
!
interface Ethernet1/3
 channel-group 10 mode active
 no shutdown
!
interface Ethernet1/4
 channel-group 11 mode active
 no shutdown
!
interface Ethernet1/5
 channel-group 11 mode active
 no shutdown
!
interface Management1/1
 management-only
 nameif management
 ip address 10.53.195.230 cluster-pool mgmt-pool
 security-level 100
 no shutdown
!
interface Port-channel10
 mac-address aaaa.bbbb.cccc
 nameif inside
 security-level 100
 ip address 209.165.200.225 255.255.255.224
!
interface Port-channel11
 mac-address aaaa.dddd.cccc
 nameif outside
 security-level 0
 ip address 209.165.201.1 255.255.255.224
```

Cisco IOS Switch Configuration

```
interface GigabitEthernet1/0/15
 switchport access vlan 201
 switchport mode access
 spanning-tree portfast
 channel-group 10 mode active
```

```
!
interface GigabitEthernet1/0/16
 switchport access vlan 201
 switchport mode access
 spanning-tree portfast
 channel-group 10 mode active
!
interface GigabitEthernet1/0/17
 switchport access vlan 401
 switchport mode access
 spanning-tree portfast
 channel-group 11 mode active
!
interface GigabitEthernet1/0/18
 switchport access vlan 401
 switchport mode access
 spanning-tree portfast
 channel-group 11 mode active

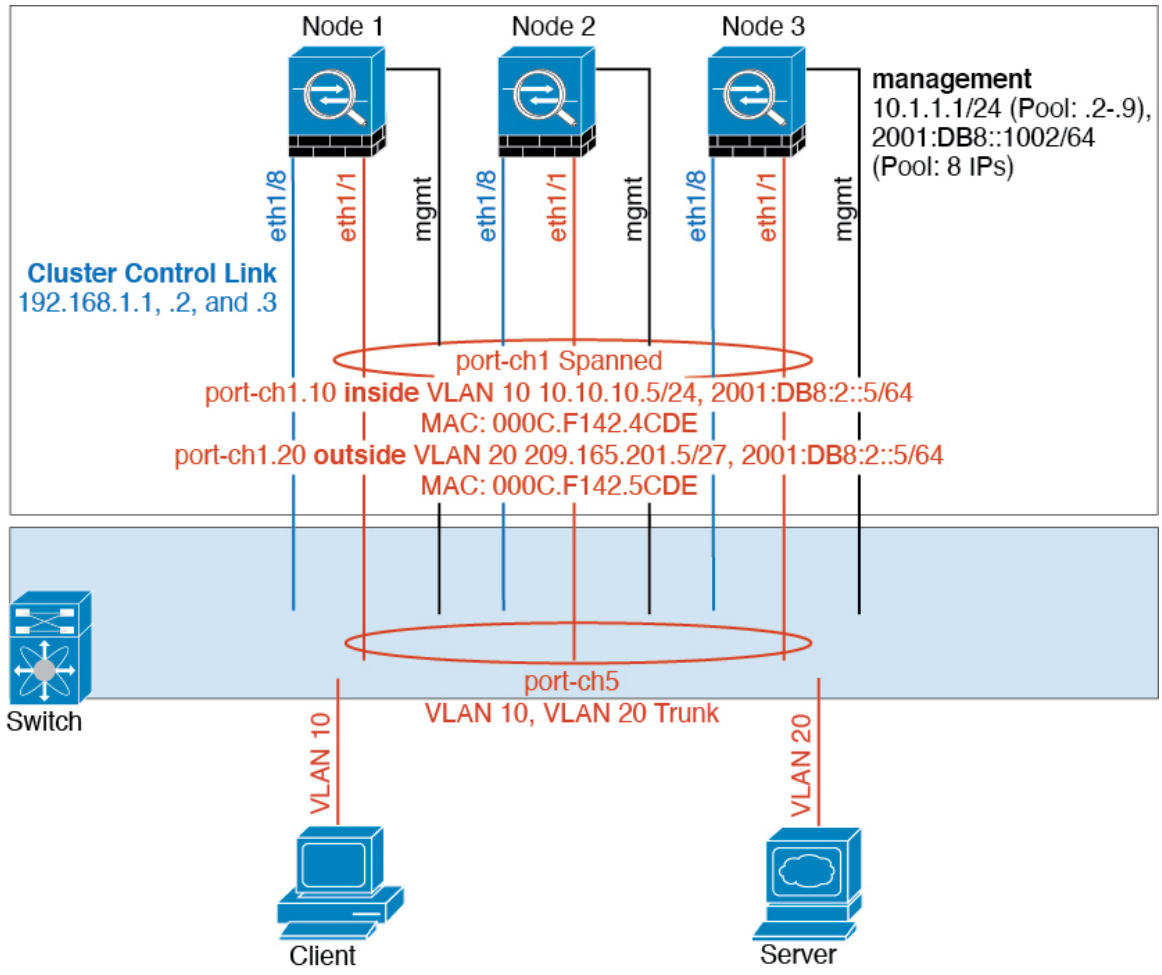
interface Port-channel10
 switchport access vlan 201
 switchport mode access

interface Port-channel11
 switchport access vlan 401
 switchport mode access
```

Firewall on a stick

Data traffic from different security domains are associated with different VLANs. For example, VLAN 10 for the inside network and VLAN 20 for the outside network. Each ASA has a single physical port connected to the external switch or router. Trunking is enabled so that all packets on the physical link are 802.1q encapsulated. The ASA is the firewall between VLAN 10 and VLAN 20.

When using Spanned EtherChannels, all data links are grouped into one EtherChannel on the switch side. If the ASA becomes unavailable, the switch will rebalance traffic between the remaining units.



Interface mode on each unit

```
cluster interface-mode spanned force
```

Unit 1 control unit bootstrap configuration

```
interface ethernet1/8
no shutdown
description CCL

cluster group cluster1
local-unit asa1
cluster-interface ethernet1/8 ip 192.168.1.1 255.255.255.0
priority 1
key chuntheunavoidable
enable noconfirm
```

Unit 2 data unit bootstrap configuration

```
interface ethernet1/8
no shutdown
description CCL

cluster group cluster1
local-unit asa2
cluster-interface ethernet1/8 ip 192.168.1.2 255.255.255.0
priority 2
key chuntheunavoidable
enable as-slave
```

Unit 3 data unit bootstrap configuration

```
interface ethernet1/8
no shutdown
description CCL

cluster group cluster1
local-unit asa3
cluster-interface ethernet1/8 ip 192.168.1.3 255.255.255.0
priority 3
key chuntheunavoidable
enable as-slave
```

Control unit interface configuration

```
ip local pool mgmt 10.1.1.2-10.1.1.9
ipv6 local pool mgmtipv6 2001:DB8::1002/64 8

interface management 1/1
nameif management
ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
ipv6 address 2001:DB8::1001/32 cluster-pool mgmtipv6
security-level 100
management-only
no shutdown

interface ethernet1/1
channel-group 1 mode active
no shutdown

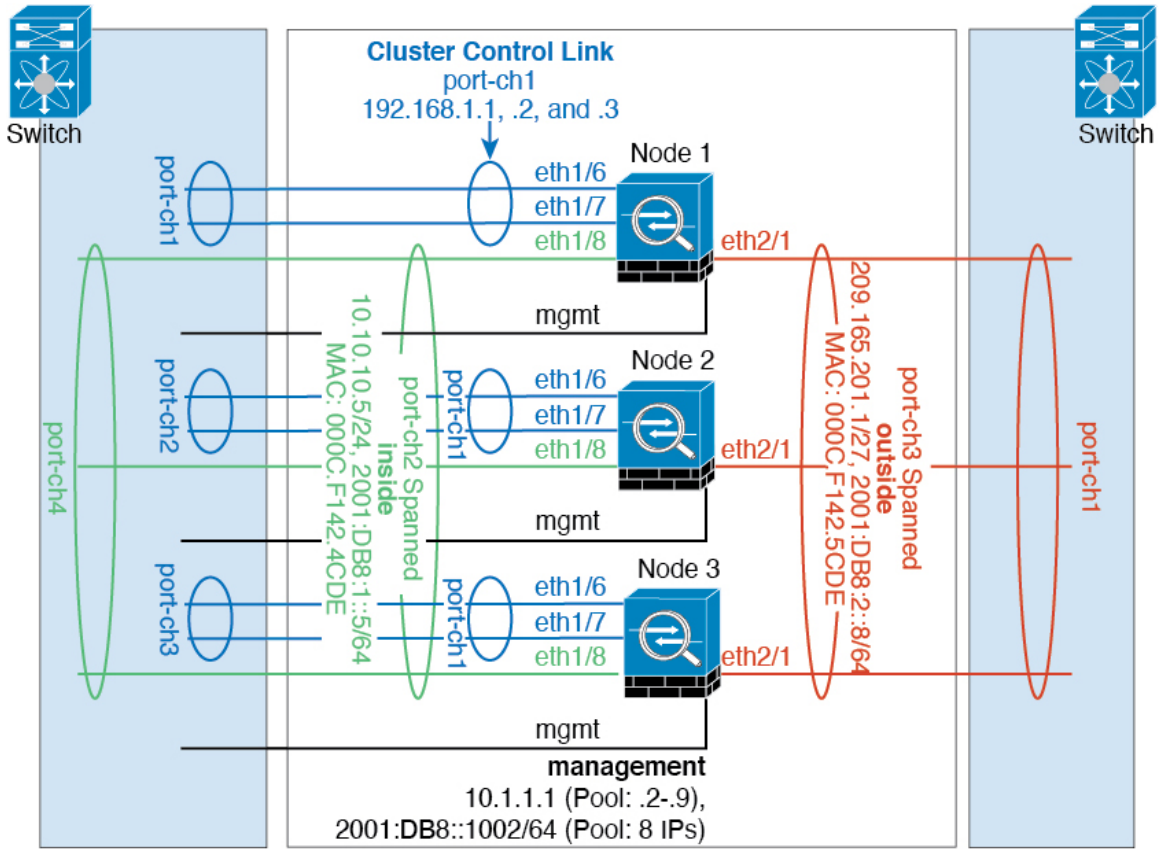
interface port-channel 1

interface port-channel 1.10
vlan 10
nameif inside
ip address 10.10.10.5 255.255.255.0
ipv6 address 2001:DB8:1::5/64
mac-address 000C.F142.4CDE

interface port-channel 1.20
vlan 20
nameif outside
ip address 209.165.201.1 255.255.255.224
ipv6 address 2001:DB8:2::8/64
mac-address 000C.F142.5CDE
```

Traffic segregation

You may prefer physical separation of traffic between the inside and outside network.



As shown in the diagram above, there is one Spanned EtherChannel on the left side that connects to the inside switch, and the other on the right side to outside switch. You can also create VLAN subinterfaces on each EtherChannel if desired.

Interface mode on each unit

```
cluster interface-mode spanned force
```

Unit 1 control unit bootstrap configuration

```
interface ethernet 1/6
 channel-group 1 mode on
 no shutdown

interface ethernet 1/7
 channel-group 1 mode on
 no shutdown

interface port-channel 1
 description CCL
```

```
cluster group cluster1
 local-unit asa1
 cluster-interface port-channel1 ip 192.168.1.1 255.255.255.0
 priority 1
 key chuntheunavoidable
 enable noconfirm
```

Unit 2 data unit bootstrap configuration

```
interface ethernet 1/6
 channel-group 1 mode on
 no shutdown

interface ethernet 1/7
 channel-group 1 mode on
 no shutdown

interface port-channel 1
 description CCL

cluster group cluster1
 local-unit asa2
 cluster-interface port-channel1 ip 192.168.1.2 255.255.255.0
 priority 2
 key chuntheunavoidable
 enable as-slave
```

Unit 3 data unit bootstrap configuration

```
interface ethernet 1/6
 channel-group 1 mode on
 no shutdown

interface ethernet 1/7
 channel-group 1 mode on
 no shutdown

interface port-channel 1
 description CCL

cluster group cluster1
 local-unit asa3
 cluster-interface port-channel1 ip 192.168.1.3 255.255.255.0
 priority 3
 key chuntheunavoidable
 enable as-slave
```

Control unit interface configuration

```
ip local pool mgmt 10.1.1.2-10.1.1.9
ipv6 local pool mgmtipv6 2001:DB8::1002/64 8

interface management 1/1
 nameif management
 ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
 ipv6 address 2001:DB8::1001/32 cluster-pool mgmtipv6
 security-level 100
```

```
 management-only
 no shutdown

interface ethernet 1/8
 channel-group 2 mode active
 no shutdown

interface port-channel 2
 nameif inside
 ip address 10.10.10.5 255.255.255.0
 ipv6 address 2001:DB8:1::5/64
 mac-address 000C.F142.4CDE

interface ethernet 2/1
 channel-group 3 mode active
 no shutdown

interface port-channel 3
 nameif outside
 ip address 209.165.201.1 255.255.255.224
 ipv6 address 2001:DB8:2::8/64
 mac-address 000C.F142.5CDE
```

OTV Configuration for Routed Mode Inter-Site Clustering

The success of inter-site clustering for routed mode with Spanned EtherChannels depends on the proper configuration and monitoring of OTV. OTV plays a major role by forwarding the packets across the DCI. OTV forwards unicast packets across the DCI only when it learns the MAC address in its forwarding table. If the MAC address is not learned in the OTV forwarding table, it will drop the unicast packets.

Sample OTV Configuration

```
//Sample OTV config:
//3151 – Inside VLAN, 3152 – Outside VLAN, 202 – CCL VLAN
//aaaa.1111.1234 – ASA inside interface global vMAC
//0050.56A8.3D22 – Server MAC

feature ospf
feature otv

mac access-list ALL_MACs
  10 permit any any
mac access-list HSRP_VMAC
  10 permit aaaa.1111.1234 0000.0000.0000 any
  20 permit aaaa.2222.1234 0000.0000.0000 any
  30 permit any aaaa.1111.1234 0000.0000.0000
  40 permit any aaaa.2222.1234 0000.0000.0000
vlan access-map Local 10
  match mac address HSRP_VMAC
  action drop
vlan access-map Local 20
  match mac address ALL_MACs
  action forward
vlan filter Local vlan-list 3151-3152


//To block global MAC with ARP inspection:
arp access-list HSRP_VMAC_ARP
  10 deny aaaa.1111.1234 0000.0000.0000 any
```

```
  20 deny aaaa.2222.1234 0000.0000.0000 any
  30 deny any aaaa.1111.1234 0000.0000.0000
  40 deny any aaaa.2222.1234 0000.0000.0000
  50 permit ip any mac
ip arp inspection filter HSRP_VMAC_ARP 3151-3152

no ip igmp snooping optimise-multicast-flood
vlan 1,202,1111,2222,3151-3152

otv site-vlan 2222
mac-list GMAC_DENY seq 10 deny aaaa.aaaa.aaaa ffff.ffff.ffff
mac-list GMAC_DENY seq 20 deny aaaa.bbbb.bbbb ffff.ffff.ffff
mac-list GMAC_DENY seq 30 permit 0000.0000.0000 0000.0000.0000
route-map stop-GMAC permit 10
  match mac-list GMAC_DENY

interface Overlay1
  otv join-interface Ethernet8/1
  otv control-group 239.1.1.1
  otv data-group 232.1.1.0/28
  otv extend-vlan 202, 3151
  otv arp-nd timeout 60
  no shutdown

interface Ethernet8/1
  description uplink_to_OTV_cloud
  mtu 9198
  ip address 10.4.0.18/24
  ip igmp version 3
  no shutdown

interface Ethernet8/2

interface Ethernet8/3
  description back_to_default_vdc_e6/39
  switchport
    switchport mode trunk
    switchport trunk allowed vlan 202,2222,3151-3152
  mac packet-classify
  no shutdown

otv-isis default
  vpn Overlay1
    redistribute filter route-map stop-GMAC
otv site-identifier 0x2
//OTV flood not required for ARP inspection:
otv flood mac 0050.56A8.3D22 vlan 3151
```

OTV Filter Modifications Required Because of Site Failure

If a site goes down, the filters need to be removed from OTV because you do not want to block the global MAC address anymore. There are some additional configurations required.

You need to add a static entry for the ASA global MAC address on the OTV switch in the site that is functional. This entry will let the OTV at the other end add these entries on the overlay interface. This step is required because if the server and client already have the ARP entry for the ASA, which is the case for existing connections, then they will not send the ARP again. Therefore, OTV will not get a chance to learn the ASA global MAC address in its forwarding table. Because OTV does not have the global MAC address in its forwarding table, and per OTV design it will not flood unicast packets over the overlay interface, then it will drop the unicast packets to the global MAC address from the server, and the existing connections will break.

```
//OTV filter configs when one of the sites is down

mac-list GMAC_A seq 10 permit 0000.0000.0000 0000.0000.0000
route-map a-GMAC permit 10
  match mac-list GMAC_A

otv-isis default
  vpn Overlay1
    redistribute filter route-map a-GMAC

no vlan filter Local vlan-list 3151

//For ARP inspection, allow global MAC:
arp access-list HSRP_VMAC_ARP_Allow
  50 permit ip any mac
ip arp inspection filter HSRP_VMAC_ARP_Allow 3151-3152

mac address-table static aaaa.1111.1234 vlan 3151 interface Ethernet8/3
//Static entry required only in the OTV in the functioning Site
```

When the other site is restored, you need to add the filters back again and remove this static entry on the OTV. It is very important to clear the dynamic MAC address table on both the OTVs to clear the overlay entry for the global MAC address.

MAC Address Table Clearing

When a site goes down, and a static entry for the global MAC address is added to OTV, you need to let the other OTV learn the global MAC address on the overlay interface. After the other site comes up, these entries should be cleared. Make sure to clear the mac address table to make sure OTV does not have these entries in its forwarding table.

```
cluster-N7k6-OTV# show mac address-table
Legend:
* - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
age - seconds since last seen,+ - primary entry using vPC Peer-Link,
(T) - True, (F) - False
VLAN MAC Address Type age Secure NTFY Ports/SWID.SSID.LID
---------+-----------------+--------+---------+------+----+------------------
G -    d867.d900.2e42 static  -  F F sup-eth1(R)
O 202  885a.92f6.44a5 dynamic -  F F Overlay1
* 202  885a.92f6.4b8f dynamic 5  F F Eth8/3
O 3151 0050.5660.9412 dynamic -  F F Overlay1
* 3151 aaaa.1111.1234 dynamic 50 F F Eth8/3
```

OTV ARP Cache Monitoring

OTV maintains an ARP cache to proxy ARP for IP addresses that it learned across the OTV interface.

```
cluster-N7k6-OTV# show otv arp-nd-cache
OTV ARP/ND L3->L2 Address Mapping Cache

Overlay Interface Overlay1
VLAN MAC Address Layer-3 Address Age Expires In
3151 0050.5660.9412 10.0.0.2 1w0d 00:00:31
cluster-N7k6-OTV#
```

Examples for Inter-Site Clustering

The following examples show supported cluster deployments.

Spanned EtherChannel Routed Mode Example with Site-Specific MAC and IP Addresses

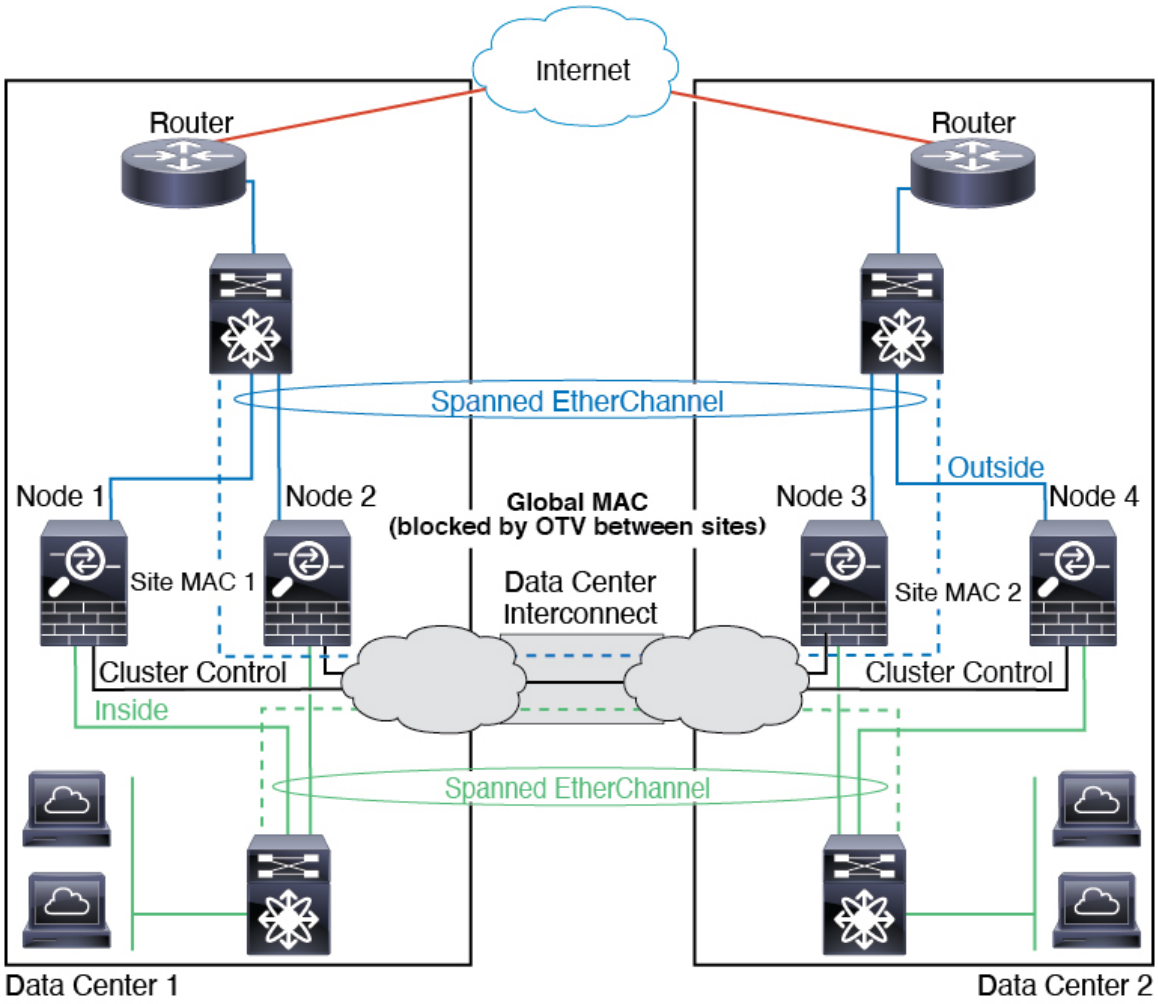
The following example shows 2 cluster members at each of 2 data centers placed between the gateway router and an inside network at each site (East-West insertion). The cluster members are connected by the cluster control link over the DCI. The cluster members at each site connect to the local switches using spanned EtherChannels for both the inside and outside networks. Each EtherChannel is spanned across all chassis in the cluster.

The data VLANs are extended between the sites using Overlay Transport Virtualization (OTV) (or something similar). You must add filters blocking the global MAC address to prevent traffic from traversing the DCI to the other site when the traffic is destined for the cluster. If the cluster nodes at one site become unreachable, you must remove the filters so traffic can be sent to the other site's cluster nodes. You should use VACLs to filter the global MAC address.For some switches, such as Nexus with the F3-series line card, you must also use ARP inspection to block ARP packets from the global MAC address. ARP inspection requires you to set both the site MAC address and the site IP address on the ASA. If you only configure the site MAC address be sure to disable ARP inspection.

The cluster acts as the gateway for the inside networks. The global virtual MAC, which is shared across all cluster nodes, is used only to receive packets. Outgoing packets use a site-specific MAC address from each DC cluster. This feature prevents the switches from learning the same global MAC address from both sites on two different ports, which causes MAC flapping; instead, they only learn the site MAC address.

In this scenario:

- All egress packets sent from the cluster use the site MAC address and are localized at the data center.
- All ingress packets to the cluster are sent using the global MAC address, so they can be received by any of the nodes at both sites; filters at the OTV localize the traffic within the data center.



Spanned EtherChannel Transparent Mode North-South Inter-Site Example

The following example shows 2 cluster members at each of 2 data centers placed between inside and outside routers (North-South insertion). The cluster members are connected by the cluster control link over the DCI. The cluster members at each site connect to the local switches using spanned EtherChannels for the inside and outside. Each EtherChannel is spanned across all chassis in the cluster.

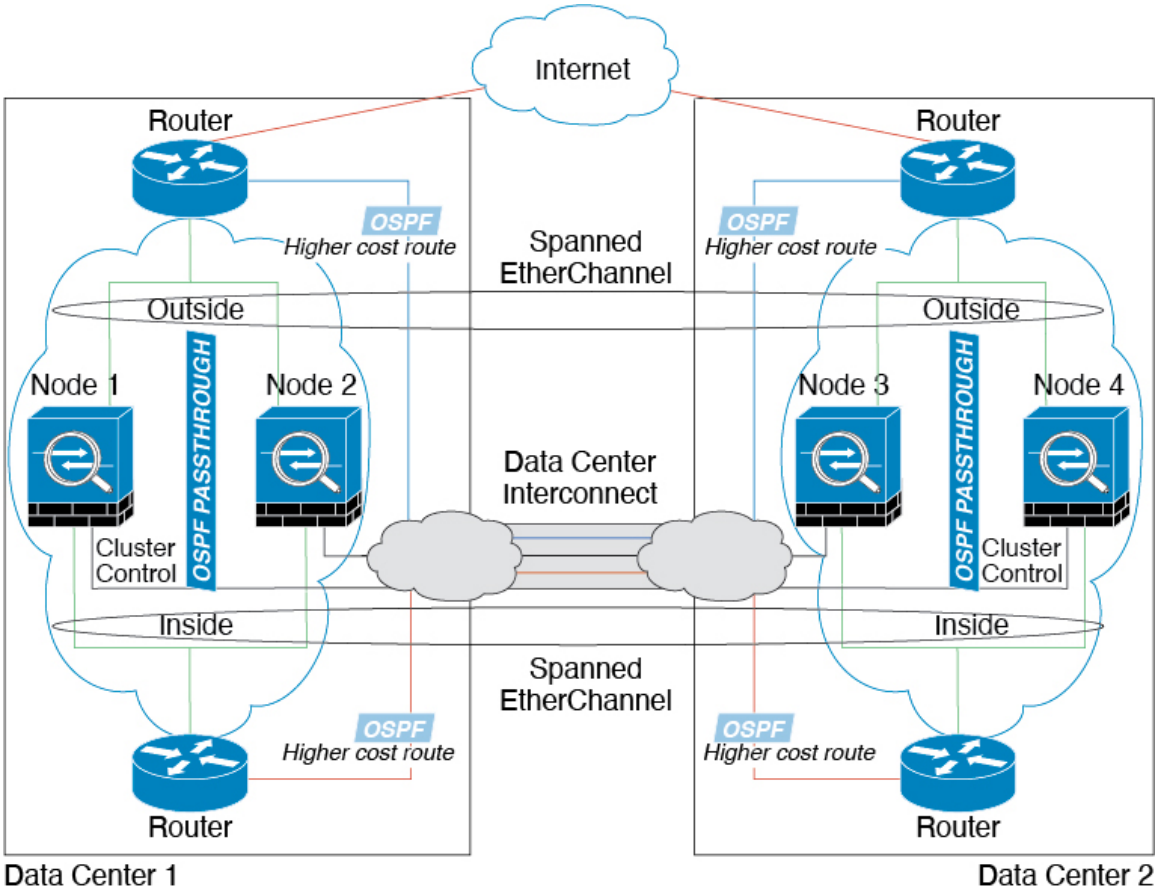
The inside and outside routers at each data center use OSPF, which is passed through the transparent ASAs. Unlike MACs, router IPs are unique on all routers. By assigning a higher cost route across the DCI, traffic stays within each data center unless all cluster members at a given site go down. The lower cost route through the ASAs must traverse the same bridge group at each site for the cluster to maintain asymmetric connections. In the event of a failure of all cluster members at one site, traffic goes from each router over the DCI to the cluster members at the other site.

The implementation of the switches at each site can include:

- Inter-site VSS, vPC, StackWise, or StackWise Virtual—In this scenario, you install one switch at Data Center 1, and the other at Data Center 2. One option is for the cluster nodes at each Data Center to only connect to the local switch, while the redundant switch traffic goes across the DCI. In this case, connections are for the most part kept local to each datacenter. You can optionally connect each node to both switches

across the DCI if the DCI can handle the extra traffic. In this case, traffic is distributed across the data centers, so it is essential for the DCI to be very robust.

- Local VSS, vPC, StackWise, or StackWise Virtual at each site—For better switch redundancy, you can install 2 separate redundant switch pairs at each site. In this case, although the cluster nodes still have a spanned EtherChannel with Data Center 1 chassis connected only to both local switches, and Data Center 2 chassis connected to those local switches, the spanned EtherChannel is essentially "split." Each local redundant switch system sees the spanned EtherChannel as a site-local EtherChannel.

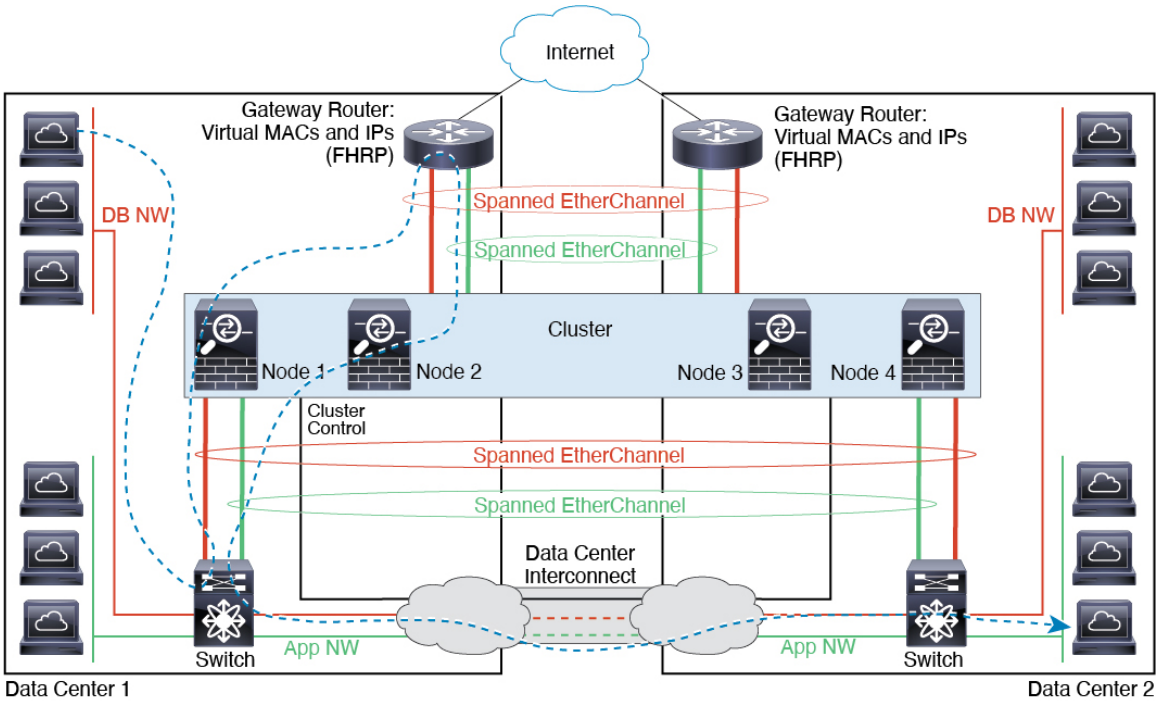


Spanned EtherChannel Transparent Mode East-West Inter-Site Example

The following example shows 2 cluster members at each of 2 data centers placed between the gateway router and two inside networks at each site, the App network and the DB network (East-West insertion). The cluster members are connected by the cluster control link over the DCI. The cluster members at each site connect to the local switches using spanned EtherChannels for both the App and DB networks on the inside and outside. Each EtherChannel is spanned across all chassis in the cluster.

The gateway router at each site uses an FHRP such as HSRP to provide the same destination virtual MAC and IP addresses at each site. A good practice to avoid unintended MAC address flapping is to statically add the gateway routers real MAC addresses to the ASA MAC address table using the **mac-address-table static** *outside_interface mac_address* command. Without these entries, if the gateway at site 1 communicates with the gateway at site 2, that traffic might pass through the ASA and attempt to reach site 2 from the inside interface and cause problems. The data VLANs are extended between the sites using Overlay Transport

Virtualization (OTV) (or something similar). You must add filters to prevent traffic from traversing the DCI to the other site when the traffic is destined for the gateway router. If the gateway router at one site becomes unreachable, you must remove the filters so traffic can be sent to the other site's gateway router.



Reference for Clustering

This section includes more information about how clustering operates.

ASA Features and Clustering

Some ASA features are not supported with ASA clustering, and some are only supported on the control node. Other features might have caveats for proper usage.

Unsupported Features with Clustering

These features cannot be configured with clustering enabled, and the commands will be rejected.

- Unified Communication features that rely on TLS Proxy
- Remote access VPN (SSL VPN and IPsec VPN)
- Virtual Tunnel Interfaces (VTIs)
- The following application inspections:
  - CTIQBE
  - H323, H225, and RAS

    - IPsec passthrough
    - MGCP
    - MMP
    - RTSP
    - SCCP (Skinny)
    - WAAS
    - WCCP
- Botnet Traffic Filter
- Auto Update Server
- DHCP client, server, and proxy. DHCP relay is supported.
- VPN load balancing
- Failover
- Integrated Routing and Bridging
- FIPS mode

Centralized features for clustering

These features are only supported on the control node, and are not scaled for the cluster.



Note Traffic for centralized features is forwarded from member nodes to the control node over the cluster control link.

If you use the rebalancing feature, traffic for centralized features may be rebalanced to non-control nodes before the traffic is classified as a centralized feature; if this occurs, the traffic is then sent back to the control node.

For centralized features, if the control node fails, all connections are dropped, and you have to re-establish the connections on the new control node.

- These application inspections are supported:
    - DCERPC and ESMTP
    - IM
    - NetBIOS and PPTP
    - RADIUS
    - RSH
    - SNMP
    - SQLNET and SUNRPC

  - TFTP
  - XDMCP
- Static route monitoring
- Authentication and Authorization for network access. Accounting is decentralized.
- Filtering Services
- Site-to-site VPN
- IGMP multicast control plane protocol processing (data plane forwarding is distributed across the cluster)
- PIM multicast control plane protocol processing (data plane forwarding is distributed across the cluster)
- Dynamic routing

Features Applied to Individual Nodes

These features are applied to each ASA node, instead of the cluster as a whole or to the control node.

- QoS—The QoS policy is synced across the cluster as part of configuration replication. However, the policy is enforced on each node independently. For example, if you configure policing on output, then the conform rate and conform burst values are enforced on traffic exiting a particular ASA. In a cluster with 3 nodes and with traffic evenly distributed, the conform rate actually becomes 3 times the rate for the cluster.
- Threat detection—Threat detection works on each node independently; for example, the top statistics is node-specific. Port scanning detection, for example, does not work because scanning traffic will be load-balanced between all nodes, and one node will not see all traffic.
- Resource management—Resource management in multiple context mode is enforced separately on each node based on local usage.
- LISP traffic—LISP traffic on UDP port 4342 is inspected by each receiving node, but is not assigned a director. Each node adds to the EID table that is shared across the cluster, but the LISP traffic itself does not participate in cluster state sharing.

AAA for Network Access and Clustering

AAA for network access consists of three components: authentication, authorization, and accounting. Authentication and authorization are implemented as centralized features on the clustering control node with replication of the data structures to the cluster data nodes. If a control node is elected, the new control node will have all the information it needs to continue uninterrupted operation of the established authenticated users and their associated authorizations. Idle and absolute timeouts for user authentications are preserved when a control node change occurs.

Accounting is implemented as a distributed feature in a cluster. Accounting is done on a per-flow basis, so the cluster node owning a flow will send accounting start and stop messages to the AAA server when accounting is configured for a flow.

Connection settings and clustering

Connection limits are enforced cluster-wide (see the **set connection conn-max**, **set connection embryonic-conn-max**, **set connection per-client-embryonic-max**, and **set connection per-client-max** commands). Each node has an estimate of the cluster-wide counter values based on broadcast messages. Due to efficiency considerations, the configured connection limit across the cluster might not be enforced exactly at the limit number. Each node may overestimate or underestimate the cluster-wide counter value at any given time. However, the information will get updated over time in a load-balanced cluster.

FTP and Clustering

- If FTP data channel and control channel flows are owned by different cluster members, then the data channel owner will periodically send idle timeout updates to the control channel owner and update the idle timeout value. However, if the control flow owner is reloaded, and the control flow is re-hosted, the parent/child flow relationship will not longer be maintained; the control flow idle timeout will not be updated.
- If you use AAA for FTP access, then the control channel flow is centralized on the control node.

ICMP Inspection and Clustering

The flow of ICMP and ICMP error packets through the cluster varies depending on whether ICMP/ICMP error inspection is enabled. Without ICMP inspection, ICMP is a one-direction flow, and there is no director flow support. With ICMP inspection, the ICMP flow becomes two-directional and is backed up by a director/backup flow. One difference for an inspected ICMP flow is in the director handling of a forwarded packet: the director will forward the ICMP echo reply packet to the flow owner instead of returning the packet to the forwarder.

Multicast Routing and Clustering

Multicast Routing in Spanned EtherChannel Mode

In Spanned EtherChannel mode: The control unit handles all multicast routing packets and data packets until fast-path forwarding is established. After the connection is established, each data unit can forward multicast data packets.

NAT and clustering

NAT can impact the overall throughput of the cluster. Inbound and outbound NAT packets can be sent to different devices in the cluster because the load balancing algorithm relies on IP addresses and ports, and NAT causes inbound and outbound packets to have different IP addresses and/or ports. When a packet arrives at the device that is not the NAT owner, it is forwarded over the cluster control link to the owner, causing large amounts of traffic on the cluster control link.

Note that the receiving node does not create a forwarding flow to the owner, because the NAT owner may not end up creating a connection for the packet depending on the results of security and policy checks.

Dynamic Routing and Clustering

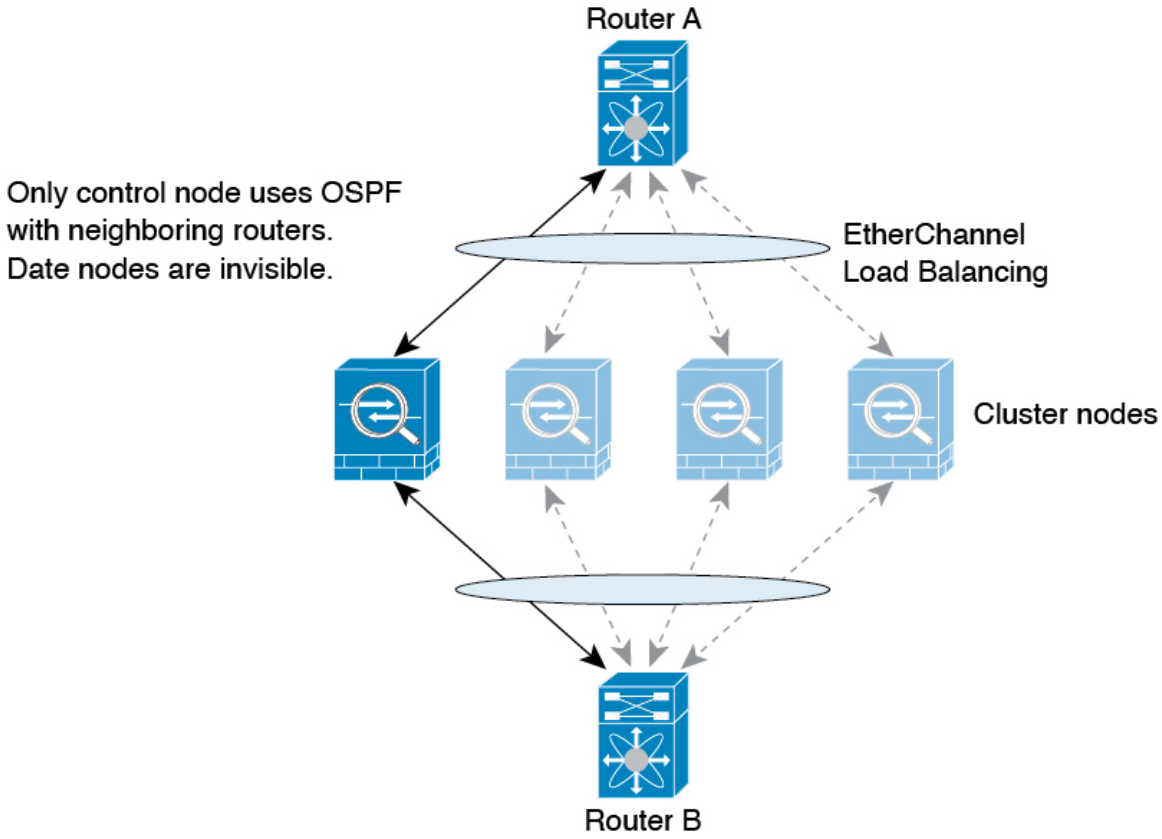
This section describes how to use dynamic routing with clustering.

Dynamic routing in spanned EtherChannel mode

The routing process only runs on the control node, and routes are learned through the control node and replicated to data nodes. If a routing packet arrives at a data node, it is redirected to the control node.

Note IS-IS is not supported in Spanned EtherChannel mode.

Figure 47: Dynamic routing in spanned EtherChannel mode



After the data node learn the routes from the control node, each node makes forwarding decisions independently.

The OSPF LSA database is not synchronized from the control node to data nodes. If there is a control node switchover, the neighboring router will detect a restart; the switchover is not transparent. The OSPF process picks an IP address as its router ID. Although not required, you can assign a static router ID to ensure a consistent router ID is used across the cluster. See the OSPF Non-Stop Forwarding feature to address the interruption.

SCTP and Clustering

An SCTP association can be created on any node (due to load balancing); its multi-homing connections must reside on the same node.

SIP inspection and clustering

A control flow can be created on any node (due to load balancing); its child data flows must reside on the same node.

TLS Proxy configuration is not supported.

SNMP and clustering

You should always use the local address, and not the main cluster IP address for SNMP polling. If the SNMP agent polls the Main cluster IP address, if a new control node is elected, the poll to the new control node will fail.

When using SNMPv3 with clustering, if you add a new cluster node after the initial cluster formation, then SNMPv3 users are not replicated to the new node.You must re-add them on the control node to force the users to replicate to the new node, or directly on the data node.

STUN and Clustering

STUN inspection is supported in failover and cluster modes, as pinholes are replicated. However, the transaction ID is not replicated among nodes. In the case where a node fails after receiving a STUN Request and another node received the STUN Response, the STUN Response will be dropped.

Syslog and NetFlow and clustering

Syslog and NetFlow and clustering is a network management configuration that enables logging and flow monitoring functionality within clustered network environments.

- Syslog—Each node in the cluster generates its own syslog messages. You can configure logging so that each node uses either the same or a different device ID in the syslog message header field. For example, the hostname configuration is replicated and shared by all nodes in the cluster. If you configure logging to use the hostname as the device ID, syslog messages generated by all nodes look as if they come from a single node. If you configure logging to use the local-node name that is assigned in the cluster bootstrap configuration as the device ID, syslog messages look as if they come from different nodes.
- NetFlow—Each node in the cluster generates its own NetFlow stream. The NetFlow collector can only treat each ASA as a separate NetFlow exporter.

Cisco TrustSec and clustering

Only the control node learns security group tag (SGT) information. The control node then populates the SGT to data nodes, and data nodes can make a match decision for SGT based on the security policy.

VPN and clustering

Site-to-site VPN is a centralized feature; only the control node supports VPN connections.

Note Remote access VPN is not supported with clustering.

VPN functionality is limited to the control node and does not take advantage of the cluster high availability capabilities. If the control node fails, all existing VPN connections are lost, and VPN users will see a disruption in service. When a new control node is elected, you must reestablish the VPN connections.

When you connect a VPN tunnel to a Spanned EtherChannel address, connections are automatically forwarded to the control node.

VPN-related keys and certificates are replicated to all nodes.

Performance scaling factor

When you combine multiple units into a cluster, you can expect the total cluster performance to be approximately 80% of the maximum combined throughput.

Throughput calculation

For example, if your model can handle approximately 10 Gbps of traffic when running alone, then for a cluster of 8 units, the maximum combined throughput will be approximately 80% of 80 Gbps (8 units x 10 Gbps): 64 Gbps.

Control node election

Summary

The key components involved in the control node election process are:

- Cluster nodes: Communicate over the cluster control link and participate in the election process
- Election requests: Broadcast every 3 seconds by nodes when clustering is enabled
- Priority values: Set between 1 and 100 where 1 is the highest priority
- Control node: The elected node that coordinates cluster operations

Workflow

These stages describe how cluster nodes elect a control node:

1. When clustering is enabled for a node (or when it first starts up with clustering already enabled), the node broadcasts an election request every 3 seconds.
2. Any other nodes with a higher priority respond to the election request. The priority is set between 1 and 100, where 1 is the highest priority.
3. If after 45 seconds, a node does not receive a response from another node with a higher priority, then it becomes the control node.



Note If multiple nodes tie for the highest priority, the cluster node name and then the serial number is used to determine the control node.

4. If a node later joins the cluster with a higher priority, it does not automatically become the control node. The existing control node always remains as the control node unless it stops responding, at which point a new control node is elected.
5. In a "split brain" scenario when there are temporarily multiple control nodes, the node with highest priority retains the role while the other nodes return to data node roles.



Note You can manually force a node to become the control node. For centralized features, if you force a control node change, then all connections are dropped, and you have to re-establish the connections on the new control node.

High availability within clusters

Clustering provides high availability by monitoring node and interface health and by replicating connection states between nodes.

Node health monitoring

Each node periodically sends a broadcast heartbeat packet over the cluster control link. If the control node does not receive any heartbeat packets or other packets from a data node within the configurable timeout period, then the control node removes the data node from the cluster. If the data nodes do not receive packets from the control node, then a new control node is elected from the remaining nodes.

Split brain scenario

If nodes cannot reach each other over the cluster control link because of a network failure and not because a node has actually failed, then the cluster may go into a "split brain" scenario where isolated data nodes will elect their own control nodes. For example, if a router fails between two cluster locations, then the original control node at location 1 will remove the location 2 data nodes from the cluster. Meanwhile, the nodes at location 2 will elect their own control node and form their own cluster. Note that asymmetric traffic may fail in this scenario. After the cluster control link is restored, then the control node that has the higher priority will keep the control node's role.

Interface monitoring

Each node monitors the link status of all named hardware interfaces in use, and reports status changes to the control node.

- **Spanned EtherChannel**—Uses cluster Link Aggregation Control Protocol (cLACP). Each node monitors the link status and the cLACP protocol messages to determine if the port is still active in the EtherChannel. The status is reported to the control node.

When you enable health monitoring, all physical interfaces (including the main EtherChannel) are monitored by default; you can optionally disable monitoring per interface. Only named interfaces can be monitored. For example, the named EtherChannel must fail to be considered failed, which means all member ports of an EtherChannel must fail to trigger cluster removal (depending on your minimum port bundling setting).

A node is removed from the cluster if its monitored interfaces fail. The amount of time before the ASA removes a member from the cluster depends on whether the node is an established member or is joining the cluster. If the interface is down on an established member, then the ASA removes the member after 9 seconds. The ASA

does not monitor interfaces for the first 90 seconds that a node joins the cluster. Interface status changes during this time will not cause the ASA to be removed from the cluster. For non-EtherChannels, the node is removed after 500 ms, regardless of the member state.

Status after failure

When a node in the cluster fails, the connections hosted by that node are seamlessly transferred to other nodes; state information for traffic flows is shared over the control node's cluster control link.

If the control node fails, then another member of the cluster with the highest priority (lowest number) becomes the control node.

The device automatically tries to rejoin the cluster, depending on the failure event.



Note When the device becomes inactive and fails to automatically rejoin the cluster, all data interfaces are shut down; only the management-only interface can send and receive traffic. The management interface remains up using the IP address the node received from the cluster IP pool. However if you reload, and the node is still inactive in the cluster, the management interface is disabled. You must use the console port for any further configuration.

Rejoining the Cluster

After a cluster node is removed from the cluster, how it can rejoin the cluster depends on why it was removed:

- Failed cluster control link when initially joining—After you resolve the problem with the cluster control link, you must manually rejoin the cluster by re-enabling clustering at the console port by entering **cluster group** *name*, and then **enable**.
- Failed cluster control link after joining the cluster—The ASA automatically tries to rejoin every 5 minutes, indefinitely. This behavior is configurable.
- Failed data interface—The ASA automatically tries to rejoin at 5 minutes, then at 10 minutes, and finally at 20 minutes. If the join is not successful after 20 minutes, then the ASA disables clustering. After you resolve the problem with the data interface, you have to manually enable clustering at the console port by entering **cluster group** *name*, and then **enable**. This behavior is configurable.
- Failed node—If the node was removed from the cluster because of a node health check failure, then rejoining the cluster depends on the source of the failure. For example, a temporary power failure means the node will rejoin the cluster when it starts up again as long as the cluster control link is up and clustering is still enabled with the **enable** command. The ASA attempts to rejoin the cluster every 5 seconds.
- Internal error—Internal failures include: application sync timeout; inconsistent application statuses; and so on. A node will attempt to rejoin the cluster automatically at the following intervals: 5 minutes, 10 minutes, and then 20 minutes. This behavior is configurable.

See Configure the Control Node Bootstrap Settings, on page 359.

Data path connection state replication

Every connection has one owner and at least one backup owner in the cluster. The backup owner does not take over the connection in the event of a failure; instead, it stores TCP/UDP state information, so that the

connection can be seamlessly transferred to a new owner in case of a failure. The backup owner is usually also the director.

Some traffic requires state information above the TCP or UDP layer. See the following table for clustering support or lack of support for this kind of traffic.

Table 18: Features replicated across the cluster

Traffic	State Support	Notes
Up time	Yes	Keeps track of the system up time.
ARP Table	Yes	—
MAC address table	Yes	—
User Identity	Yes	Includes AAA rules (uauth).
IPv6 Neighbor database	Yes	—
Dynamic routing	Yes	—
SNMP Engine ID	No	—
Distributed VPN (Site-to-Site)	Yes	Backup session becomes the active session, then a new backup session is created.

How the cluster manages connections

Connections can be load-balanced to multiple nodes of the cluster. Connection roles determine how connections are handled in both normal operation and in a high availability situation.

Connection roles

These roles are defined for each connection:

- **Owner**—Usually, the node that initially receives the connection. The owner maintains the TCP state and processes packets. A connection has only one owner. If the original owner fails, then when new nodes receive packets from the connection, the director chooses a new owner from those nodes.
- **Backup owner**—The node that stores TCP/UDP state information received from the owner, so that the connection can be seamlessly transferred to a new owner in case of a failure. The backup owner does not take over the connection in the event of a failure. If the owner becomes unavailable, then the first node to receive packets from the connection (based on load balancing) contacts the backup owner for the relevant state information so it can become the new owner.

  As long as the director (see below) is not the same node as the owner, then the director is also the backup owner. If the owner chooses itself as the director, then a separate backup owner is chosen.

  For clustering on the Firepower 9300, which can include up to 3 cluster nodes in one chassis, if the backup owner is on the same chassis as the owner, then an additional backup owner will be chosen from another chassis to protect flows from a chassis failure.

  If you enable director localization for inter-site clustering, then there are two backup owner roles: the local backup and the global backup. The owner always chooses a local backup at the same site as itself

(based on site ID). The global backup can be at any site, and might even be the same node as the local backup. The owner sends connection state information to both backups.

If you enable site redundancy, and the backup owner is at the same site as the owner, then an additional backup owner will be chosen from another site to protect flows from a site failure. Chassis backup and site backup are independent, so in some cases a flow will have both a chassis backup and a site backup.

- **Director**—The node that handles owner lookup requests from forwarders. When the owner receives a new connection, it chooses a director based on a hash of the source/destination IP address and ports (see below for ICMP hash details), and sends a message to the director to register the new connection. If packets arrive at any node other than the owner, the node queries the director about which node is the owner so it can forward the packets. A connection has only one director. If a director fails, the owner chooses a new director.

  As long as the director is not the same node as the owner, then the director is also the backup owner (see above). If the owner chooses itself as the director, then a separate backup owner is chosen.

  If you enable director localization for inter-site clustering, the cluster uses two director roles: the local director and the global director. For a new-flow packet, the receiving member selects the local director from members at the receiving member's site. The global director can be at any site and might be the same member as the local director. If the original owner fails and localization applies, the local director chooses a new connection owner at the local director's site.

  ICMP/ICMPv6 hash details:

  - For Echo packets, the source port is the ICMP identifier, and the destination port is 0.
  - For Reply packets, the source port is 0, and the destination port is the ICMP identifier.
  - For other packets, both source and destination ports are 0.

- **Forwarder**—A node that forwards packets to the owner. If a forwarder receives a packet for a connection it does not own, it queries the director for the owner, and then establishes a flow to the owner for any other packets it receives for this connection. The director can also be a forwarder. If you enable director localization, the forwarder queries the local director when localization applies. The forwarder queries the global director if the local director does not know the owner, if the connection is owned at another site, or if localization is bypassed. A cross-site director query is expected behavior in these cases. Note that if a forwarder receives the SYN-ACK packet, it can derive the owner directly from a SYN cookie in the packet, so it does not need to query the director. (If you disable TCP sequence randomization, the SYN cookie is not used; a query to the director is required.) For short-lived flows such as DNS and ICMP, instead of querying, the forwarder immediately sends the packet to the director, which then sends them to the owner. A connection can have multiple forwarders; the most efficient throughput is achieved by a good load-balancing method where there are no forwarders and all packets of a connection are received by the owner.



Note We do not recommend disabling TCP sequence randomization when using clustering. There is a small chance that some TCP sessions won't be established, because the SYN/ACK packet might be dropped.

- **Fragment Owner**—For fragmented packets, cluster nodes that receive a fragment determine a fragment owner using a hash of the fragment source IP address, destination IP address, and the packet ID. All fragments are then forwarded to the fragment owner over the cluster control link. Fragments may be load-balanced to different cluster nodes, because only the first fragment includes the 5-tuple used in the

switch load balance hash. Other fragments do not contain the source and destination ports and may be load-balanced to other cluster nodes. The fragment owner temporarily reassembles the packet so it can determine the director based on a hash of the source/destination IP address and ports. If it is a new connection, the fragment owner will register to be the connection owner. If it is an existing connection, the fragment owner forwards all fragments to the provided connection owner over the cluster control link. The connection owner will then reassemble all fragments.

Port address translation connections

When a connection uses Port Address Translation (PAT), then the PAT type (per-session or multi-session) influences which member of the cluster becomes the owner of a new connection:

- Per-session PAT—The owner is the node that receives the initial packet in the connection.

  By default, TCP and DNS UDP traffic use per-session PAT.

- Multi-session PAT—The owner is always the control node. If a multi-session PAT connection is initially received by a data node, then the data node forwards the connection to the control node.

  By default, UDP (except for DNS UDP) and ICMP traffic use multi-session PAT, so these connections are always owned by the control node.

You can change the per-session PAT defaults for TCP and UDP so connections for these protocols are handled per-session or multi-session depending on the configuration. For ICMP, you cannot change from the default multi-session PAT. For more information about per-session PAT, see the firewall configuration guide.

New connection ownership

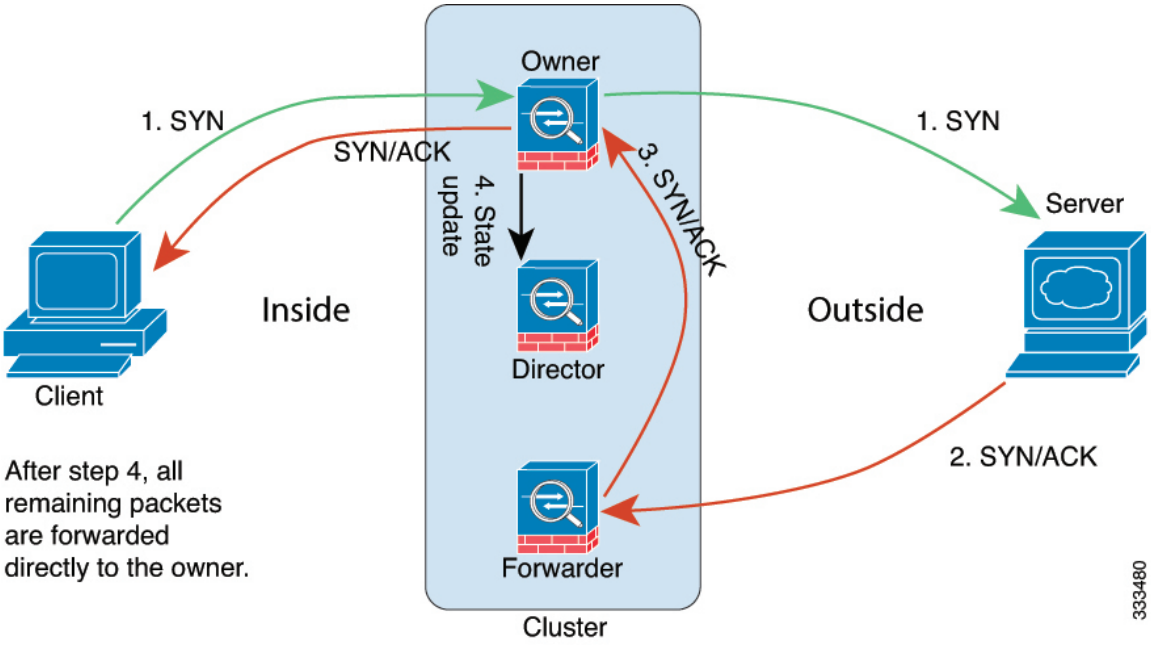
When a new connection is directed to a node of the cluster via load balancing, that node owns both directions of the connection. If any connection packets arrive at a different node, they are forwarded to the owner node over the cluster control link. For best performance, proper external load balancing is required for both directions of a flow to arrive at the same node, and for flows to be distributed evenly between nodes. If a reverse flow arrives at a different node, it is redirected back to the original node.

Sample data flow for TCP

Summary

This example shows the establishment of a new connection.

Workflow



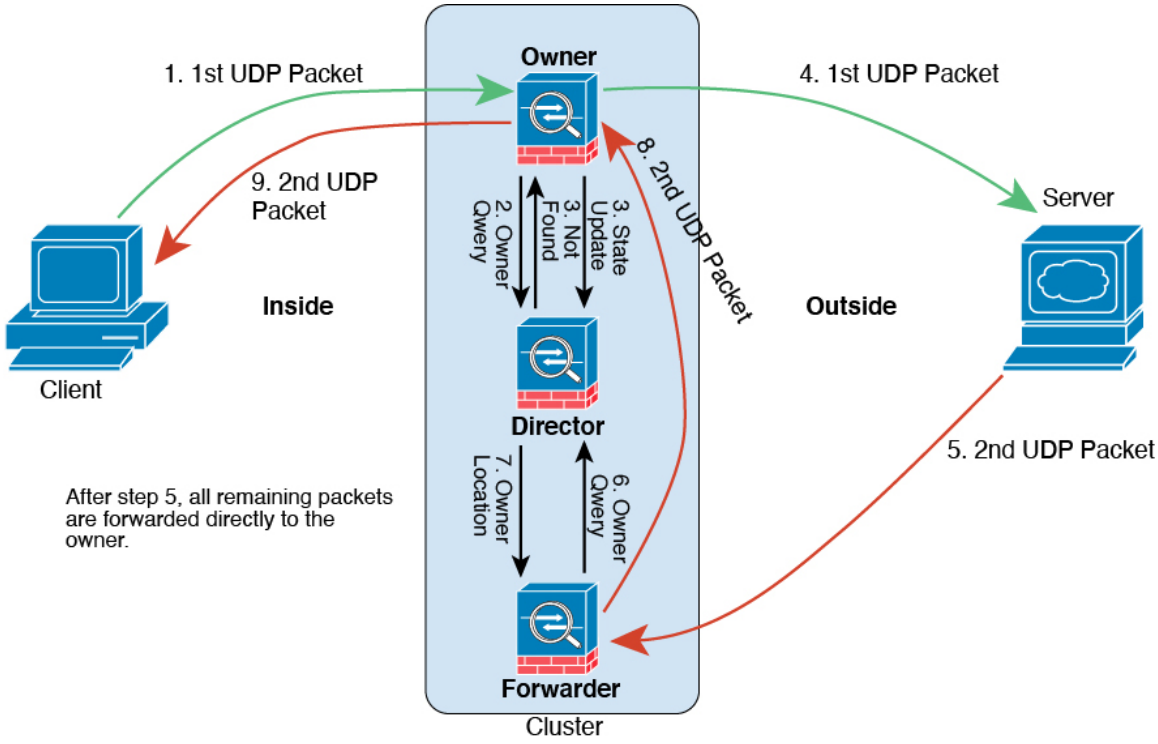
1. The SYN packet originates from the client and is delivered to one device (based on the load balancing method), which becomes the owner. The owner creates a flow, encodes owner information into a SYN cookie, and forwards the packet to the server.
2. The SYN-ACK packet originates from the server and is delivered to a different device (based on the load balancing method). This device is the forwarder.
3. Because the forwarder does not own the connection, it decodes owner information from the SYN cookie, creates a forwarding flow to the owner, and forwards the SYN-ACK to the owner.
4. The owner sends a state update to the director, and forwards the SYN-ACK to the client.
5. The director receives the state update from the owner, creates a flow to the owner, and records the TCP state information as well as the owner. The director acts as the backup owner for the connection.
6. Any subsequent packets delivered to the forwarder will be forwarded to the owner.
7. If packets are delivered to any additional nodes, it will query the director for the owner and establish a flow.
8. Any state change for the flow results in a state update from the owner to the director.

Sample data flows for ICMP and UDP

Summary

This example shows the establishment of a new connection.

Workflow



1. The first UDP packet originates from the client and is delivered to one device (based on the load balancing method).
2. The node that received the first packet queries the director node that is chosen based on a hash of the source/destination IP address and ports.
3. The director finds no existing flow, creates a director flow and forwards the packet back to the previous node. In other words, the director has elected an owner for this flow.
4. The owner creates the flow, sends a state update to the director, and forwards the packet to the server.
5. The second UDP packet originates from the server and is delivered to the forwarder.
6. The forwarder queries the director for ownership information. For short-lived flows such as DNS, instead of querying, the forwarder immediately sends the packet to the director, which then sends it to the owner.
7. The director replies to the forwarder with ownership information.
8. The forwarder creates a forwarding flow to record owner information and forwards the packet to the owner.
9. The owner forwards the packet to the client.

Rebalancing New TCP Connections Across the Cluster

If the load balancing capabilities of the upstream or downstream routers result in unbalanced flow distribution, you can configure new connection rebalancing so nodes with higher new connections per second will redirect new TCP flows to other nodes. No existing flows will be moved to other nodes.

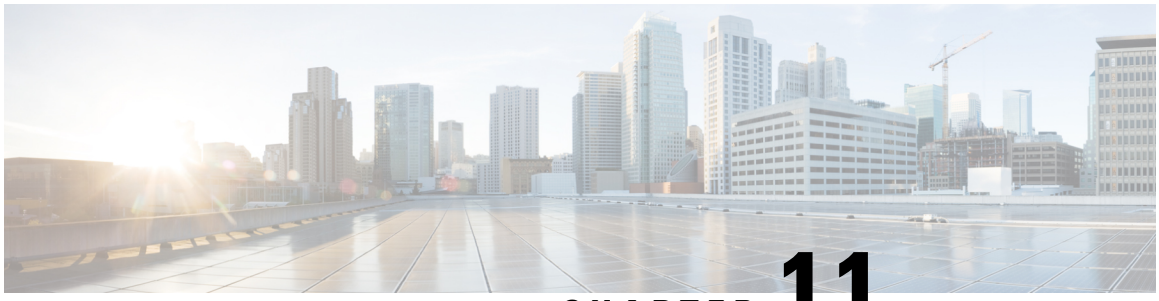
Because this command only rebalances based on connections per second, the total number of established connections on each node is not considered, and the total number of connections may not be equal.

Once a connection is offloaded to a different node, it becomes an asymmetric connection.

Do not configure connection rebalancing for inter-site topologies; you do not want new connections rebalanced to cluster members at a different site.

History for ASA Clustering for the Secure Firewall 3100

Feature Name	Version	Feature Information
Distributed site-to-site VPN with clustering on the Secure Firewall 4200	9.23(1)	An ASA cluster on the Secure Firewall 4200 supports site-to-site VPN in distributed mode. Distributed mode provides the ability to have many site-to-site IPsec IKEv2 VPN connections distributed across members of an ASA cluster, not just on the control node (as in centralized mode). This significantly scales VPN support beyond centralized VPN capabilities and provides high availability. Added/modified commands: **cluster redistribute vpn-sessiondb**, **show cluster vpn-sessiondb**, **vpn-mode** , **show cluster resource usage**, **show vpn-sessiondb** , **show conn detail**, **show crypto ikev2 stats**
Support for clustering on the Secure Firewall 3100 was introduced	9.17(1)	You can cluster up to 8 Secure Firewall 3100 nodes in Spanned EtherChannel mode.



CHAPTER 11

ASA Cluster for the Firepower 4100/9300

Clustering lets you group multiple Firepower 4100/9300 chassis ASAs together as a single logical device. The Firepower 4100/9300 chassis series includes the Firepower 9300 and Firepower 4100 series. A cluster provides all the convenience of a single device (management, integration into a network) while achieving the increased throughput and redundancy of multiple devices.

Note Some features are not supported when using clustering. See Unsupported Features with Clustering, on page 500.

- About Clustering on the Firepower 4100/9300 Chassis, on page 423
- Requirements and Prerequisites for Clustering on the Firepower 4100/9300 Chassis, on page 430
- Licenses for Clustering on the Firepower 4100/9300 Chassis, on page 431
- Clustering Guidelines and Limitations, on page 433
- Configure Clustering on the Firepower 4100/9300 Chassis, on page 437
- FXOS: Remove a Cluster Node, on page 471
- ASA: Manage Cluster Members, on page 472
- ASA: Monitoring the ASA Cluster on the Firepower 4100/9300 chassis, on page 477
- Troubleshooting Distributed S2S VPN, on page 487
- Examples for ASA Clustering, on page 489
- Reference for Clustering, on page 500
- History for ASA Clustering on the Firepower 4100/9300, on page 516

About Clustering on the Firepower 4100/9300 Chassis

When you deploy a cluster on the Firepower 4100/9300 chassis, it does the following:

- Creates a *cluster-control link* (by default, port-channel 48) for node-to-node communication.

  For a cluster isolated to security modules within one Firepower 9300 chassis, this link utilizes the Firepower 9300 backplane for cluster communications.

  For clustering with multiple chassis, you need to manually assign physical interface(s) to this EtherChannel for communications between chassis.

- Creates the cluster bootstrap configuration within the application.

When you deploy the cluster, the chassis supervisor pushes a minimal bootstrap configuration to each unit that includes the cluster name, cluster control link interface, and other cluster settings. Some parts of the bootstrap configuration may be user-configurable within the application if you want to customize your clustering environment.

- Assigns data interfaces to the cluster as *Spanned* interfaces.

  For a cluster isolated to security modules within one Firepower 9300 chassis, spanned interfaces are not limited to EtherChannels, like it is for clustering with multiple chassis. The Firepower 9300 supervisor uses EtherChannel technology internally to load-balance traffic to multiple modules on a shared interface, so any data interface type works for Spanned mode. For clustering with multiple chassis, you must use Spanned EtherChannels for all data interfaces.



Note Individual interfaces are not supported, with the exception of a management interface.

- Assigns a management interface to all units in the cluster.

See the following sections for more information about clustering.

Bootstrap Configuration

When you deploy the cluster, the Firepower 4100/9300 chassis supervisor pushes a minimal bootstrap configuration to each unit that includes the cluster name, cluster control link interface, and other cluster settings. Some parts of the bootstrap configuration are user-configurable if you want to customize your clustering environment.

Cluster Members

Cluster members work together to accomplish the sharing of the security policy and traffic flows.

One member of the cluster is the **control** unit. The control unit is determined automatically. All other members are **data** units.

You must perform all configuration on the control unit only; the configuration is then replicated to the data units.

Some features do not scale in a cluster, and the control unit handles all traffic for those features. See Centralized Features for Clustering, on page 501.

Cluster Control Link

The cluster-control link is an EtherChannel (port-channel 48) for unit-to-unit communication. For intra-chassis clustering, this link utilizes the Firepower 9300 backplane for cluster communications. For inter-chassis clustering, you need to manually assign physical interface(s) to this EtherChannel on the Firepower 4100/9300 chassis for communications between chassis.

For a 2-chassis inter-chassis cluster, do not directly-connect the cluster control link from one chassis to the other chassis. If you directly connect the interfaces, then when one unit fails, the cluster control link fails, and

thus the remaining healthy unit fails. If you connect the cluster control link through a switch, then the cluster control link remains up for the healthy unit.

Cluster control link traffic includes both control and data traffic.

Control traffic includes:

- Control node election.
- Configuration replication.
- Health monitoring.

Data traffic includes:

- State replication.
- Connection ownership queries and data packet forwarding.

See the following sections for more information about the cluster control link.

Size the cluster control link

Cluster control link traffic is comprised mainly of state update and forwarded packets. The amount of traffic at any given time on the cluster control link varies. The amount of forwarded traffic depends on the load-balancing efficacy or whether there is a lot of traffic for centralized features.

For example:

- NAT results in poor load balancing of connections, and the need to rebalance all returning traffic to the correct units.
- AAA for network access is a centralized feature, so all traffic is forwarded to the control unit.
- When membership changes, the cluster needs to rebalance a large number of connections, thus temporarily using a large amount of cluster control link bandwidth.

A higher-bandwidth cluster control link helps the cluster to converge faster when there are membership changes and prevents throughput bottlenecks.

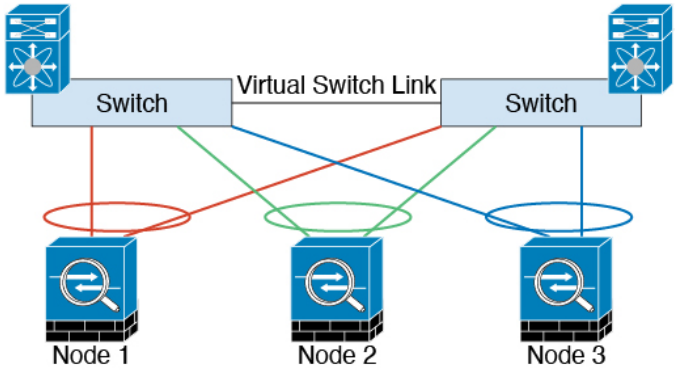


Note If your cluster has large amounts of asymmetric (rebalanced) traffic, then you should increase the cluster control link size.

Cluster control link redundancy

We recommend using an EtherChannel for the cluster control link, so that you can pass traffic on multiple links in the EtherChannel while still achieving redundancy.

The diagram shows how to use an EtherChannel as a cluster control link in a Virtual Switching System (VSS), Virtual Port Channel (vPC), StackWise, or StackWise Virtual environment. All links in the EtherChannel are active. When the switch is part of a redundant system, then you can connect firewall interfaces within the same EtherChannel to separate switches in the redundant system. The switch interfaces are members of the same EtherChannel port-channel interface, because the separate switches act like a single switch. Note that this EtherChannel is device-local, not a Spanned EtherChannel.



Cluster control link reliability

To ensure cluster control link functionality, be sure the round-trip time (RTT) between units is less than 20 ms. This maximum latency enhances compatibility with cluster members installed at different geographical sites. To check your latency, perform a ping on the cluster control link between units.

The cluster control link must be reliable, with no out-of-order or dropped packets; for example, for inter-site deployment, you should use a dedicated link.

Cluster Control Link Network

The Firepower 4100/9300 chassis auto-generates the cluster control link interface IP address for each unit based on the chassis ID and slot ID: 127.2.*chassis_id*.*slot_id*.You can customize this IP address when you deploy the cluster. The cluster control link network cannot include any routers between units; only Layer 2 switching is allowed. For inter-site traffic, Cisco recommends using Overlay Transport Virtualization (OTV).

Cluster control link MTU ping testing

Use Cluster control link MTU ping testing to

- verify MTU compatibility between cluster nodes during the join process
- use ping-based fragmentation tests to confirm the network path supports proper packet handling, and
- allow nodes to join even if tests fail while providing diagnostic messages for troubleshooting MTU mismatches.

When a node joins the cluster, the control node send a ping with a payload size of twice the MTU. This process tests the network's ability to handle packet fragmentation because the underlying IP layer will fragment packets that exceed the MTU limit.

A successful ping confirms that the network path supports proper fragmentation and that the cluster control link can reliably process traffic at the configured MTU size.

If the ping fails, view these messages:

- **show cluster history**—`Event: CCL MTU test to unit name failed`
- Console WARNING—`WARNING: Unit name is not reachable in CCL jumbo frame ICMP test, please check cluster interface and switch MTU configuration`

Even if the ping fails, the node is allowed to join the cluster. In this case, you need to resolve the MTU mismatch as soon as possible.

Cluster Interfaces

For a cluster isolated to security modules within one Firepower 9300 chassis, you can assign both physical interfaces or EtherChannels (also known as port channels) to the cluster. Interfaces assigned to the cluster are Spanned interfaces that load-balance traffic across all members of the cluster.

For clustering with multiple chassis, you can only assign data EtherChannels to the cluster. These Spanned EtherChannels include the same member interfaces on each chassis; on the upstream switch, all of these interfaces are included in a single EtherChannel, so the switch does not know that it is connected to multiple devices.

Individual interfaces are not supported, with the exception of a management interface.

Connecting to a Redundant Switch System

We recommend connecting EtherChannels to a redundant switch system such as a VSS, vPC, StackWise, or StackWise Virtual system to provide redundancy for your interfaces.

Configuration Replication

All nodes in the cluster share a single configuration. You can only make configuration changes on the control node (with the exception of the bootstrap configuration), and changes are automatically synced to all other nodes in the cluster.

ASA Cluster Management

One of the benefits of using ASA clustering is the ease of management. This section describes how to manage the cluster.

Management Network

We recommend connecting all units to a single management network. This network is separate from the cluster control link.

Management Interface

You must assign a Management type interface to the cluster. This interface is a special individual interface as opposed to a Spanned interface. The management interface lets you connect directly to each unit.

The Main cluster IP address is a fixed address for the cluster that always belongs to the current control unit. You also configure a range of addresses so that each unit, including the current control unit, can use a Local address from the range. The Main cluster IP address provides consistent management access to an address; when a control unit changes, the Main cluster IP address moves to the new control unit, so management of the cluster continues seamlessly.

For example, you can manage the cluster by connecting to the Main cluster IP address, which is always attached to the current control unit. To manage an individual member, you can connect to the Local IP address.

Note To-the-box traffic needs to be directed to the node's management IP address; to-the-box traffic is not forwarded over the cluster control link to any other node.

For outbound management traffic such as TFTP or syslog, each unit, including the control unit, uses the Local IP address to connect to the server.

Control Unit Management Vs. Data Unit Management

All management and monitoring can take place on the control node. From the control node, you can check runtime statistics, resource usage, or other monitoring information of all nodes. You can also issue a command to all nodes in the cluster, and replicate the console messages from data nodes to the control node.

You can monitor data nodes directly if desired. Although also available from the control node, you can perform file management on data nodes (including backing up the configuration and updating images). The following functions are not available from the control node:

- Monitoring per-node cluster-specific statistics.
- Syslog monitoring per node (except for syslogs sent to the console when console replication is enabled).
- SNMP
- NetFlow

Crypto Key Replication

When you create a crypto key on the control node, the key is replicated to all data nodes. If you have an SSH session to the Main cluster IP address, you will be disconnected if the control node fails. The new control node uses the same key for SSH connections, so that you do not need to update the cached SSH host key when you reconnect to the new control node.

ASDM Connection Certificate IP Address Mismatch

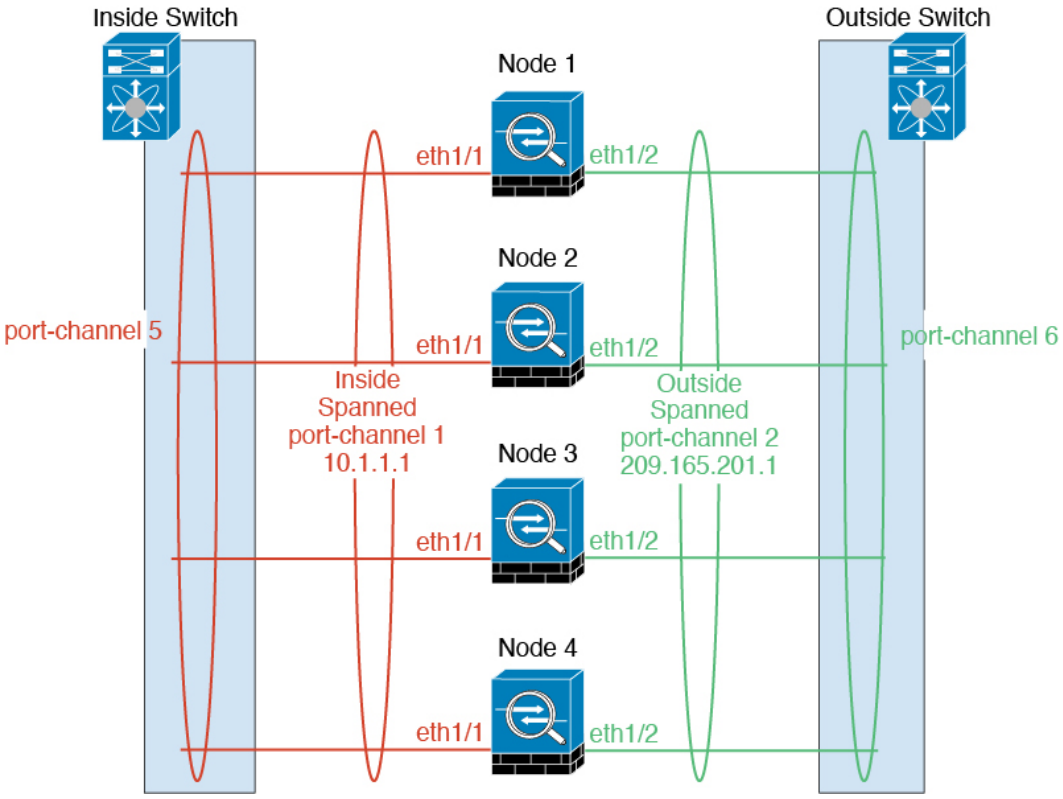
By default, a self-signed certificate is used for the ASDM connection based on the Local IP address. If you connect to the Main cluster IP address using ASDM, then a warning message about a mismatched IP address might appear because the certificate uses the Local IP address, and not the Main cluster IP address. You can ignore the message and establish the ASDM connection. However, to avoid this type of warning, you can enroll a certificate that contains the Main cluster IP address and all the Local IP addresses from the IP address pool. You can then use this certificate for each cluster member. See https://www.cisco.com/c/en/us/td/docs/security/asdm/identity-cert/cert-install.html for more information.

Spanned EtherChannels (Recommended)

You can group one or more interfaces per chassis into an EtherChannel that spans all chassis in the cluster. The EtherChannel aggregates the traffic across all the available active interfaces in the channel.

A Spanned EtherChannel can be configured in both routed and transparent firewall modes. In routed mode, the EtherChannel is configured as a routed interface with a single IP address. In transparent mode, the IP address is assigned to the BVI, not to the bridge group member interface.

The EtherChannel inherently provides load balancing as part of basic operation.



Inter-Site Clustering

For inter-site installations, you can take advantage of ASA clustering as long as you follow the recommended guidelines.

You can configure each cluster chassis to belong to a separate site ID.

Site IDs work with site-specific MAC addresses and IP addresses. Packets egressing the cluster use a site-specific MAC address and IP address, while packets received by the cluster use a global MAC address and IP address. This feature prevents the switches from learning the same global MAC address from both sites on two different ports, which causes MAC flapping; instead, they only learn the site MAC address. Site-specific MAC addresses and IP address are supported for routed mode using Spanned EtherChannels only.

Site IDs are also used to enable flow mobility using LISP inspection, director localization to improve performance and reduce round-trip time latency for inter-site clustering for data centers, and site redundancy for connections where a backup owner of a traffic flow is always at a different site from the owner.

See the following sections for more information about inter-site clustering:

- Sizing the Data Center Interconnect—Requirements and Prerequisites for Clustering on the Firepower 4100/9300 Chassis, on page 430
- Inter-Site Guidelines—Clustering Guidelines and Limitations, on page 433
- Configure Cluster Flow Mobility—Configure Cluster Flow Mobility, on page 460
- Enable Director Localization—Enable Director Localization, on page 458

- Enable Site Redundancy—Enable Director Localization, on page 458

Requirements and Prerequisites for Clustering on the Firepower 4100/9300 Chassis

Maximum Clustering Units Per Model

- Firepower 4100—16 chassis
- Firepower 9300—16 modules. For example, you can use 1 module in 16 chassis, or 2 modules in 8 chassis, or any combination that provides a maximum of 16 modules.

Hardware and Software Requirements for Inter-Chassis Clustering

All chassis in a cluster:

- For the Firepower 4100: All chassis must be the same model. For the Firepower 9300: All security modules must be the same type. For example, if you use clustering, all modules in the Firepower 9300 must be SM-40s. You can have different quantities of installed security modules in each chassis, although all modules present in the chassis must belong to the cluster including any empty slots.
- Must run the identical FXOS and application software except at the time of an image upgrade. Mismatched software versions can lead to poor performance, so be sure to upgrade all nodes in the same maintenance window.
- Must include the same interface configuration for interfaces you assign to the cluster, such as the same Management interface, EtherChannels, active interfaces, speed and duplex, and so on. You can use different network module types on the chassis as long as the capacity matches for the same interface IDs and interfaces can successfully bundle in the same spanned EtherChannel. Note that all data interfaces must be EtherChannels in clusters with multiple chassis. If you change the interfaces in FXOS after you enable clustering (by adding or removing interface modules, or configuring EtherChannels, for example), then perform the same changes on each chassis, starting with the data nodes, and ending with the control node. Note that if you remove an interface in FXOS, the ASA configuration retains the related commands so that you can make any necessary adjustments; removing an interface from the configuration can have wide effects. You can manually remove the old interface configuration.
- Must use the same NTP server. Do not set the time manually.
- ASA: Each FXOS chassis must be registered with the License Authority or satellite server. There is no extra cost for data nodes. For permanent license reservation, you must purchase separate licenses for each chassis. For Firepower Threat Defense, all licensing is handled by the FMC.

Switch Requirements

- Be sure to complete the switch configuration and successfully connect all the EtherChannels from the chassis to the switch(es) before you configure clustering on the Firepower 4100/9300 chassis.
- For supported switch characteristics, see Cisco FXOS Compatibility.

Sizing the Data Center Interconnect for Inter-Site Clustering

You should reserve bandwidth on the data center interconnect (DCI) for cluster control link traffic equivalent to the following calculation:

$$\frac{\text{\# of cluster members per site}}{2} \times \text{cluster control link size per member}$$

If the number of members differs at each site, use the larger number for your calculation. The minimum bandwidth for the DCI should not be less than the size of the cluster control link for one member.

For example:

- For 4 members at 2 sites:
  - 4 cluster members total
  - 2 members at each site
  - 5 Gbps cluster control link per member

  Reserved DCI bandwidth = 5 Gbps (2/2 x 5 Gbps).
- For 6 members at 3 sites, the size increases:
  - 6 cluster members total
  - 3 members at site 1, 2 members at site 2, and 1 member at site 3
  - 10 Gbps cluster control link per member

  Reserved DCI bandwidth = 15 Gbps (3/2 x 10 Gbps).
- For 2 members at 2 sites:
  - 2 cluster members total
  - 1 member at each site
  - 10 Gbps cluster control link per member

  Reserved DCI bandwidth = 10 Gbps (1/2 x 10 Gbps = 5 Gbps; but the minimum bandwidth should not be less than the size of the cluster control link (10 Gbps)).

Licenses for Clustering on the Firepower 4100/9300 Chassis

Smart Software Manager Regular and On-Prem

The clustering feature itself does not require any licenses. To use Strong Encryption and other optional licenses, each Firepower 4100/9300 chassis must be registered with the License Authority or Smart Software Manager Regular and On-Prem server. There is no extra cost for data units.

The Strong Encryption license is automatically enabled for qualified customers when you apply the registration token. When using the token, each chassis must have the same encryption license. For the optional Strong Encryption (3DES/AES) feature license enabled in the ASA configuration, see below.

In the ASA license configuration, you can only configure smart licensing on the control unit. The configuration is replicated to the data units, but for some licenses, they do not use the configuration; it remains in a cached state, and only the control unit requests the license. The licenses are aggregated into a single cluster license that is shared by the cluster units, and this aggregated license is also cached on the data units to be used if one of them becomes the control unit in the future. Each license type is managed as follows:

- Standard—Only the control unit requests the Standard license from the server, and both units can use it due to license aggregation.
- Context—Only the control unit requests the Context license from the server. The Standard license includes 10 contexts by default and is present on all cluster members. The value from each unit's Standard license plus the value of the Context license on the control unit are combined up to the platform limit in an aggregated cluster license. For example:
  - You have 6 Firepower 9300 modules in the cluster. The Standard license includes 10 contexts; for 6 units, these licenses add up to 60 contexts. You configure an additional 20-Context license on the control unit. Therefore, the aggregated cluster license includes 80 contexts. Because the platform limit for one module is 250, the combined license allows a maximum of 250 contexts; the 80 contexts are within the limit. Therefore, you can configure up to 80 contexts on the control unit; each data unit will also have 80 contexts through configuration replication.
  - You have 3 Firepower 4112 units in the cluster. The Standard license includes 10 contexts; for 3 units, these licenses add up to 30 contexts. You configure an additional 250-Context license on the control unit. Therefore, the aggregated cluster license includes 280 contexts. Because the platform limit for one unit is 250, the combined license allows a maximum of 250 contexts; the 280 contexts are over the limit. Therefore, you can only configure up to 250 contexts on the control unit; each data unit will also have 250 contexts through configuration replication. In this case, you should only configure the control unit Context license to be 220 contexts.
- Carrier—Required for Distributed S2S VPN. This license is a per-unit entitlement, and each unit requests its own license from the server.
- Strong Encryption (3DES)—For pre-2.3.0 Cisco Smart Software Manager On-Prem deployment; or if your Smart Account is not authorized for strong encryption, but Cisco has determined that you are allowed to use strong encryption, you can manually add a strong encryption license to your account. This license is a per-unit entitlement, and each unit requests its own license from the server.

If a new control unit is elected, the new control unit continues to use the aggregated license. It also uses the cached license configuration to re-request the control unit license. When the old control unit rejoins the cluster as a data unit, it releases the control unit license entitlement. Before the data unit releases the license, the control unit's license might be in a non-compliant state if there are no available licenses in the account. The retained license is valid for 30 days, but if it is still non-compliant after the grace period, you will not be able to make configuration changes to features requiring special licenses; operation is otherwise unaffected. The new active unit sends an entitlement authorization renewal request every 12 hours until the license is compliant. You should refrain from making configuration changes until the license requests are completely processed. If a unit leaves the cluster, the cached control configuration is removed, while the per-unit entitlements are retained. In particular, you would need to re-request the Context license on non-cluster units.

Permanent License Reservation

For permanent license reservation, you must purchase separate licenses for each chassis and enable the licenses *before* you configure clustering.

Clustering Guidelines and Limitations

Switches for Clustering

- Make sure connected switches match the MTU for both cluster data interfaces and the cluster control link interface. You should configure the cluster control link interface MTU to be at least 100 bytes higher than the data interface MTU, so make sure to configure the cluster control link connecting switch appropriately. Because the cluster control link traffic includes data packet forwarding, the cluster control link needs to accommodate the entire size of a data packet plus cluster traffic overhead. In addition, we do not recommend setting the cluster control link MTU between 2561 and 8362; due to block pool handling, this MTU size is not optimal for system operation.
- For Cisco IOS XR systems, if you want to set a non-default MTU, set the IOS XR interface MTU to be 14 bytes higher than the cluster device MTU. Otherwise, OSPF adjacency peering attempts may fail unless the **mtu-ignore** option is used. Note that the cluster device MTU should match the IOS XR *IPv4* MTU. This adjustment is not required for Cisco Catalyst and Cisco Nexus switches.
- On the switch(es) for the cluster control link interfaces, you can optionally enable Spanning Tree PortFast on the switch ports connected to the cluster unit to speed up the join process for new units.
- On the switch, we recommend that you use one of the following EtherChannel load-balancing algorithms: **source-dest-ip** or **src-dst-mixed-ip-port** (see the Cisco Nexus OS and Cisco IOS-XE **port-channel load-balance** command). Do not use a **vlan** keyword in the load-balance algorithm because it can cause unevenly distributed traffic to the devices in a cluster. *Do not* change the load-balancing algorithm from the default on the cluster device.
- If you change the load-balancing algorithm of the EtherChannel on the switch, the EtherChannel interface on the switch temporarily stops forwarding traffic, and the Spanning Tree Protocol restarts. There will be a delay before traffic starts flowing again.
- Switches on the cluster control link path should not verify the L4 checksum. Redirected traffic over the cluster control link does not have a correct L4 checksum. Switches that verify the L4 checksum could cause traffic to be dropped.
- Port-channel bundling downtime should not exceed the configured keepalive interval.
- On Supervisor 2T EtherChannels, the default hash distribution algorithm is adaptive. To avoid asymmetric traffic in a VSS design, change the hash algorithm on the port-channel connected to the cluster device to fixed:

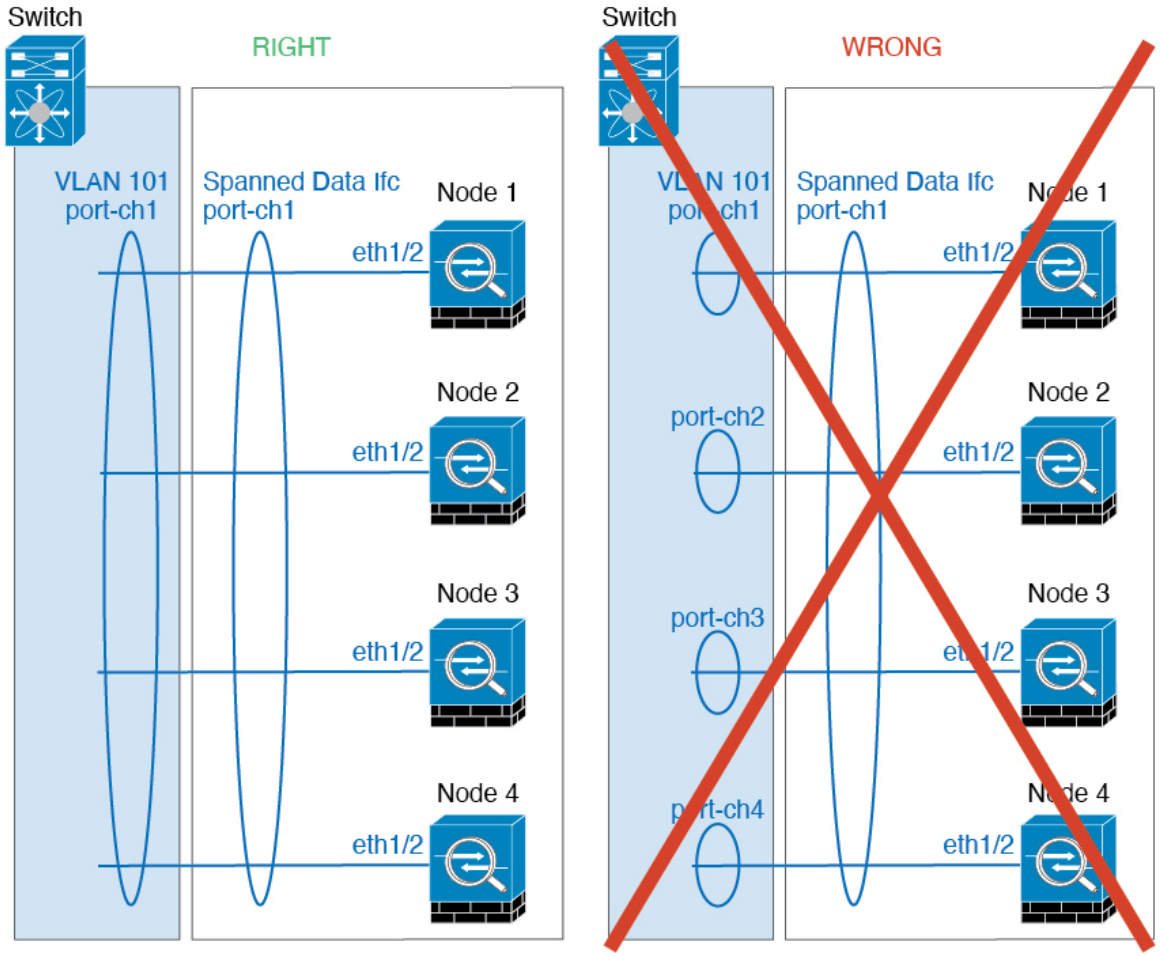
  router(config)# **port-channel** *id* **hash-distribution fixed**

  Do not change the algorithm globally; you may want to take advantage of the adaptive algorithm for the VSS peer link.

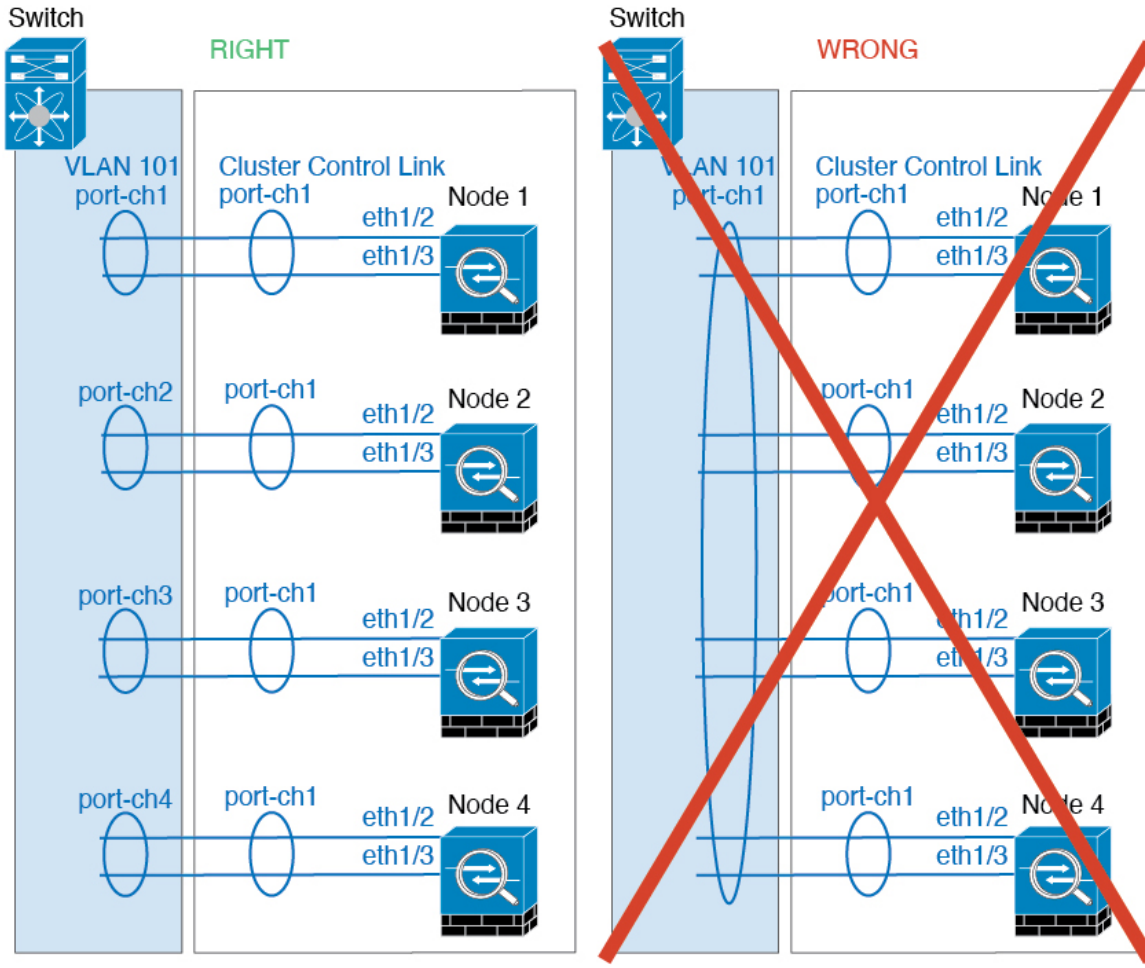
- Unlike ASA hardware clusters, Firepower 4100/9300 clusters support LACP graceful convergence. So for the platform, you can leave LACP graceful convergence enabled on connected Cisco Nexus switches.
- When you see slow bundling of a Spanned EtherChannel on the switch, you can enable LACP rate fast for an individual interface on the switch. FXOS EtherChannels have the LACP rate set to fast by default. Note that some switches, such as the Nexus series, do not support LACP rate fast when performing in-service software upgrades (ISSUs), so we do not recommend using ISSUs with clustering.

EtherChannels for Clustering

- In Catalyst 3750-X Cisco IOS software versions earlier than 15.1(1)S2, the cluster unit did not support connecting an EtherChannel to a switch stack. With default switch settings, if the cluster unit EtherChannel is connected cross stack, and if the control unit switch is powered down, then the EtherChannel connected to the remaining switch will not come up. To improve compatibility, set the **stack-mac persistent timer** command to a large enough value to account for reload time; for example, 8 minutes or 0 for indefinite. Or, you can upgrade to more a more stable switch software version, such as 15.1(1)S2.
- Spanned vs. Device-Local EtherChannel Configuration—Be sure to configure the switch appropriately for Spanned EtherChannels vs. Device-local EtherChannels.
  - Spanned EtherChannels—For cluster unit *Spanned* EtherChannels, which span across all members of the cluster, the interfaces are combined into a single EtherChannel on the switch. Make sure each interface is in the same channel group on the switch.



  - Device-local EtherChannels—For cluster unit *Device-local* EtherChannels including any EtherChannels configured for the cluster control link, be sure to configure discrete EtherChannels on the switch; do not combine multiple cluster unit EtherChannels into one EtherChannel on the switch.



Inter-Site Clustering

See the following guidelines for inter-site clustering:

- The cluster control link latency must be less than 20 ms round-trip time (RTT).
- The cluster control link must be reliable, with no out-of-order or dropped packets; for example, you should use a dedicated link.
- Do not configure connection rebalancing; you do not want connections rebalanced to cluster members at a different site.
- The ASA does not encrypt forwarded data traffic on the cluster control link because it is a dedicated link, even when used on a Data Center Interconnect (DCI). If you use Overlay Transport Virtualization (OTV), or are otherwise extending the cluster control link outside of the local administrative domain, you can configure encryption on your border routers such as 802.1AE MacSec over OTV.
- The cluster implementation does not differentiate between members at multiple sites for incoming connections; therefore, connection roles for a given connection may span across sites. This is expected behavior. However, if you enable director localization, the local director role is always chosen from the same site as the connection owner (according to site ID). Also, the local director chooses a new owner

at the same site if the original owner fails (Note: if the traffic is asymmetric across sites, and there is continuous traffic from the remote site after the original owner fails, then a node from the remote site might become the new owner if it receives a data packet within the re-hosting window.).

- For director localization, the following traffic types do not support localization: NAT or PAT traffic; SCTP-inspected traffic; Fragmentation owner query.
- For UDP long-lived flows in a North-South deployment, routing loops can occur if nodes at the original flow owner site fail and then come back up, after which the flow is directed back to the original site. If the new owner at the other site doesn't have a route to the destination, it will route the flow back to the internet, causing a loop. In this case, use the **clear conn** command on the new owner to force the flow to be reestablished.
- For transparent mode, if the cluster is placed between a pair of inside and outside routers (AKA North-South insertion), you must ensure that both inside routers share a MAC address, and also that both outside routers share a MAC address. When a cluster member at site 1 forwards a connection to a member at site 2, the destination MAC address is preserved. The packet will only reach the router at site 2 if the MAC address is the same as the router at site 1.
- For transparent mode, if the cluster is placed between data networks and the gateway router at each site for firewalling between internal networks (AKA East-West insertion), then each gateway router should use a First Hop Redundancy Protocol (FHRP) such as HSRP to provide identical virtual IP and MAC address destinations at each site. The data VLANs are extended across the sites using Overlay Transport Virtualization (OTV), or something similar. You need to create filters to prevent traffic that is destined to the local gateway router from being sent over the DCI to the other site. If the gateway router becomes unreachable at one site, you need to remove any filters so traffic can successfully reach the other site's gateway.
- For transparent mode, if the cluster is connected to an HSRP router, you must add the router HSRP MAC address as a static MAC address table entry on the ASA (see Add a Static MAC Address for Bridge Groups, on page 829). When adjacent routers use HSRP, traffic destined to the HSRP IP address will be sent to the HSRP MAC Address, but return traffic will be sourced from the MAC address of a particular router's interface in the HSRP pair. Therefore, the ASA MAC address table is typically only updated when the ASA ARP table entry for the HSRP IP address expires, and the ASA sends an ARP request and receives a reply. Because the ASA's ARP table entries expire after 14400 seconds by default, but the MAC address table entry expires after 300 seconds by default, a static MAC address entry is required to avoid MAC address table expiration traffic drops.
- For routed mode using Spanned EtherChannel, configure site-specific MAC addresses. Extend the data VLANs across the sites using OTV, or something similar. You need to create filters to prevent traffic that is destined to the global MAC address from being sent over the DCI to the other site. If the cluster becomes unreachable at one site, you need to remove any filters so traffic can successfully reach the other site's cluster nodes. Dynamic routing is not supported when an inter-site cluster acts as the first hop router for an extended segment.

Additional Guidelines

- When significant topology changes occur (such as adding or removing an EtherChannel interface, enabling or disabling an interface on the Firepower 4100/9300 chassis or the switch, adding an additional switch to form a VSS, vPC, StackWise, or StackWise Virtual) you should disable the health check feature, and also disable interface monitoring for the disabled interfaces . When the topology change is complete, and the configuration change is synced to all units, you can re-enable the health check feature.

- When adding a unit to an existing cluster, or when reloading a unit, there will be a temporary, limited packet/connection drop; this is expected behavior. In some cases, the dropped packets can hang connections; for example, dropping a FIN/ACK packet for an FTP connection will make the FTP client hang. In this case, you need to reestablish the FTP connection.
- If you use a Windows 2003 server connected to a Spanned EtherChannel interface, when the syslog server port is down, and the server does not throttle ICMP error messages, then large numbers of ICMP messages are sent back to the cluster. These messages can result in some units of the cluster experiencing high CPU, which can affect performance. We recommend that you throttle ICMP error messages.
- We recommend connecting EtherChannels to a VSS, vPC, StackWise, or StackWise Virtual for redundancy.
- Within a chassis, you cannot cluster some security modules and run other security modules in standalone mode; you must include all security modules in the cluster.

Defaults

- The cluster health check feature is enabled by default with the holdtime of 3 seconds. Interface health monitoring is enabled on all interfaces by default.
- Connection rebalancing is disabled by default. If you enable connection rebalancing, the default time between load information exchanges is 5 seconds.
- The cluster auto-rejoin feature for a failed cluster control link is set to unlimited attempts every 5 minutes.
- The cluster auto-rejoin feature for a failed data interface is set to 3 attempts every 5 minutes, with the increasing interval set to 2.
- Connection replication delay of 5 seconds is enabled by default for HTTP traffic.

Configure Clustering on the Firepower 4100/9300 Chassis

You can easily deploy the cluster from the Firepower 4100/9300 chassis supervisor. All initial configuration is automatically generated for each unit. This section describes the default bootstrap configuration and optional customization you can perform on the ASA. This section also describes how to manage cluster members from within the ASA. You can also manage cluster membership from the Firepower 4100/9300 chassis. See the Firepower 4100/9300 chassis documentation for more information.

Procedure

Step 1 FXOS: Add an ASA Cluster, on page 438
Step 2 ASA: Change the Firewall Mode and Context Mode, on page 448
Step 3 ASA: Configure Data Interfaces, on page 448
Step 4 ASA: Customize the Cluster Configuration, on page 451
Step 5 ASA: Manage Cluster Members, on page 472

FXOS: Add an ASA Cluster

You can add a single Firepower 9300 chassis as an intra-chassis cluster, or add multiple chassis for inter-chassis clustering. For inter-chassis clustering, you must configure each chassis separately. Add the cluster on one chassis; you can then enter most of the same settings on the next chassis.

Create an ASA Cluster

Set the scope to the image version.

You can easily deploy the cluster from the Firepower 4100/9300 chassis supervisor. All initial configuration is automatically generated for each unit.

For clustering on multiple chassis, you must configure each chassis separately. Deploy the cluster on one chassis; you can then copy the bootstrap configuration from the first chassis to the next chassis for ease of deployment.

In a Firepower 9300 chassis, you must enable clustering for all 3 module slots, even if you do not have a module installed. If you do not configure all 3 modules, the cluster will not come up.

For multiple context mode, you must first deploy the logical device, and then enable multiple context mode in the ASA application.

When you deploy a cluster, the Firepower 4100/9300 chassis supervisor configures each ASA application with the following bootstrap configuration. You can later modify parts of the bootstrap configuration from the ASA, if desired (shown in **Bold** text).

```
interface Port-channel48
   description Clustering Interface
cluster group <service_type_name>
   key <secret>
   local-unit unit-<chassis#-module#>
   site-id <number>
   cluster-interface port-channel48 ip 127.2.<chassis#>.<module#> 255.255.255.0
   priority <auto>
   health-check holdtime 3
   health-check data-interface auto-rejoin 3 5 2
   health-check cluster-interface auto-rejoin unlimited 5 1
   enable

ip local pool cluster_ipv4_pool <ip_address>-<ip_address> mask <mask>

interface <management_ifc>
   management-only individual
   nameif management
   security-level 0
   ip address <ip_address> <mask> cluster-pool cluster_ipv4_pool
   no shutdown

http server enable
http 0.0.0.0 0.0.0.0 management
route management <management_host_ip> <mask> <gateway_ip> 1
```



Note The **local-unit** name can only be changed if you disable clustering.

Before you begin

- Download the application image you want to use for the logical device from Cisco.com, and then upload that image to the Firepower 4100/9300 chassis.
- Gather the following information:
  - Management interface ID, IP address, and network mask
  - Gateway IP address

Procedure

Step 1 Configure interfaces.

a) Add at least one Data type interface or EtherChannel (also known as a port-channel) before you deploy the cluster. See Add an EtherChannel (Port Channel), on page 171 or Configure a Physical Interface, on page 169.

For clustering on multiple chassis, all data interfaces must be Spanned EtherChannels with at least one member interface. Add the same EtherChannels on each chassis. Combine the member interfaces from all cluster units into a single EtherChannel on the switch. See Clustering Guidelines and Limitations, on page 433 for more information about EtherChannels.

b) Add a Management type interface or EtherChannel. See Add an EtherChannel (Port Channel), on page 171 or Configure a Physical Interface, on page 169.

The management interface is required. Note that this management interface is not the same as the chassis management interface that is used only for chassis management (in FXOS, you might see the chassis management interface displayed as MGMT, management0, or other similar names).

For clustering on multiple chassis, add the same Management interface on each chassis.

c) For clustering on multiple chassis, add a member interface to the cluster control link EtherChannel (by default, port-channel 48). See Add an EtherChannel (Port Channel), on page 171.

Do not add a member interface for a cluster isolated to security modules within one Firepower 9300 chassis. If you add a member, the chassis assumes this cluster will be using multiple chassis, and will only allow you to use Spanned EtherChannels, for example.

Add the same member interfaces on each chassis. The cluster control link is a device-local EtherChannel on each chassis. Use separate EtherChannels on the switch per device. See Clustering Guidelines and Limitations, on page 433 for more information about EtherChannels.

Step 2 Enter security services mode.

scope ssa

Example:

```
Firepower# scope ssa
Firepower /ssa #
```

Step 3 Set the application instance parameters, including the image version.

a) View available images. Note the Version number that you want to use.

show app

Example:

```
Firepower /ssa # show app
    Name       Version         Author     Supported Deploy Types CSP Type    Is Default
 App
    ---------- --------------- ---------- ---------------------- -----------
-------------
    asa        9.9.1           cisco      Native                 Application No
    asa        9.10.1          cisco      Native                 Application Yes
    ftd        6.2.3           cisco      Native                 Application Yes
    ftd        6.3.0           cisco      Native,Container       Application Yes
```

b) Set the scope to the image version.

scope app asa *application_version*

Example:

```
Firepower /ssa # scope app asa 9.10.1
Firepower /ssa/app #
```

c) Set this version as the default.

set-default

Example:

```
Firepower /ssa/app # set-default
Firepower /ssa/app* #
```

d) Exit to ssa mode.

exit

Example:

```
Firepower /ssa/app* # exit
Firepower /ssa* #
```

Example:

```
Firepower /ssa # scope app asa 9.12.1
Firepower /ssa/app # set-default
Firepower /ssa/app* # exit
Firepower /ssa* #
```

Step 4 Create the cluster.

enter logical-device *device_name* **asa** *slots* **clustered**

- *device_name*—Used by the Firepower 4100/9300 chassis supervisor to configure clustering settings and assign interfaces; it is not the cluster name used in the security module configuration. You must specify all three security modules, even if you have not yet installed the hardware.

- *slots*—Assigns the chassis modules to the cluster. For the Firepower 4100, specify **1**. For the Firepower 9300, specify **1,2,3**. You must enable clustering for all 3 module slots in a Firepower 9300 chassis, even if you do not have a module installed. If you do not configure all 3 modules, the cluster will not come up.

Example:

```
Firepower /ssa # enter logical-device ASA1 asa 1,2,3 clustered
Firepower /ssa/logical-device* #
```

Step 5 Configure the cluster bootstrap parameters.

These settings are meant for initial deployment only, or for disaster recovery. For normal operation, you can later change most values in the application CLI configuration.

a) Create the cluster bootstrap object.

enter cluster-bootstrap

Example:

```
Firepower /ssa/logical-device* # enter cluster-bootstrap
Firepower /ssa/logical-device/cluster-bootstrap* #
```

b) Set the chassis ID.

set chassis-id *id*

Each chassis in the cluster needs a unique ID.

c) For inter-site clustering, set the site ID between 1 and 8.

set site-id *number*.

To remove the site ID, set the value to **0**.

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set site-id 1
Firepower /ssa/logical-device/cluster-bootstrap* #
```

d) Configure an authentication key for control traffic on the cluster control link.

set key

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set key
Key: diamonddogs
```

You are prompted to enter the shared secret.

The shared secret is an ASCII string from 1 to 63 characters. The shared secret is used to generate the key. This option does not affect datapath traffic, including connection state update and forwarded packets, which are always sent in the clear.

e) Set the cluster interface mode.

set mode spanned-etherchannel

Spanned EtherChannel mode is the only supported mode.

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set mode spanned-etherchannel
Firepower /ssa/logical-device/cluster-bootstrap* #
```

f) Set the cluster group name in the security module configuration.

set service-type *cluster_name*

The name must be an ASCII string from 1 to 38 characters.

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set service-type cluster1
Firepower /ssa/logical-device/cluster-bootstrap* #
```

g) (Optional) Set the cluster control link IP network.

set cluster-control-link network *a*.*b*.**0.0**

By default, the cluster control link uses the 127.2.0.0/16 network. However, some networking deployments do not allow 127.2.0.0/16 traffic to pass. In this case, you can specify a /16 address on a unique network for the cluster.

- *a*.*b*.**0.0**—Specify any /16 network address, except for loopback (127.0.0.0/8) and multicast (224.0.0.0/4) addresses. If you set the value to 0.0.0.0, then the default network is used: 127.2.0.0.

The chassis auto-generates the cluster control link interface IP address for each unit based on the chassis ID and slot ID: *a*.*b*.*chassis_id*.*slot_id*.

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set cluster-control-link network
10.10.0.0
```

h) Configure the management IP address information.

This information is used to configure a management interface in the security module configuration.

1. Configure a pool of Local IP addresses, one of which will be assigned to each cluster unit for the interface.

   set ipv4 pool *start_ip end_ip*

   set ipv6 pool *start_ip end_ip*

   Include at least as many addresses as there are units in the cluster. Note that for the Firepower 9300, you must include 3 addresses per chassis, even if you do not have all module slots filled. If you plan to expand the cluster, include additional addresses. The Virtual IP address (known as the Main cluster IP address) that belongs to the current control unit is *not* a part of this pool; be sure to reserve an IP address on the same network for the Main cluster IP address. You can use IPv4 and/or IPv6 addresses.

2. Configure the Main cluster IP address for the management interface.

   set virtual ipv4 *ip_address* **mask** *mask*

set virtual ipv6 *ip_address* **prefix-length** *prefix*

This IP address must be on the same network as the cluster pool addresses, but not be part of the pool.

3. Enter the network gateway address.

set ipv4 gateway *ip_address*

set ipv6 gateway *ip_address*

Example:

```
Firepower /ssa/logical-device/cluster-bootstrap* # set ipv4 gateway 10.1.1.254
Firepower /ssa/logical-device/cluster-bootstrap* # set ipv4 pool 10.1.1.11 10.1.1.27
Firepower /ssa/logical-device/cluster-bootstrap* # set ipv6 gateway 2001:DB8::AA
Firepower /ssa/logical-device/cluster-bootstrap* # set ipv6 pool 2001:DB8::11 2001:DB8::27
Firepower /ssa/logical-device/cluster-bootstrap* # set virtual ipv4 10.1.1.1 mask
255.255.255.0
Firepower /ssa/logical-device/cluster-bootstrap* # set virtual ipv6 2001:DB8::1
prefix-length 64
```

i) Exit the cluster bootstrap mode.

exit

Example:

```
Firepower /ssa/logical-device* # enter cluster-bootstrap
Firepower /ssa/logical-device/cluster-bootstrap* # set chassis-id 1
Firepower /ssa/logical-device/cluster-bootstrap* # set key
  Key: f@arscape
Firepower /ssa/logical-device/cluster-bootstrap* # set mode spanned-etherchannel
Firepower /ssa/logical-device/cluster-bootstrap* # set service-type cluster1
Firepower /ssa/logical-device/cluster-bootstrap* # exit
Firepower /ssa/logical-device/* #
```

Step 6 Configure the management bootstrap parameters.

These settings are meant for initial deployment only, or for disaster recovery. For normal operation, you can later change most values in the application CLI configuration.

a) Create the management bootstrap object.

enter mgmt-bootstrap asa

Example:

```
Firepower /ssa/logical-device* # enter mgmt-bootstrap asa
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

b) Specify the admin and enable password.

create bootstrap-key-secret PASSWORD

set value

Enter a value: *password*

Confirm the value: *password*

exit

Example:

The pre-configured ASA admin user and enable password is useful for password recovery; if you have FXOS access, you can reset the admin user password if you forget it.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key-secret PASSWORD
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # set value
Enter a value: floppylampshade
Confirm the value: floppylampshade
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key-secret* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

c) Specify the firewall mode, routed or transparent.

create bootstrap-key FIREWALL_MODE

set value {**routed** | **transparent**}

exit

In routed mode, the device is considered to be a router hop in the network. Each interface that you want to route between is on a different subnet. A transparent firewall, on the other hand, is a Layer 2 firewall that acts like a "bump in the wire," or a "stealth firewall," and is not seen as a router hop to connected devices.

The firewall mode is only set at initial deployment. If you re-apply the bootstrap settings, this setting is not used.

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # create bootstrap-key FIREWALL_MODE
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # set value routed
Firepower /ssa/logical-device/mgmt-bootstrap/bootstrap-key* # exit
Firepower /ssa/logical-device/mgmt-bootstrap* #
```

d) Exit the management bootstrap mode.

exit

Example:

```
Firepower /ssa/logical-device/mgmt-bootstrap* # exit
Firepower /ssa/logical-device* #
```

Step 7 Save the configuration.

commit-buffer

The chassis deploys the logical device by downloading the specified software version and pushing the bootstrap configuration and management interface settings to the application instance. Check the status of the deployment using the **show app-instance** command. The application instance is running and ready to use when the **Admin State** is **Enabled** and the **Oper State** is **Online**.

Example:

```
Firepower /ssa/logical-device* # commit-buffer
Firepower /ssa/logical-device # exit
Firepower /ssa # show app-instance
App Name   Identifier Slot ID    Admin State Oper State      Running Version Startup Version
 Deploy Type Profile Name Cluster State   Cluster Role
---------- ---------- ---------- ----------- --------------- --------------- ---------------
 ----------- ------------ --------------- ------------
ftd        cluster1   1          Enabled     Online          7.3.0.49        7.3.0.49
    Native                   In Cluster      Data Node
ftd        cluster1   2          Enabled     Online          7.3.0.49        7.3.0.49
    Native                   In Cluster      Control Node
ftd        cluster1   3          Disabled    Not Available                   7.3.0.49
    Native                   Not Applicable  None
```

Step 8 To add another chassis to the cluster, repeat this procedure except you must configure a unique **chassis-id** and the correct **site-id**; otherwise, use the same configuration for both chassis.

Make sure the interface configuration is the same on the new chassis. You can export and import FXOS chassis configuration to make this process easier.

Step 9 Connect to the control unit ASA to customize your clustering configuration.

Example

For chassis 1:

```
scope eth-uplink
  scope fabric a
    enter port-channel 1
      set port-type data
      enable
      enter member-port Ethernet1/1
        exit
      enter member-port Ethernet1/2
        exit
      exit
    enter port-channel 2
      set port-type data
      enable
      enter member-port Ethernet1/3
        exit
      enter member-port Ethernet1/4
        exit
      exit
    enter port-channel 3
      set port-type data
      enable
      enter member-port Ethernet1/5
        exit
      enter member-port Ethernet1/6
        exit
      exit
    enter port-channel 4
      set port-type mgmt
      enable
      enter member-port Ethernet2/1
        exit
      enter member-port Ethernet2/2
```

```
        exit
      exit
    enter port-channel 48
      set port-type cluster
      enable
      enter member-port Ethernet2/3
        exit
      exit
    exit
  exit
commit-buffer

scope ssa
  enter logical-device ASA1 asa "1,2,3" clustered
    enter cluster-bootstrap
      set chassis-id 1
      set ipv4 gateway 10.1.1.254
      set ipv4 pool 10.1.1.11 10.1.1.27
      set ipv6 gateway 2001:DB8::AA
      set ipv6 pool 2001:DB8::11 2001:DB8::27
      set key
      Key: f@arscape
      set mode spanned-etherchannel
      set service-type cluster1
      set virtual ipv4 10.1.1.1 mask 255.255.255.0
      set virtual ipv6 2001:DB8::1 prefix-length 64
      exit
    exit
  scope app asa 9.5.2.1
    set-default
    exit
  commit-buffer
```

For chassis 2:

```
scope eth-uplink
  scope fabric a
    create port-channel 1
      set port-type data
      enable
      create member-port Ethernet1/1
        exit
      create member-port Ethernet1/2
        exit
      exit
    create port-channel 2
      set port-type data
      enable
      create member-port Ethernet1/3
        exit
      create member-port Ethernet1/4
        exit
      exit
    create port-channel 3
      set port-type data
      enable
      create member-port Ethernet1/5
        exit
      create member-port Ethernet1/6
        exit
      exit
    create port-channel 4
      set port-type mgmt
```

```
      enable
      create member-port Ethernet2/1
        exit
      create member-port Ethernet2/2
        exit
      exit
    create port-channel 48
      set port-type cluster
      enable
      create member-port Ethernet2/3
        exit
      exit
    exit
  exit
commit-buffer

scope ssa
  enter logical-device ASA1 asa "1,2,3" clustered
    enter cluster-bootstrap
      set chassis-id 2
      set ipv4 gateway 10.1.1.254
      set ipv4 pool 10.1.1.11 10.1.1.15
      set ipv6 gateway 2001:DB8::AA
      set ipv6 pool 2001:DB8::11 2001:DB8::19
      set key
      Key: f@rscape
      set mode spanned-etherchannel
      set service-type cluster1
      set virtual ipv4 10.1.1.1 mask 255.255.255.0
      set virtual ipv6 2001:DB8::1 prefix-length 64
      exit
    exit
  scope app asa 9.5.2.1
    set-default
    exit
  commit-buffer
```

Add More Cluster Members

Add or replace the ASA cluster member.



Note This procedure only applies to adding or replacing a *chassis*; if you are adding or replacing a module to a Firepower 9300 where clustering is already enabled, the module will be added automatically.

Before you begin

- Make sure your existing cluster has enough IP addresses in the management IP address pool for this new member. If not, you need to edit the existing cluster bootstrap configuration on each chassis before you add this new member. This change causes a restart of the logical device.
- The interface configuration must be the same on the new chassis. You can export and import FXOS chassis configuration to make this process easier.
- For multiple context mode, enable multiple context mode in the ASA application on the first cluster member; additional cluster members will inherit the multiple context mode configuration automatically.

Procedure

Step 1 Click **OK**.

Step 2 To add another chassis to the cluster, repeat the procedure in Create an ASA Cluster, on page 438 except you must configure a unique **chassis-id** and the correct **site-id**; otherwise, use the same configuration for the new chassis.

ASA: Change the Firewall Mode and Context Mode

By default, the FXOS chassis deploys a cluster in routed firewall mode, and single context mode.

- Change the firewall mode— To change the mode after you depoy, change the mode on the control unit; the mode is automatically changed on all data units to match. See Set the Firewall Mode, on page 195. In multiple context mode, you set the firewall mode per context.
- Change to multiple context mode—To change to multiple context mode after you deploy, change the mode on the control unit; the mode is automatically changed on all data units to match. See Enable Multiple Context Mode, on page 224.

ASA: Configure Data Interfaces

This procedure configures basic parameters for each data interface that you assigned to the cluster when you deployed it in FXOS. For clustering on multiple chassis, data interfaces are always Spanned EtherChannel interfaces.



Note The management interface was pre-configured when you deployed the cluster. You can also change the management interface parameters in ASA, but this procedure focuses on data interfaces. The management interface is an individual interface, as opposed to a Spanned interface. See Management Interface, on page 427 for more information.

Before you begin

- For multiple context mode, start this procedure in the system execution space. If you are not already in the System configuration mode, enter the **changeto system** command.
- For transparent mode, configure the bridge group. See Configure the Bridge Virtual Interface (BVI), on page 677.
- When using Spanned EtherChannels for a cluster with multiple chassis, the port-channel interface will not come up until clustering is fully enabled. This requirement prevents traffic from being forwarded to a node that is not an active node in the cluster.

Procedure

Step 1 Specify the interface ID.

interface *id*

Refer to the FXOS chassis for the interfaces assigned to this cluster. The interface ID can be:

- **port-channel** *integer*
- **ethernet** *slot/port*

Example:

```
ciscoasa(config)# interface port-channel 1
```

Step 2 Enable the interface:

no shutdown

Step 3 (Optional) If you are creating VLAN subinterfaces on this interface, do so now.

Example:

```
ciscoasa(config)# interface port-channel 1.10
ciscoasa(config-if)# vlan 10
```

The rest of this procedure applies to the subinterfaces.

Step 4 (Multiple Context Mode) Allocate the interface to a context, then changeto the context and enter interface mode.

Example:

```
ciscoasa(config)# context admin
ciscoasa(config)# allocate-interface port-channel1
ciscoasa(config)# changeto context admin
ciscoasa(config-if)# interface port-channel 1
```

For multiple context mode, the rest of the interface configuration occurs within each context.

Step 5 Name the interface:

nameif *name*

Example:

```
ciscoasa(config-if)# nameif inside
```

The *name* is a text string up to 48 characters, and is not case-sensitive. You can change the name by reentering this command with a new value.

Step 6 Perform one of the following, depending on the firewall mode.

- Routed Mode—Set the IPv4 and/or IPv6 address:

  (IPv4)

  ip address *ip_address* [*mask*]

  (IPv6)

  ipv6 address *ipv6-prefix*/*prefix-length*

  Example:

  ```
  ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0
  ciscoasa(config-if)# ipv6 address 2001:DB8::1001/32
  ```

  DHCP, PPPoE, and IPv6 autoconfig are not supported. For point-to-point connections, you can specify a 31-bit subnet mask (255.255.255.254). In this case, no IP addresses are reserved for the network or broadcast addresses. Manually configuring the link-local address is also not supported.

- Transparent Mode—Assign the interface to a bridge group:

  bridge-group *number*

  Example:

  ```
  ciscoasa(config-if)# bridge-group 1
  ```

  Where *number* is an integer between 1 and 100. You can assign up to 64 interfaces to a bridge group. You cannot assign the same interface to more than one bridge group. Note that the BVI configuration includes the IP address.

Step 7 Set the security level:

security-level *number*

Example:

```
ciscoasa(config-if)# security-level 50
```

Where *number* is an integer between 0 (lowest) and 100 (highest).

Step 8 (Clustering on multiple chassis) Configure a unique, manual global MAC address for a Spanned EtherChannel to avoid potential network connectivity problems.

mac-address *mac_address*

- *mac_address*—The MAC address is in H.H.H format, where H is a 16-bit hexadecimal digit. For example, the MAC address 00-0C-F1-42-4C-DE is entered as 000C.F142.4CDE. The first two bytes of a manual MAC address cannot be A2 if you also want to use auto-generated MAC addresses.

You must configure a unique MAC address not currently in use on your network. With a manually-configured MAC address, the MAC address stays with the current control unit. If you do not configure a MAC address, then if the control unit changes, the new control unit uses a new MAC address for the interface, which can cause a temporary network outage.

In multiple context mode, if you share an interface between contexts, you should instead enable auto-generation of MAC addresses so you do not need to set the MAC address manually. Note that you must manually configure the MAC address using this command for *non-shared* interfaces.

Example:

```
ciscoasa(config-if)# mac-address 000C.F142.4CDE
```

Step 9 (Inter-site clustering) Configure a site-specific MAC address and, for routed mode, an IP address for each site:

mac-address *mac_address* **site-id** *number* **site-ip** *ip_address*

Example:

```
ciscoasa(config-if)# mac-address aaaa.1111.1234
ciscoasa(config-if)# mac-address aaaa.1111.aaaa site-id 1 site-ip 10.9.9.1
ciscoasa(config-if)# mac-address aaaa.1111.bbbb site-id 2 site-ip 10.9.9.2
ciscoasa(config-if)# mac-address aaaa.1111.cccc site-id 3 site-ip 10.9.9.3
ciscoasa(config-if)# mac-address aaaa.1111.dddd site-id 4 site-ip 10.9.9.4
```

The site-specific IP addresses must be on the same subnet as the global IP address. The site-specific MAC address and IP address used by a unit depends on the site ID you specify in each unit's bootstrap configuration.

ASA: Customize the Cluster Configuration

If you want to change bootstrap settings after you deploy the cluster or configure additional options, such as clustering health monitoring, TCP connection replication delay, flow mobility, and other optimizations, you can do so on the control unit.

Configure Basic ASA Cluster Parameters

You can customize cluster settings on the control node.

Before you begin

- For multiple context mode, complete this procedure in the system execution space on the control unit. To change from the context to the system execution space, enter the **changeto system** command.
- The local-unit name and several other options can only be set on the FXOS chassis, or they can only be changed on the ASA if you disable clustering, so they are not included in the following procedure.

Procedure

Step 1 Confirm that this unit is the control unit:

show cluster info

Example:

```
asa(config)# show cluster info
Cluster cluster1: On
    Interface mode: spanned
    This is "unit-1-2" in state MASTER
        ID        : 2
        Version   : 9.5(2)
        Serial No.: FCH183770GD
        CCL IP    : 127.2.1.2
        CCL MAC   : 0015.c500.019f
        Last join : 01:18:34 UTC Nov 4 2015
        Last leave: N/A
Other members in the cluster:
    Unit "unit-1-3" in state SLAVE
        ID        : 4
        Version   : 9.5(2)
        Serial No.: FCH19057ML0
        CCL IP    : 127.2.1.3
        CCL MAC   : 0015.c500.018f
        Last join : 20:29:57 UTC Nov 4 2015
        Last leave: 20:24:55 UTC Nov 4 2015
    Unit "unit-1-1" in state SLAVE
        ID        : 1
        Version   : 9.5(2)
        Serial No.: FCH19057ML0
        CCL IP    : 127.2.1.1
        CCL MAC   : 0015.c500.017f
        Last join : 20:20:53 UTC Nov 4 2015
        Last leave: 20:18:15 UTC Nov 4 2015
    Unit "unit-2-1" in state SLAVE
        ID        : 3
        Version   : 9.5(2)
        Serial No.: FCH19057ML0
        CCL IP    : 127.2.2.1
        CCL MAC   : 0015.c500.020f
        Last join : 20:19:57 UTC Nov 4 2015
        Last leave: 20:24:55 UTC Nov 4 2015
```

If a different unit is the control unit, exit the connection and connect to the correct unit.

Step 2 Specify the maximum transmission unit for the cluster control link interface to be at least 100 bytes higher than the highest MTU of the data interfaces.

mtu cluster *bytes*

Example:

```
ciscoasa(config)# mtu cluster 9184
```

We suggest setting the cluster control link MTU to the maximum; the minimum value is 1400 bytes. In addition, we do not recommend setting the cluster control link MTU between 2561 and 8362; due to block pool handling, this MTU size is not optimal for system operation. Because the cluster control link traffic includes data packet forwarding, the cluster control link needs to accommodate the entire size of a data packet plus cluster traffic overhead. For example, because the maximum MTU is 9184, then the highest data interface MTU can be 9084, while the cluster control link can be set to 9184.

Step 3 Enter cluster configuration mode:

cluster group *name*

Step 4 (Optional) Enable console replication from data units to the control unit:

console-replicate

This feature is disabled by default. The ASA prints out some messages directly to the console for certain critical events. If you enable console replication, data units send the console messages to the control unit so that you only need to monitor one console port for the cluster.

Step 5 Set the minimum trace level for clustering events:

trace-level *level*

Set the minimum level as desired:

- **critical**—Critical events (severity=1)
- **warning**—Warnings (severity=2)
- **informational**—Informational events (severity=3)
- **debug**—Debugging events (severity=4)

Step 6 (Optional) (Firepower 9300 only) Ensure that the security modules in a chassis join the cluster simultaneously, so that traffic is evenly distributed between the modules. If a module joins very much in advance of other modules, it can receive more traffic than desired, because the other modules cannot yet share the load.

unit parallel-join *num_of_units* **max-bundle-delay** *max_delay_time*

- *num_of_units*—Specifies the minimum number of modules in the same chassis required to be ready before a module can join the cluster, between 1 and 3. The default is 1, meaning that a module will not wait for other modules to be ready before it joins the cluster. If you set the value to 3, for example, then each module will wait the *max_delay_time* or until all 3 modules are ready before joining the cluster. All 3 modules will request to join the cluster roughly simultaneously, and will all start receiving traffic around the same time.
- *max_delay_time*—Specifies the maximum delay time in minutes before a module stops waiting for other modules to be ready before it joins the cluster, between 0 and 30 minutes. The default is 0, meaning the module will not wait for other modules to be ready before it joins the cluster. If you set the *num_of_units* to 1, then this value must be 0. If you set the *num_of_units* to 2 or 3, then this value must be 1 or more. This timer is per module, but when the first module joins the cluster, then all other module timers end, and the remaining modules join the cluster.

For example, you set the *num_of_units* to 3, and the *max_delay_time* to 5 minutes. When module 1 comes up, it starts its 5 minute timer. Module 2 comes up 2 minutes later and starts its 5 minute timer. Module 3 comes up 1 minute later, therefore all modules will now join the cluster at the 4 minute mark; they will not wait for the timers to complete. If module 3 never comes up, then Module 1 will join the cluster at the end of its 5 minute timer, and Module 2 will also join, even though its timer still has 2 minutes remaining; it will not wait for its timer to complete.

Step 7 Configure the maximum number of cluster members.

cluster-member-limit *number*

- *number*—2 to 16. The default is 16.

If you know that your cluster will be fewer than the maximum of 16 units, then we recommend that you set the actual planned number of units. Setting the maximum units lets the cluster manage resources better. For

example, if you use port address translation (PAT), then the control unit can allocate port blocks to the planned number of members, and it will not have to reserve ports for extra units you don't plan to use.

Configure Health Monitoring and Auto-Rejoin Settings

This procedure configures unit and interface health monitoring.

You might want to disable health monitoring of non-essential interfaces, for example, the management interface. You can monitor any port-channel ID or single physical interface ID. Health monitoring is not performed on VLAN subinterfaces or virtual interfaces such as VNIs or BVIs. You cannot configure monitoring for the cluster control link; it is always monitored.

Procedure

Step 1 Enter cluster configuration mode:

cluster group *name*

Step 2 Customize the cluster unit health check feature:

health-check [**holdtime** *timeout*]

The **holdime** determines the amount of time between unit heartbeat status messages, between .3 and 45 seconds; The default is 3 seconds.

To determine unit health, the ASA cluster units send heartbeat messages on the cluster control link to other units. If a unit does not receive any heartbeat messages from a peer unit within the holdtime period, the peer unit is considered unresponsive or dead.

When any topology changes occur (such as adding or removing a data interface, enabling or disabling an interface on the ASA, Firepower 4100/9300 chassis, or the switch, or adding an additional switch to form a VSS, vPC, StackWise, or StackWise Virtual) you should disable the health check feature and also disable interface monitoring for the disabled interfaces (**no health-check monitor-interface**). When the topology change is complete, and the configuration change is synced to all units, you can re-enable the health check feature.

Example:

```
ciscoasa(cfg-cluster)# health-check holdtime 5
```

Step 3 Disable the interface health check on an interface:

no health-check monitor-interface [*interface_id* | **service-application**]

The interface health check monitors for link failures. If all physical ports for a given logical interface fail on a particular unit, but there are active ports under the same logical interface on other units, then the unit is removed from the cluster. The amount of time before the ASA removes a member from the cluster depends on the type of interface and whether the unit is an established member or is joining the cluster.

Health check is enabled by default for all interfaces. You can disable it per interface using the **no** form of this command. You might want to disable health monitoring of non-essential interfaces, for example, the management interface. Health monitoring is not performed on VLAN subinterfaces or virtual interfaces such

as VNIs or BVIs. You cannot configure monitoring for the cluster control link; it is always monitored. Specify the **service-application** to disable monitoring of a decorator application.

When any topology changes occur (such as adding or removing a data interface, enabling or disabling an interface on the ASA, Firepower 4100/9300 chassis, or the switch, or adding an additional switch to form a VSS, vPC, StackWise, or StackWise Virtual) you should disable the health check feature (**no health-check**) and also disable interface monitoring for the disabled interfaces. When the topology change is complete, and the configuration change is synced to all units, you can re-enable the health check feature.

Example:

```
ciscoasa(cfg-cluster)# no health-check monitor-interface port-channel1
```

Step 4 Customize the auto-rejoin cluster settings after a health check failure:

health-check {**data-interface** | **cluster-interface** | **system**} **auto-rejoin** [**unlimited** | *auto_rejoin_max*] *auto_rejoin_interval auto_rejoin_interval_variation*

- **system**—Specifies the auto-rejoin settings for internal errors. Internal failures include: application sync timeout; inconsistent application statuses; and so on.
- **unlimited**—(Default for the **cluster-interface**) Does not limit the number of rejoin attempts.
- *auto-rejoin-max*—Sets the number of rejoin attempts, between 0 and 65535. **0** disables auto-rejoining. The default for the **data-interface** and **system** is 3.
- *auto_rejoin_interval*—Defines the interval duration in minutes between rejoin attempts, between 2 and 60. The default value is 5 minutes. The maximum total time that the unit attempts to rejoin the cluster is limited to 14400 minutes (10 days) from the time of last failure.
- *auto_rejoin_interval_variation*—Defines if the interval duration increases. Set the value between 1 and 3: **1** (no change); **2** (2 x the previous duration), or **3** (3 x the previous duration). For example, if you set the interval duration to 5 minutes, and set the variation to 2, then the first attempt is after 5 minutes; the 2nd attempt is 10 minutes (2 x 5); the 3rd attempt 20 minutes (2 x 10), and so on. The default value is **1** for the cluster-interface and **2** for the data-interface and system.

Example:

```
ciscoasa(cfg-cluster)# health-check data-interface auto-rejoin 10 3 3
```

Step 5 Configure the debounce time before the ASA considers an interface to be failed and the unit is removed from the cluster.

health-check monitor-interface debounce-time *ms*

Set the debounce time between 300 and 9000 ms. The default is 500 ms. Lower values allow for faster detection of interface failures. Note that configuring a lower debounce time increases the chances of false-positives. When an interface status update occurs, the ASA waits the number of milliseconds specified before marking the interface as failed and the unit is removed from the cluster. In the case of an EtherChannel that transitions from a down state to an up state (for example, the switch reloaded, or the switch enabled an EtherChannel), a longer debounce time can prevent the interface from appearing to be failed on a cluster unit just because another cluster unit was faster at bundling the ports.

Example:

```
ciscoasa(cfg-cluster)# health-check monitor-interface debounce-time 300
```

Step 6 Configure the chassis health check interval:

app-agent heartbeat [**interval** *ms*] [**retry-count** *number*]

- **interval** *ms*—Set the amount of time between heartbeats, between 100 and 6000 ms, in multiples of 100. The default is 1000 ms.
- **retry-count** *number*—Set the number of retries, between 1 and 30. The default is 3 retries.

The ASA checks whether it can communicate over the backplane with the host chassis.

The minimum combined time (*interval* x *retry-count*) cannot be less than 600 ms. For example, if you set the interval to 100, and the retry count to 3, then the total combined time is 300 ms, which is not supported. For example, you can set the interval to 100, and the retry count to 6 to meet the minimum time (600 ms).

Example:

```
ciscoasa(cfg-cluster)# app-agent heartbeat interval 1000 retry-count 10
```

Step 7 (Optional) Configure traffic load monitoring.

load-monitor [**frequency** *seconds*] [**intervals** *intervals*]

- **frequency** *seconds*—Sets the time in seconds between monitoring messages, between 10 and 360 seconds. The default is 20 seconds.
- **intervals** *intervals*—Sets the number of intervals for which the ASA maintains data, between 1 and 60. The default is 30.

You can monitor the traffic load for cluster members, including total connection count, CPU and memory usage, and buffer drops. If the load is too high, you can choose to manually disable clustering on the unit if the remaining units can handle the load, or adjust the load balancing on the external switch. This feature is enabled by default. For example, for inter-chassis clustering on the Firepower 9300 with 3 security modules in each chassis, if 2 security modules in a chassis leave the cluster, then the same amount of traffic to the chassis will be sent to the remaining module and potentially overwhelm it. You can periodically monitor the traffic load. If the load is too high, you can choose to manually disable clustering on the unit.

Use the **show cluster info load-monitor** command to view the traffic load.

Example:

```
ciscoasa(cfg-cluster)# load-monitor frequency 50 intervals 25
ciscoasa(cfg-cluster)# show cluster info load-monitor
ID  Unit Name
0  B
1  A_1
Information from all units with 50 second interval:
Unit     Connections    Buffer Drops   Memory Used   CPU Used
Average from last 1 interval:
   0           0            0            14              25
   1           0            0            16              20
Average from last 25 interval:
   0           0            0            12              28
```

```
1           0           0           13              27
```

Configure Connection Rebalancing and the Cluster TCP Replication Delay

You can configure connection rebalancing. You can enable the cluster replication delay for TCP connections to help eliminate the "unnecessary work" related to short-lived flows by delaying the director/backup flow creation. Note that if a unit fails before the director/backup flow is created, then those flows cannot be recovered. Similarly, if traffic is rebalanced to a different unit before the flow is created, then the flow cannot be recovered. You should not enable the TCP replication delay for traffic on which you disable TCP randomization.

Procedure

Step 1 Enter cluster configuration mode:

cluster group *name*

Step 2 (Optional) Enable connection rebalancing for TCP traffic:

conn-rebalance [**frequency** *seconds*]

Example:

```
ciscoasa(cfg-cluster)# conn-rebalance frequency 60
```

This command is disabled by default. If enabled, ASAs exchange information about the connections per second periodically, and offload new connections from devices with more connections per second to less loaded devices. Existing connections are never moved. Moreover, because this command only rebalances based on connections per second, the total number of established connections on each node is not considered, and the total number of connections may not be equal. The frequency, between 1 and 360 seconds, specifies how often the load information is exchanged. The default is 5 seconds.

Once a connection is offloaded to a different node, it becomes an asymmetric connection.

Do not configure connection rebalancing for inter-site topologies; you do not want new connections rebalanced to cluster members at a different site.

Step 3 Enable the cluster replication delay for TCP connections:

cluster replication delay *seconds* {**http | match tcp {host** *ip_address* | *ip_address mask* | **any | any4 | any6} [{eq | lt | gt}** *port*] {**host** *ip_address* | *ip_address mask* | **any | any4 | any6} [{eq | lt | gt}** *port*]}

Example:

```
ciscoasa(config)# cluster replication delay 15 match tcp any any eq ftp
ciscoasa(config)# cluster replication delay 15 http
```

Set the *seconds* between 1 and 15. The **http** delay is enabled by default for 5 seconds.

Configure Inter-Site Features

For inter-site clustering, you can customize your configuration to enhance redundancy and stability.

Enable Director Localization

To improve performance and reduce round-trip time latency for inter-site clustering for data centers, you can enable director localization. New connections are typically load-balanced and owned by cluster members within a given site. However, the ASA assigns the director role to a member at *any* site. Director localization enables additional director roles: a local director at the same site as the owner, and a global director that can be at any site. Keeping the owner and director at the same site improves performance. Also, if the original owner fails, the local director chooses a new connection owner at the same site. The global director is used if a cluster member receives packets for a connection that is owned on a different site.

Before you begin

- Set the site ID for the chassis on Firepower 4100/9300 chassis supervisor.
- The following traffic types do not support localization: NAT or PAT traffic; SCTP-inspected traffic; Fragmentation owner query.

Procedure

Step 1 Enter cluster configuration mode:

cluster group *name*

Example:

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)#
```

Step 2 Enable director localization:

director-localization

Enable Site Redundancy

To protect flows from a site failure, you can enable site redundancy. If the connection backup owner is at the same site as the owner, then an additional backup owner will be chosen from another site to protect flows from a site failure.

Before you begin

- Set the site ID for the chassis on Firepower 4100/9300 chassis supervisor.

Procedure

Step 1 Enter cluster configuration mode:

cluster group *name*

Example:

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)#
```

Step 2 Enable site redundancy.

site-redundancy

Configure Per-Site Gratuitous ARP

The ASA generates gratuitous ARP (GARP) packets to keep the switching infrastructure up to date: the highest priority member at each site periodically generates GARP traffic for the global MAC/IP addresses.

When using per-site MAC and IP addresses, packets sourced from the cluster use a site-specific MAC address and IP address, while packets received by the cluster use a global MAC address and IP address. If traffic is not generated from the global MAC address periodically, you could experience a MAC address timeout on your switches for the global MAC address. After a timeout, traffic destined for the global MAC address will be flooded across the entire switching infrastructure, which can cause performance and security concerns.

GARP is enabled by default when you set the site ID for each unit and the site MAC address for each Spanned EtherChannel. You can customize the GARP interval, or you can disable GARP.

Before you begin

- Set the site ID for the cluster member in the bootstrap configuration.
- Set the per-site MAC address for the Spanned EtherChannel in the control unit configuration.

Procedure

Step 1 Enter cluster configuration mode.

cluster group *name*

Example:

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)#
```

Step 2 Customize the GARP interval.

site-periodic-garp interval *seconds*

- *seconds*—Sets the time in seconds between GARP generation, between 1 and 1000000 seconds. The default is 290 seconds.

To disable GARP, enter **no site-periodic-garp interval**.

Example:

```
ciscoasa(cfg-cluster)# site-periodic-garp interval 500
```

Configure Cluster Flow Mobility

You can inspect LISP traffic to enable flow mobility when a server moves between sites.

About LISP Inspection

You can inspect LISP traffic to enable flow mobility between sites.

About LISP

Data center virtual machine mobility such as VMware VMotion enables servers to migrate between data centers while maintaining connections to clients. To support such data center server mobility, routers need to be able to update the ingress route towards the server when it moves. Cisco Locator/ID Separation Protocol (LISP) architecture separates the device identity, or endpoint identifier (EID), from its location, or routing locator (RLOC), into two different numbering spaces, making server migration transparent to clients. For example, when a server moves to a new site and a client sends traffic to the server, the router redirects traffic to the new location.

LISP requires routers and servers in certain roles, such as the LISP egress tunnel router (ETR), ingress tunnel router (ITR), first hop routers, map resolver (MR), and map server (MS). When the first hop router for the server senses that the server is connected to a different router, it updates all of the other routers and databases so that the ITR connected to the client can intercept, encapsulate, and send traffic to the new server location.

ASA LISP Support

The ASA does not run LISP itself; it can, however, inspect LISP traffic for location changes and then use this information for seamless clustering operation. Without LISP integration, when a server moves to a new site, traffic comes to an ASA cluster member at the new site instead of to the original flow owner. The new ASA forwards traffic to the ASA at the old site, and then the old ASA has to send traffic back to the new site to reach the server. This traffic flow is sub-optimal and is known as “tromboning” or “hair-pinning.”

With LISP integration, the ASA cluster members can inspect LISP traffic passing between the first hop router and the ETR or ITR, and can then change the flow owner to be at the new site.

LISP Guidelines

- The ASA cluster members must reside between the first hop router and the ITR or ETR for the site. The ASA cluster itself cannot be the first hop router for an extended segment.
- Only fully-distributed flows are supported; centralized flows, semi-distributed flows, or flows belonging to individual nodes are not moved to new owners. Semi-distributed flows include applications, such as SIP, where all child flows are owned by the same ASA that owns the parent flow.
- The cluster only moves Layer 3 and 4 flow states; some application data might be lost.

- For short-lived flows or non-business-critical flows, moving the owner may not be worthwhile. You can control the types of traffic that are supported with this feature when you configure the inspection policy, and should limit flow mobility to essential traffic.

ASA LISP Implementation

This feature includes several inter-related configurations (all of which are described in this chapter):

1. (Optional) Limit inspected EIDs based on the host or server IP address—The first hop router might send EID-notify messages for hosts or networks the ASA cluster is not involved with, so you can limit the EIDs to only those servers or networks relevant to your cluster. For example, if the cluster is only involved with 2 sites, but LISP is running on 3 sites, you should only include EIDs for the 2 sites involved with the cluster.
2. LISP traffic inspection—The ASA inspects LISP traffic on UDP port 4342 for the EID-notify message sent between the first hop router and the ITR or ETR. The ASA maintains an EID table that correlates the EID and the site ID. For example, you should inspect LISP traffic with a source IP address of the first hop router and a destination address of the ITR or ETR. Note that LISP traffic is not assigned a director, and LISP traffic itself does not participate in cluster state sharing.
3. Service Policy to enable flow mobility on specified traffic—You should enable flow mobility on business-critical traffic. For example, you can limit flow mobility to only HTTPS traffic, and/or to traffic to specific servers.
4. Site IDs—The ASA uses the site ID for each cluster node to determine the new owner.
5. Cluster-level configuration to enable flow mobility—You must also enable flow mobility at the cluster level. This on/off toggle lets you easily enable or disable flow mobility for a particular class of traffic or applications.

Configure LISP Inspection

You can inspect LISP traffic to enable flow mobility when a server moves between sites.

Before you begin

- Set the site ID for the chassis on Firepower 4100/9300 chassis supervisor.
- LISP traffic is not included in the default-inspection-traffic class, so you must configure a separate class for LISP traffic as part of this procedure.

Procedure

Step 1 (Optional) Configure a LISP inspection map to limit inspected EIDs based on IP address, and to configure the LISP pre-shared key:

a) Create an extended ACL; only the destination IP address is matched to the EID embedded address:

access list *eid_acl_name* **extended permit ip** *source_address mask destination_address mask*

Both IPv4 and IPv6 ACLs are accepted. See the command reference for exact **access-list extended** syntax.

b) Create the LISP inspection map, and enter parameters mode:

policy-map type inspect lisp *inspect_map_name*

parameters

c) Define the allowed EIDs by identifying the ACL you created:

allowed-eid access-list *eid_acl_name*

The first hop router or ITR/ETR might send EID-notify messages for hosts or networks the ASA cluster is not involved with, so you can limit the EIDs to only those servers or networks relevant to your cluster. For example, if the cluster is only involved with 2 sites, but LISP is running on 3 sites, you should only include EIDs for the 2 sites involved with the cluster.

d) If necessary, enter the pre-shared key:

validate-key *key*

Example:

```
ciscoasa(config)# access-list TRACKED_EID_LISP extended permit ip any 10.10.10.0 255.255.255.0
ciscoasa(config)# policy-map type inspect lisp LISP_EID_INSPECT
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# allowed-eid access-list TRACKED_EID_LISP
ciscoasa(config-pmap-p)# validate-key MadMaxShinyandChrome
```

Step 2 Configure LISP inspection for UDP traffic between the first hop router and the ITR or ETR on port 4342:

a) Configure the extended ACL to identify LISP traffic:

access list *inspect_acl_name* **extended permit udp** *source_address mask destination_address mask* **eq 4342**

You *must* specify UDP port 4342. Both IPv4 and IPv6 ACLs are accepted. See the command reference for exact **access-list extended** syntax.

b) Create a class map for the ACL:

class-map *inspect_class_name*

match access-list *inspect_acl_name*

c) Specify the policy map, the class map, enable inspection using the optional LISP inspection map, and apply the service policy to an interface (if new):

policy-map *policy_map_name*

class *inspect_class_name*

inspect lisp [*inspect_map_name*]

service-policy *policy_map_name* {**global** | **interface** *ifc_name*}

If you have an existing service policy, specify the existing policy map name. By default, the ASA includes a global policy called **global_policy**, so for a global policy, specify that name. You can also create one service policy per interface if you do not want to apply the policy globally. LISP inspection is applied to traffic bidirectionally so you do not need to apply the service policy on both the source and destination interfaces; all traffic that enters or exits the interface to which you apply the policy map is affected if the traffic matches the class map for both directions.

Example:

```
ciscoasa(config)# access-list LISP_ACL extended permit udp host 192.168.50.89 host
```

```
192.168.10.8 eq 4342
ciscoasa(config)# class-map LISP_CLASS
ciscoasa(config-cmap)# match access-list LISP_ACL
ciscoasa(config-cmap)# policy-map INSIDE_POLICY
ciscoasa(config-pmap)# class LISP_CLASS
ciscoasa(config-pmap-c)# inspect lisp LISP_EID_INSPECT
ciscoasa(config)# service-policy INSIDE_POLICY interface inside
```

The ASA inspects LISP traffic for the EID-notify message sent between the first hop router and the ITR or ETR. The ASA maintains an EID table that correlates the EID and the site ID.

Step 3 Enable Flow Mobility for a traffic class:

a) Configure the extended ACL to identify business critical traffic that you want to re-assign to the most optimal site when servers change sites:

access list *flow_acl_name* **extended permit udp** *source_address mask destination_address mask* **eq** *port*

Both IPv4 and IPv6 ACLs are accepted. See the command reference for exact **access-list extended** syntax. You should enable flow mobility on business-critical traffic. For example, you can limit flow mobility to only HTTPS traffic, and/or to traffic to specific servers.

b) Create a class map for the ACL:

class-map *flow_map_name*

match access-list *flow_acl_name*

c) Specify the same policy map on which you enabled LISP inspection, the flow class map, and enable flow mobility:

policy-map *policy_map_name*

class *flow_map_name*

cluster flow-mobility lisp

Example:

```
ciscoasa(config)# access-list IMPORTANT-FLOWS extended permit tcp any 10.10.10.0 255.255.255.0
 eq https
ciscoasa(config)# class-map IMPORTANT-FLOWS-MAP
ciscoasa(config)# match access-list IMPORTANT-FLOWS
ciscoasa(config-cmap)# policy-map INSIDE_POLICY
ciscoasa(config-pmap)# class IMPORTANT-FLOWS-MAP
ciscoasa(config-pmap-c)# cluster flow-mobility lisp
```

Step 4 Enter cluster group configuration mode, and enable flow mobility for the cluster:

cluster group *name*

flow-mobility lisp

This on/off toggle lets you easily enable or disable flow mobility.

Examples

The following example:

- Limits EIDs to those on the 10.10.10.0/24 network
- Inspects LISP traffic (UDP 4342) between a LISP router at 192.168.50.89 (on inside) and an ITR or ETR router (on another ASA interface) at 192.168.10.8
- Enables flow mobility for all inside traffic going to a server on 10.10.10.0/24 using HTTPS.
- Enables flow mobility for the cluster.

```
access-list TRACKED_EID_LISP extended permit ip any 10.10.10.0 255.255.255.0
policy-map type inspect lisp LISP_EID_INSPECT
   parameters
      allowed-eid access-list TRACKED_EID_LISP
      validate-key MadMaxShinyandChrome
!
access-list LISP_ACL extended permit udp host 192.168.50.89 host 192.168.10.8 eq 4342
class-map LISP_CLASS
   match access-list LISP_ACL
policy-map INSIDE_POLICY
   class LISP_CLASS
      inspect lisp LISP_EID_INSPECT
service-policy INSIDE_POLICY interface inside
!
access-list IMPORTANT-FLOWS extended permit tcp any 10.10.10.0 255.255.255.0 eq https
class-map IMPORTANT-FLOWS-MAP
   match access-list IMPORTANT-FLOWS
policy-map INSIDE_POLICY
   class IMPORTANT-FLOWS-MAP
      cluster flow-mobility lisp
!
cluster group cluster1
   flow-mobility lisp
```

Configure Distributed Site-to-Site VPN

By default, the cluster uses centralized site-to-site VPN mode. To take advantage of the scalability of clustering, you can enable distributed site-to-site VPN mode.

About Distributed Site-to-Site VPN

In distributed mode, site-to-site IPsec IKEv2 VPN connections are distributed across nodes of a cluster. Distributing VPN connections across the nodes of a cluster allows both the capacity and throughput of the cluster to be fully utilized, significantly scaling VPN support beyond centralized VPN capabilities.

Distributed VPN Connection Roles

Connections can be load-balanced to multiple nodes of the cluster. Connection roles determine how connections are handled in both normal operation and in a high availability situation. When running in distributed VPN mode, the following roles are assigned to the cluster nodes:

- Active Session Owner—The node that initially receives the connection or that has transitioned a backup session to an active session. The owner maintains the state and processes packets for the complete session, including the IKE and IPsec tunnels and all traffic associated with them.
- Backup Session Owner—The node that is handling the backup session for an existing active session. If the active session owner fails, the backup session owner becomes the active session owner, and a new backup session is established on a different node.
- Forwarder—If traffic associated with a VPN session is sent to a node that does not own the VPN session, that node will use the cluster control link to forward the traffic to the node that owns the VPN session.
- Orchestrator—The orchestrator (always the control node of the cluster) is responsible for calculating which sessions will move and where they will move to when executing an Active Session Redistribution (ASR). It sends a request to the owner node X to move N sessions to node Y. Node X will respond back to the orchestrator when complete, specifying how many sessions it was able to move.

Distributed VPN Session Characteristics

Distributed site-to-site VPN Sessions have the following characteristics. Otherwise, VPN connections behave as they normally do if not on a cluster.

- VPN sessions are distributed across the cluster at the session level. Meaning the same cluster node handles the IKE and IPsec tunnels and all their traffic for a VPN connection. If VPN session traffic is sent to a cluster node that does not own that VPN session, traffic is forwarded to the cluster node that owns the VPN session.
- VPN sessions have a Session ID that is unique across the cluster. Using the session ID, traffic is validated, forwarding decisions are made, and IKE negotiation is completed.
- In a site-to-site VPN hub and spoke configuration, when clients connect through the cluster (called hair-pinning), the session traffic flowing in and the session traffic flowing out may be on different cluster nodes.
- You can require that the backup session to be allocated on a security module in another chassis; this provides protection against chassis failure. Or, you can choose to allocate backup sessions on any node in the cluster; this provides protection against node failure only. When there are two chassis in the cluster, remote-chassis backup is strongly recommended.

Distributed VPN Handling of Cluster Events

Event	Distributed VPN
Node failure	For all active sessions on this failed node, the backup sessions (on another node) become active, and backup sessions are reallocated on another node according to the backup strategy.

Event	Distributed VPN
Chassis failure	When a remote-chassis backup strategy is being used, for all active sessions on the failed chassis, the backup sessions (on a node in the other chassis) become active. When the nodes are replaced, backup sessions for these now active sessions will be reallocated on nodes in the replaced chassis. When a flat backup strategy is being used, if both the active and backup sessions are on the failed chassis, the connection will drop. All active sessions with backup sessions on a node in the other chassis fall back to these sessions. New backup sessions will be allocated on another node in the surviving chassis.
Inactivate a cluster node	For all active sessions on the cluster node being inactivated, backup sessions (on another node) become active and reallocate backup sessions on another node according to the backup strategy.
Cluster node join	If the VPN cluster mode on the new node is not set to distributed, the control node will request a mode change. After the VPN mode is compatible, the cluster node will be assigned active and backup sessions in the flow of normal operations.

IPsec IKEv2 Modifications

IKEv2 is modified while in distributed site-to-site VPN mode in the following ways:

- An identity is used in place of IP/port tuples. This allows for proper forwarding decisions on the packets, and cleanup of previous connections that may be on other cluster members.
- The (SPI) identifiers that identify a single IKEv2 session are locally generated, random 8-byte values that are unique across the cluster. An SPI embeds a time stamp and a cluster node ID. Upon receipt of an IKE negotiation packet, if the time stamp or cluster node ID check fails, the packet is dropped and a message is logged indicating the reason.
- IKEv2 processing has been modified to prevent NAT-T negotiations from failing by being split across cluster members. A new ASP classify domain, *cluster_isakmp_redirect*, and rules are added when IKEv2 is enabled on an interface. Use the **show asp table classify domain cluster_isakmp_redirect** command to see the rules.

High Availability for Distributed Site-to-Site VPN Within the Cluster

The following capabilities provide resiliency against single failure of a security module or chassis:

- VPN Sessions that are backed up on another security module in the cluster, on any chassis, withstand security module failures.
- VPN Sessions that are backed up on another chassis withstand chassis failures.
- The control node can change without losing VPN site-to-site sessions.

If an additional failure occurs before the cluster has stabilized, connections may be lost if the both active and backup sessions are on the failed nodes.

All attempts are made to ensure no sessions are lost when a node leaves the cluster in a graceful manner such as disabling the VPN cluster mode, reloading a cluster node, and other anticipated chassis changes. During these types of operations, sessions will not be lost as long as the cluster is given time to re-establish session

backups between operations. If a graceful exit is triggered on the last cluster node, it will gracefully tear down existing sessions.

CMPv2

The CMPv2 ID certificate and key pairs are synchronized across the cluster nodes. However, only the control node in the cluster automatically renews and rekeys the CMPv2 certificate. The control node synchronizes these new ID certificates and keys to all cluster nodes on a renewal. In this way, all nodes in the cluster utilize the CMPv2 certificates for authentication, and also any node is capable of taking over as the control node.

Licenses for Distributed Site-to-Site VPN

A Carrier license is required for distributed site-to-site VPN, on each member of the cluster.

Each VPN connection requires two *Other VPN* licensed sessions (the *Other VPN* license is part of the *Standard* license), one for the active session and one for the backup session. The maximum VPN session capacity of the cluster can be no more than half of the licensed capacity due to using two licenses for each session.

Prerequisites for Distributed Site-to-Site VPN

Model Support

- Firepower 9300
- Maximum of 6 modules on up to 2 chassis. You can have different quantities of installed security modules in each chassis, although we recommend an equal distribution.

Maximum VPN Sessions

Each security module supports up to 6K VPN sessions for a maximum of approximately 36K sessions across 6 nodes.

The actual number of sessions supported on a cluster node is determined by platform capacity, allocated licenses, and per context resource allocation. When utilization is close to the limit, there may be cases where session creation fails, even though the maximum capacity has not been reached on each cluster node. This is because active session allocation is determined by external switching, and backup session allocation is determined by an internal cluster algorithm. Customers are encouraged to size their utilization accordingly and allow room for uneven distribution.

Guidelines for Distributed Site-to-Site VPN

Firewall Mode

Distributed site-to-site VPN is supported in routed mode only.

Context Mode

Distributed site-to-site VPN operates in both single and multiple context modes. However, in multiple context mode, active session redistribution is done at the system level, not at the context level. This prevents an active session associated with a context from moving to a cluster member that contains active sessions associated with a different context, unknowingly creating an unsupportable load.

Unsupported Inspections

The following types of inspections are not supported or are disabled in distributed site-to-site VPN mode:

- CTIQBE
- DCERPC
- H323, H225, and RAS
- IPsec pass-through
- MGCP
- MMP
- NetBIOS
- PPTP
- RADIUS
- RSH
- RTSP
- SCCP (Skinny)
- SUNRPC
- TFTP
- WAAS
- WCCP
- XDMCP

Additional Guidelines

- Only IKEv2 IPsec site-to-site VPN is supported in distributed site-to-site VPN mode. IKEv1 is not supported. IKEv1 site-to-site is supported in centralized VPN mode.
- Inter-site clustering is not supported.
- Dynamic PAT is not available while in distributed site-to-site VPN mode.

Enable Distributed Site-to-Site VPN

Enable distributed site-to-site VPN to take advantage of the scalability of clustering for VPN sessions.



Note Changing the VPN mode between centralized and distributed requires all nodes in the cluster to reload. Changing the backup mode is dynamic and will not terminate sessions.

Before you begin

Configure site-to-site VPN according to the VPN configuration guide.

Procedure

Step 1 On the control node of the cluster, enter cluster configuration mode.

cluster group *name*

Example:

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)#
```

Step 2 Enable distributed site-to-site VPN.

vpn-mode distributed backup flat

or

vpn-mode distributed backup remote-chassis

In **flat** backup mode, standby sessions are established on any other cluster node. This will protect users from module failures; however, chassis failure protection is not guaranteed.

In **remote-chassis** backup mode, standby sessions are established on a node of another chassis in the cluster. This will protect users from both module failures and chassis failures.

If **remote-chassis** is configured in a single chassis environment (intentionally configured or the result of a failure), no backups will be created until another chassis joins.

Example:

```
ciscoasa(cfg-cluster)# vpn-mode distributed backup remote-chassis
WARNING: Do you want to proceed with changing the vpn-mode, save the device configuration,
and initiate a reboot? [confirm]
```

Redistribute Distributed S2S VPN Sessions

Active session redistribution redistributes the active VPN session load across the cluster nodes. Due to the dynamic nature of beginning and ending sessions, active session redistribution is a best effort balancing of the sessions across all cluster nodes. Repeated redistribution actions will optimize the balance.

Redistribution can be run at any time, should be run after any topology change in the cluster, and is recommended after a new node joins the cluster. The goal of redistribution is to create a stable VPN cluster. A stable VPN cluster has an almost equal number of active and backup sessions across the nodes.

To move a session, the backup session becomes the active one and another node is selected to host a new backup session. Moving sessions is dependent on the location of the active session's backup and the number of active sessions already on that particular backup node. If the backup session node is unable to host the active session for some reason, the original node remains owner of the session.

In multiple-context mode, active session redistribution is done at the system level, not the individual context level. It is not done at the context level because an active session in one context could be moved a node that contains many more active sessions in a different context, creating more load on that cluster node.

Before you begin

- Enable system logs if you would like to monitor redistribution activity.
- This procedure must be carried out on the control unit of the cluster.

Procedure

Step 1 On the control node, view how active and backup sessions are distributed across the cluster.

show cluster vpn-sessiondb distribution

Example:

Distribution information displays as follows:

```
ciscoasa# show cluster vpn-sessiondb distribution
Member 0 (unit-1-1): active: 209; backups at: 1(111), 2(98)
Member 1 (unit-1-3): active: 204; backups at: 0(108), 2(96)
Member 2 (unit-1-2): active: 0
```

Each row contains the member ID, member name, number of active sessions, and on which members the backup sessions reside. For the example above, one would read the information as:

- Member 0 has 209 active sessions, 111 sessions are backed up on member 1, 98 sessions are backed up on member 2
- Member 1 has 204 active sessions, 108 sessions are backed up on member 0, 96 sessions are backed up on member 2
- Member 2 has NO active sessions; therefore, no cluster members are backing up sessions for this node. This member has recently joined the cluster.

Step 2 Redistribute sessions.

cluster redistribute vpn-sessiondb

This command returns immediately (with no message) while it executes in the background.

Depending on the number of sessions to redistribute and the load on the cluster, this may take some time. Syslogs containing the following phrases (and other system details not shown here) are provided as redistribution activity occurs:

Syslog Phrase	Notes
VPN session redistribution started	Control node only
Sent request to move *number* sessions from *orig-member-name* to *dest-member-name*	Control node only
Failed to send session redistribution message to *member-name*	Control node only

Syslog Phrase	Notes
Received request to move *number* sessions from *orig-member-name* to *dest-member-name*	Data node only
Moved *number* sessions to *member-name*	The number of active sessions moved to the named cluster.
Failed to receive session move response from *dest-member-name*	Control node only
VPN session completed	Control node only
Cluster topology change detected. VPN session redistribution aborted.	

Step 3 Re-enter the **show cluster vpn-sessiondb distribution** command to view the results.

FXOS: Remove a Cluster Node

The following sections describe how to remove nodes temporarily or permanently from the cluster.

Temporary Removal

A cluster node will be automatically removed from the cluster due to a hardware or network failure, for example. This removal is temporary until the conditions are rectified, and it can rejoin the cluster. You can also manually disable clustering.

To check whether a device is currently in the cluster, check the cluster status within the application using the **show cluster info** command:

```
ciscoasa# show cluster info
Clustering is not enabled
```

- Disable clustering in the application—You can disable clustering using the application CLI. Enter the **cluster remove unit** *name* command to remove any node other than the one you are logged into. The bootstrap configuration remains intact, as well as the last configuration synced from the control node, so you can later re-add the node without losing your configuration. If you enter this command on a data node to remove the control node, a new control node is elected.

  When a device becomes inactive, all data interfaces are shut down; only the Management interface can send and receive traffic. To resume traffic flow, re-enable clustering. The Management interface remains up using the IP address the node received from the bootstrap configuration. However if you reload, and the node is still inactive in the cluster (for example, you saved the configuration with clustering disabled), the Management interface is disabled.

  To reenable clustering, on the ASA enter **cluster group** *name* and then **enable**.

- Disable the application instance—At the FXOS CLI, see the following example:

```
Firepower-chassis# scope ssa
Firepower-chassis /ssa # scope slot 1
Firepower-chassis /ssa/slot # scope app-instance asa asa1
Firepower-chassis /ssa/slot/app-instance # disable
Firepower-chassis /ssa/slot/app-instance* # commit-buffer
```

```
Firepower-chassis /ssa/slot/app-instance #
```

To reenable:

```
Firepower-chassis /ssa/slot/app-instance # enable
Firepower-chassis /ssa/slot/app-instance* # commit-buffer
Firepower-chassis /ssa/slot/app-instance #
```

- Shut down the security module/engine—At the FXOS CLI, see the following example:

```
Firepower-chassis# scope service-profile server 1/1
Firepower-chassis /org/service-profile # power down soft-shut-down
Firepower-chassis /org/service-profile* # commit-buffer
Firepower-chassis /org/service-profile #
```

To power up:

```
Firepower-chassis /org/service-profile # power up
Firepower-chassis /org/service-profile* # commit-buffer
Firepower-chassis /org/service-profile #
```

- Shut down the chassis—At the FXOS CLI, see the following example:

```
Firepower-chassis# scope chassis 1
Firepower-chassis /chassis # shutdown no-prompt
```

Permanent Removal

You can permanently remove a cluster node using the following methods.

- Delete the logical device—At the FXOS CLI, see the following example:

```
Firepower-chassis# scope ssa
Firepower-chassis /ssa # delete logical-device cluster1
Firepower-chassis /ssa* # commit-buffer
Firepower-chassis /ssa #
```

- Remove the chassis or security module from service—If you remove a device from service, you can add replacement hardware as a new node of the cluster.

ASA: Manage Cluster Members

After you deploy the cluster, you can change the configuration and manage cluster members.

Become an Inactive Member

To become an inactive member of the cluster, disable clustering on the node while leaving the clustering configuration intact.

Note When an ASA becomes inactive (either manually or through a health check failure), all data interfaces are shut down; only the management-only interface can send and receive traffic. To resume traffic flow, re-enable clustering; or you can remove the node altogether from the cluster. The management interface remains up using the IP address the node received from the cluster IP pool. However if you reload, and the node is still inactive in the cluster (for example, you saved the configuration with clustering disabled), then the management interface is disabled. You must use the console port for any further configuration.

Before you begin

- You must use the console port; you cannot enable or disable clustering from a remote CLI connection.
- For multiple context mode, perform this procedure in the system execution space. If you are not already in the System configuration mode, enter the **changeto system** command.

Procedure

Step 1 Enter cluster configuration mode:

cluster group *name*

Example:

```
ciscoasa(config)# cluster group pod1
```

Step 2 Disable clustering:

no enable

If this node was the control node, a new control election takes place, and a different member becomes the control node.

The cluster configuration is maintained, so that you can enable clustering again later.

Deactivate a Unit

To deactivate a member other than the node you are logged into, perform the following steps.

Note When an ASA becomes inactive, all data interfaces are shut down; only the management-only interface can send and receive traffic. To resume traffic flow, re-enable clustering. The management interface remains up using the IP address the node received from the cluster IP pool. However if you reload, and the node is still inactive in the cluster (for example, if you saved the configuration with clustering disabled), the management interface is disabled. You must use the console port for any further configuration.

Before you begin

For multiple context mode, perform this procedure in the system execution space. If you are not already in the System configuration mode, enter the **changeto system** command.

Procedure

Remove the node from the cluster.

cluster remove unit *node_name*

The bootstrap configuration remains intact, as well as the last configuration synched from the control node, so that you can later re-add the node without losing your configuration. If you enter this command on a data node to remove the control node, a new control node is elected.

To view member names, enter **cluster remove unit ?**, or enter the **show cluster info** command.

Example:

```
ciscoasa(config)# cluster remove unit ?

Current active units in the cluster:
asa2

ciscoasa(config)# cluster remove unit asa2
WARNING: Clustering will be disabled on unit asa2. To bring it back
to the cluster please logon to that unit and re-enable clustering
```

Rejoin the Cluster

If a node was removed from the cluster, for example for a failed interface or if you manually deactivated a member, you must manually rejoin the cluster.

Before you begin

- You must use the console port to reenable clustering. Other interfaces are shut down.
- For multiple context mode, perform this procedure in the system execution space. If you are not already in the System configuration mode, enter the **changeto system** command.
- Make sure the failure is resolved before you try to rejoin the cluster.

Procedure

Step 1 At the console, enter cluster configuration mode:

cluster group *name*

Example:

```
ciscoasa(config)# cluster group pod1
```

Step 2 Enable clustering.

enable

Change the Control Unit



Caution The best method to change the control node is to disable clustering on the control node, wait for a new control election, and then re-enable clustering. If you must specify the exact node you want to become the control node, use the procedure in this section. Note, however, that for centralized features, if you force a control node change using this procedure, then all connections are dropped, and you have to re-establish the connections on the new control node.

To change the control node, perform the following steps.

Before you begin

For multiple context mode, perform this procedure in the system execution space. If you are not already in the System configuration mode, enter the **changeto system** command.

Procedure

Set a new node as the control node:

cluster master unit*node_name*

Example:

```
ciscoasa(config)# cluster master unit asa2
```

You will need to reconnect to the Main cluster IP address.

To view member names, enter **cluster master unit ?** (to see all names except the current node), or enter the **show cluster info** command.

Execute a Command Cluster-Wide

To send a command to all members in the cluster, or to a specific member, perform the following steps. Sending a **show** command to all members collects all output and displays it on the console of the current unit. (Note that alternatively there are show commands that you can enter on the control unit to view cluster-wide statistics.) Other commands, such as **capture** and **copy**, can also take advantage of cluster-wide execution.

Procedure

Send a command to all members, or if you specify the unit name, a specific member:

cluster exec [unit *unit_name*] *command*

Example:

```
ciscoasa# cluster exec show xlate
```

To view member names, enter **cluster exec unit ?** (to see all names except the current unit), or enter the **show cluster info** command.

Examples

To copy the same capture file from all units in the cluster at the same time to a TFTP server, enter the following command on the control unit:

```
ciscoasa# cluster exec copy /pcap capture: tftp://10.1.1.56/capture1.pcap
```

Multiple PCAP files, one from each unit, are copied to the TFTP server. The destination capture file name is automatically attached with the unit name, such as capture1_asa1.pcap, capture1_asa2.pcap, and so on. In this example, asa1 and asa2 are cluster unit names.

The following sample output for the **cluster exec show memory** command shows memory information for each member in the cluster:

```
ciscoasa# cluster exec show memory
unit-1-1(LOCAL):*******************************************************
Free memory:      108724634538 bytes (92%)
Used memory:        9410087158 bytes ( 8%)
-------------     ------------------
Total memory:     118111600640 bytes (100%)


unit-1-3:*******************************************************************
Free memory:      108749922170 bytes (92%)
Used memory:        9371097334 bytes ( 8%)
-------------     ------------------
Total memory:     118111600640 bytes (100%)


unit-1-2:*******************************************************************
Free memory:      108426753537 bytes (92%)
Used memory:        9697869087 bytes ( 8%)
-------------     ------------------
Total memory:     118111600640 bytes (100%)
```

ASA: Monitoring the ASA Cluster on the Firepower 4100/9300 chassis

You can monitor and troubleshoot cluster status and connections.

Note When comparing the packets on the cluster control link in ASA vs. FXOS, the FXOS packet count is higher than what is shown in ASA. In ASA, only packets destined for the cluster control link IP address are counted towards input packets. Forwarded packets, which are re-injected to the data interfaces, only get counted towards the *input* statistics of data interfaces and the *output* statistics of the cluster control link.

Monitoring Cluster Status

See the following commands for monitoring cluster status:

- **show cluster info [health], show cluster chassis info**

  With no keywords, the **show cluster info** command shows the status of all members of the cluster.

  The **show cluster info health** command shows the current health of interfaces, units, and the cluster overall.

  See the following output for the **show cluster info** command:

```
asa(config)# show cluster info
Cluster cluster1: On
    Interface mode: spanned
    This is "unit-1-2" in state MASTER
        ID        : 2
        Version   : 9.5(2)
        Serial No.: FCH183770GD
        CCL IP    : 127.2.1.2
        CCL MAC   : 0015.c500.019f
        Last join : 01:18:34 UTC Nov 4 2015
        Last leave: N/A
Other members in the cluster:
    Unit "unit-1-3" in state SLAVE
        ID        : 4
        Version   : 9.5(2)
        Serial No.: FCH19057ML0
        CCL IP    : 127.2.1.3
        CCL MAC   : 0015.c500.018f
        Last join : 20:29:57 UTC Nov 4 2015
        Last leave: 20:24:55 UTC Nov 4 2015
    Unit "unit-1-1" in state SLAVE
        ID        : 1
        Version   : 9.5(2)
        Serial No.: FCH19057ML0
        CCL IP    : 127.2.1.1
        CCL MAC   : 0015.c500.017f
        Last join : 20:20:53 UTC Nov 4 2015
        Last leave: 20:18:15 UTC Nov 4 2015
    Unit "unit-2-1" in state SLAVE
```

```
ID        : 3
Version   : 9.5(2)
Serial No.: FCH19057ML0
CCL IP    : 127.2.2.1
CCL MAC   : 0015.c500.020f
Last join : 20:19:57 UTC Nov 4 2015
Last leave: 20:24:55 UTC Nov 4 2015
```

- **show cluster info auto-join**

  Shows whether the cluster unit will automatically rejoin the cluster after a time delay and if the failure conditions (such as waiting for the license, chassis health check failure, and so on) are cleared. If the unit is permanently disabled, or if the unit is already in the cluster, then this command will not show any output.

  See the following outputs for the **show cluster info auto-join** command:

```
ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster in 253 seconds.
Quit reason: Received control message DISABLE

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Master has application down that slave has up.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Chassis-blade health check failed.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Service chain application became down.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Unit is kicked out from cluster because of Application health check failure.


ciscoasa(cfg-cluster)# show cluster info auto-join
Unit join is pending (waiting for the smart license entitlement: ent1)

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit join is pending (waiting for the smart license export control flag)
```

- **show cluster info transport** {**asp** | **cp** [**detail**]}

  Shows transport related statistics for the following:

  - **asp** —Data plane transport statistics.
  - **cp** —Control plane transport statistics.

    If you enter the **detail** keyword, you can view cluster reliable transport protocol usage so you can identify packet drop issues when the buffer is full in the control plane. See the following output for the **show cluster info transport cp detail** command:

```
ciscoasa# show cluster info transport cp detail
Member ID to name mapping:
  0 - unit-1-1   2 - unit-4-1   3 - unit-2-1
```

```
Legend:
  U    - unreliable messages
  UE   - unreliable messages error
  SN   - sequence number
  ESN  - expecting sequence number
  R    - reliable messages
  RE   - reliable messages error
  RDC  - reliable message deliveries confirmed
  RA   - reliable ack packets received
  RFR  - reliable fast retransmits
  RTR  - reliable timer-based retransmits
  RDP  - reliable message dropped
  RDPR - reliable message drops reported
  RI   - reliable message with old sequence number
  RO   - reliable message with out of order sequence number
  ROW  - reliable message with out of window sequence number
  ROB  - out of order reliable messages buffered
  RAS  - reliable ack packets sent

This unit as a sender
--------------------------
        all       0         2         3
  U     123301    3867966   3230662   3850381
  UE    0         0         0         0
  SN    1656a4ce  acb26fe   5f839f76  7b680831
  R     733840    1042168   852285    867311
  RE    0         0         0         0
  RDC   699789    934969    740874    756490
  RA    385525    281198    204021    205384
  RFR   27626     56397     0         0
  RTR   34051     107199    111411    110821
  RDP   0         0         0         0
  RDPR  0         0         0         0

This unit as a receiver of broadcast messages
----------------------------------------------
        0         2         3
  U     111847    121862    120029
  R     7503      665700    749288
  ESN   5d75b4b3  6d81d23   365ddd50
  RI    630       34278     40291
  RO    0         582       850
  ROW   0         566       850
  ROB   0         16        0
  RAS   1571      123289    142256

This unit as a receiver of unicast messages
-------------------------------------------
        0         2         3
  U     1         3308122   4370233
  R     513846    879979    1009492
  ESN   4458903a  6d841a84  7b4e7fa7
  RI    66024     108924    102114
  RO    0         0         0
  ROW   0         0         0
  ROB   0         0         0
  RAS   130258    218924    228303

Gated Tx Buffered Message Statistics
------------------------------------
    current sequence number: 0

    total:                   0
    current:                 0
```

```
    high watermark:          0

    delivered:               0
    deliver failures:        0

    buffer full drops:       0
    message truncate drops:  0

    gate close ref count:    0

    num of supported clients:45

MRT Tx of broadcast messages
============================
Message high watermark: 3%
  Total messages buffered at high watermark: 5677
  [Per-client message usage at high watermark]
  ---------------------------------------------------------------
  Client name                          Total messages  Percentage
  Cluster Redirect Client                        4153         73%
  Route Cluster Client                            419          7%
  RRI Cluster Client                             1105         19%

Current MRT buffer usage: 0%
  Total messages buffered in real-time: 1
  [Per-client message usage in real-time]
  Legend:
        F - MRT messages sending when buffer is full
        L - MRT messages sending when cluster node leave
        R - MRT messages sending in Rx thread
  ------------------------------------------------------------------------------
  Client name                          Total messages  Percentage    F   L   R
  VPN Clustering HA Client                          1        100%    0   0   0

MRT Tx of unitcast messages(to member_id:0)
===========================================
Message high watermark: 31%
  Total messages buffered at high watermark: 4059
  [Per-client message usage at high watermark]
  ---------------------------------------------------------------
  Client name                          Total messages  Percentage
  Cluster Redirect Client                        3731         91%
  RRI Cluster Client                              328          8%

Current MRT buffer usage: 29%
  Total messages buffered in real-time: 3924
  [Per-client message usage in real-time]
  Legend:
        F - MRT messages sending when buffer is full
        L - MRT messages sending when cluster node leave
        R - MRT messages sending in Rx thread
  ------------------------------------------------------------------------------
  Client name                          Total messages  Percentage    F   L   R
  Cluster Redirect Client                        3607         91%    0   0   0
  RRI Cluster Client                              317          8%    0   0   0

MRT Tx of unitcast messages(to member_id:2)
===========================================
Message high watermark: 14%
  Total messages buffered at high watermark: 578
  [Per-client message usage at high watermark]
  ---------------------------------------------------------------
  Client name                          Total messages  Percentage
  VPN Clustering HA Client                        578        100%
```

```
Current MRT buffer usage: 0%
  Total messages buffered in real-time: 0

MRT Tx of unitcast messages(to member_id:3)
===========================================
Message high watermark: 12%
  Total messages buffered at high watermark: 573
  [Per-client message usage at high watermark]
  ------------------------------------------------------------------
  Client name                          Total messages  Percentage
  VPN Clustering HA Client                        572         99%
  Cluster VPN Unique ID Client                      1          0%

Current MRT buffer usage: 0%
  Total messages buffered in real-time: 0
```

- **show cluster history**

  Shows the cluster history, as well as error messages about why a cluster unit failed to join or why a unit left the cluster.

Capturing Packets Cluster-Wide

See the following command for capturing packets in a cluster:

cluster exec capture

To support cluster-wide troubleshooting, you can enable capture of cluster-specific traffic on the control node using the **cluster exec capture** command, which is then automatically enabled on all of the data nodes in the cluster.

Monitoring Cluster Resources

See the following command for monitoring cluster resources:

show cluster {cpu | memory | resource} [*options*], **show cluster chassis [cpu | memory | resource usage]**

Displays aggregated data for the entire cluster. The options available depends on the data type.

Monitoring Cluster Traffic

See the following command for monitoring cluster traffic:

- **show conn [detail | count], cluster exec show conn**

  The **show conn** command shows whether a flow is a director, backup, or forwarder flow. Use the **cluster exec show conn** command on any unit to view all connections. This command can show how traffic for a single flow arrives at different ASAs in the cluster. The throughput of the cluster is dependent on the efficiency and configuration of load balancing. This command provides an easy way to view how traffic for a connection is flowing through the cluster, and can help you understand how a load balancer might affect the performance of a flow.

  The following is sample output for the **show conn detail** command:

```
ciscoasa/ASA2/slave# show conn detail
15 in use, 21 most used
Cluster:
        fwd connections: 0 in use, 0 most used
        dir connections: 0 in use, 0 most used
        centralized connections: 0 in use, 44 most used
Flags: A - awaiting inside ACK to SYN, a - awaiting outside ACK to SYN,
       B - initial SYN from outside, b - TCP state-bypass or nailed,
       C - CTIQBE media, c - cluster centralized,
       D - DNS, d - dump, E - outside back connection, e - semi-distributed,
       F - outside FIN, f - inside FIN,
       G - group, g - MGCP, H - H.323, h - H.225.0, I - inbound data,
       i - incomplete, J - GTP, j - GTP data, K - GTP t3-response
       k - Skinny media, L - LISP triggered flow owner mobility
       M - SMTP data, m - SIP media, n - GUP
       N - inspected by Snort
       O - outbound data, o - offloaded,
       P - inside back connection,
       Q - Diameter, q - SQL*Net data,
       R - outside acknowledged FIN,
       R - UDP SUNRPC, r - inside acknowledged FIN, S - awaiting inside SYN,
       s - awaiting outside SYN, T - SIP, t - SIP transient, U - up,
       V - VPN orphan, W - WAAS,
       w - secondary domain backup,
       X - inspected by service module,
       x - per session, Y - director stub flow, y - backup stub flow,
       Z - Scansafe redirection, z - forwarding stub flow

Cluster units to ID mappings:
  ID 0: unit-2-1
  ID 1: unit-1-1
  ID 2: unit-1-2
  ID 3: unit-2-2
  ID 4: unit-2-3
  ID 255: The default cluster member ID which indicates no ownership or affiliation
          with an existing cluster member
```

- **show cluster info [conn-distribution | packet-distribution | loadbalance]**

  The **show cluster info conn-distribution** and **show cluster info packet-distribution** commands show traffic distribution across all cluster units. These commands can help you to evaluate and adjust the external load balancer.

  The **show cluster info loadbalance** command shows connection rebalance statistics.

- **show cluster info load-monitor** [**details**]

  The **show cluster info load-monitor** command shows the traffic load for cluster members for the last interval and also the average over total number of intervals configured (30 by default). Use the **details** keyword to view the value for each measure at each interval.

```
ciscoasa(cfg-cluster)# show cluster info load-monitor
ID  Unit Name
0  B
1  A_1
Information from all units with 20 second interval:
Unit     Connections    Buffer Drops   Memory Used   CPU Used
Average from last 1 interval:
   0           0            0            14              25
   1           0            0            16              20
```

```
Average from last 30 interval:
   0           0           0           12              28
   1           0           0           13              27


ciscoasa(cfg-cluster)# show cluster info load-monitor details


ID  Unit Name

 0  B

 1  A_1

Information from all units with 20 second interval


Connection count captured over 30 intervals:

Unit ID  0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

Unit ID  1

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0


Buffer drops captured over 30 intervals:

Unit ID  0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

Unit ID  1

         0          0          0          0          0          0
```

```
          0          0          0          0          0          0
          0          0          0          0          0          0
          0          0          0          0          0          0
          0          0          0          0          0          0

Memory usage(%) captured over 30 intervals:
Unit ID  0
         25         25         30         30         30         35
         25         25         35         30         30         30
         25         25         30         25         25         35
         30         30         30         25         25         25
         25         20         30         30         30         30
Unit ID  1
         30         25         35         25         30         30
         25         25         35         25         30         35
         30         30         35         30         30         30
         25         20         30         25         25         30
         20         30         35         30         30         35

CPU usage(%) captured over 30 intervals:
Unit ID  0
         25         25         30         30         30         35
         25         25         35         30         30         30
         25         25         30         25         25         35
         30         30         30         25         25         25
         25         20         30         30         30         30
Unit ID  1
         30         25         35         25         30         30
         25         25         35         25         30         35
         30         30         35         30         30         30
         25         20         30         25         25         30
         20         30         35         30         30         35
```

- **show cluster {access-list | conn [count] | traffic | user-identity | xlate}** [*options*], **show cluster chassis {access-list | conn | traffic | user-identity | xlate count}**

  Displays aggregated data for the entire cluster. The options available depends on the data type.

  See the following output for the **show cluster access-list** command:

```
ciscoasa# show cluster access-list
hitcnt display order: cluster-wide aggregated result, unit-A, unit-B,  unit-C, unit-D
access-list cached ACL log flows: total 0, denied 0 (deny-flow-max 4096) alert-interval
 300
access-list 101; 122 elements; name hash: 0xe7d586b5
access-list 101 line 1 extended permit tcp 192.168.143.0 255.255.255.0 any eq www
(hitcnt=0, 0, 0, 0, 0) 0x207a2b7d
access-list 101 line 2 extended permit tcp any 192.168.143.0 255.255.255.0 (hitcnt=0,
0, 0, 0, 0) 0xfe4f4947
access-list  101 line 3 extended permit tcp host 192.168.1.183 host 192.168.43.238
(hitcnt=1, 0, 0, 0, 1) 0x7b521307
access-list 101 line 4 extended permit tcp host 192.168.1.116 host 192.168.43.238
(hitcnt=0, 0, 0, 0, 0) 0x5795c069
access-list 101 line 5 extended permit tcp host 192.168.1.177 host 192.168.43.238
(hitcnt=1, 0, 0, 1, 0) 0x51bde7ee
access list 101 line 6 extended permit tcp host 192.168.1.177 host 192.168.43.13
(hitcnt=0, 0, 0, 0, 0) 0x1e68697c
access-list 101 line 7 extended permit tcp host 192.168.1.177 host 192.168.43.132
(hitcnt=2, 0, 0, 1, 1) 0xc1ce5c49
access-list 101 line 8 extended permit tcp host 192.168.1.177 host 192.168.43.192
(hitcnt=3, 0, 1, 1, 1) 0xb6f59512
access-list 101 line 9 extended permit tcp host 192.168.1.177 host 192.168.43.44
(hitcnt=0, 0, 0, 0, 0) 0xdc104200
access-list 101 line 10 extended permit tcp host 192.168.1.112 host 192.168.43.44
(hitcnt=429, 109, 107, 109, 104)
0xce4f281d
access-list 101 line 11 extended permit tcp host 192.168.1.170 host 192.168.43.238
(hitcnt=3, 1, 0, 0, 2) 0x4143a818
access-list 101 line 12 extended permit tcp host 192.168.1.170 host 192.168.43.169
(hitcnt=2, 0, 1, 0, 1) 0xb18dfea4
access-list 101 line 13 extended permit tcp host 192.168.1.170 host 192.168.43.229
(hitcnt=1, 1, 0, 0, 0) 0x21557d71
access-list 101 line 14 extended permit tcp host 192.168.1.170 host 192.168.43.106
(hitcnt=0, 0, 0, 0, 0) 0x7316e016
access-list 101 line 15 extended permit tcp host 192.168.1.170 host 192.168.43.196
(hitcnt=0, 0, 0, 0, 0) 0x013fd5b8
access-list 101 line 16 extended permit tcp host 192.168.1.170 host 192.168.43.75
(hitcnt=0, 0, 0, 0, 0) 0x2c7dba0d
```

  To display the aggregated count of in-use connections for all units, enter:

```
ciscoasa# show cluster conn count
Usage Summary In Cluster:*********************************************
124 in use, fwd connection 0 in use, dir connection 0 in use, centralized connection
0 in use (Cluster-wide aggregated)

unit-1-1(LOCAL):******************************************************
40 in use, 48 most used, fwd connection 0 in use, 0 most used, dir connection 0 in use,
0 most used, centralized connection 0 in use, 46 most used

unit-2-2:*************************************************************
18 in use, 40 most used, fwd connection 0 in use, 0 most used, dir connection 0 in use,
0 most used, centralized connection 0 in use, 45 most used
```

- **show asp cluster counter**

  This command is useful for datapath troubleshooting.

Monitoring Cluster Routing

See the following commands for cluster routing:

- **show route cluster**
- **debug route cluster**

  Shows cluster information for routing.

- **show lisp eid**

  Shows the ASA EID table showing EIDs and site IDs.

  See the following output from the **cluster exec show lisp eid** command.

```
ciscoasa# cluster exec show lisp eid
L1(LOCAL):*************************************************************
     LISP EID       Site ID
    33.44.33.105        2
    33.44.33.201        2
    11.22.11.1             4
    11.22.11.2             4
L2:*************************************************************************
     LISP EID       Site ID
    33.44.33.105        2
    33.44.33.201        2
    11.22.11.1   4
    11.22.11.2  4
```

- **show asp table classify domain inspect-lisp**

  This command is useful for troubleshooting.

Monitoring Distributed S2S VPN

Use the following commands to monitor status and distribution of the VPN sessions:

- The overall distribution of sessions is provided using **show cluster vpn-sessiondb distribution**. If running in a multiple context environment, this command must be run in the system execution space.

  This **show** command provides a quick view of the sessions, rather than having to execute **show vpn-sessiondb summary** on each node.

- A unified view of the VPN connections on the cluster using the **show cluster vpn-sessiondb summary** command is also available.
- Individual device monitoring using the **show vpn-sessiondb** command shows the number of active and backup sessions on a device in addition to the usual VPN information.

Configuring Logging for Clustering

See the following command for configuring logging for clustering:

logging device-id

Each node in the cluster generates syslog messages independently. You can use the **logging device-id** command to generate syslog messages with identical or different device IDs to make messages appear to come from the same or different nodes in the cluster.

Debugging Clustering

See the following commands for debugging clustering:

- **debug cluster [ccp | datapath | fsm | general | hc | license | rpc | service-module | transport]**

  Shows debug messages for clustering.

- **debug service-module**

  Shows debug messages for blade level issues including health check issues between the supervisor and the application.

- **show cluster info trace**

  The **show cluster info trace** command shows the debug information for further troubleshooting.

  See the following output for the **show cluster info trace** command:

```
ciscoasa# show cluster info trace
 Feb 02 14:19:47.456 [DBUG]Receive CCP message: CCP_MSG_LOAD_BALANCE
 Feb 02 14:19:47.456 [DBUG]Receive CCP message: CCP_MSG_LOAD_BALANCE
 Feb 02 14:19:47.456 [DBUG]Send CCP message to all: CCP_MSG_KEEPALIVE from 80-1 at
MASTER
```

  For example, if you see the following messages showing that two nodes with the same **local-unit** name are acting as the control node, it could mean that either two nodes have the same **local-unit** name (check your configuration), or a node is receiving its own broadcast messages (check your network).

```
ciscoasa# show cluster info trace
 May 23 07:27:23.113 [CRIT]Received datapath event 'multi masters' with parameter 1.
 May 23 07:27:23.113 [CRIT]Found both unit-9-1 and unit-9-1 as master units. Master
role retained by unit-9-1, unit-9-1 will leave then join as a Slave
 May 23 07:27:23.113 [DBUG]Send event (DISABLE, RESTART | INTERNAL-EVENT, 5000 msecs,
Detected another Master, leave and re-join as Slave) to FSM. Current state MASTER
 May 23 07:27:23.113 [INFO]State machine changed from state MASTER to DISABLED
```

Troubleshooting Distributed S2S VPN

Distributed VPN Notifications

You will be notified with messages containing the identified phrases when the following error situations occur on a cluster running distributed VPN:

Situation	Notification

If an existing or joining cluster data node is not in distributed VPN mode when attempting to join the cluster:	`New cluster member (member-name) rejected due to vpn mode mismatch.`
	and
	`Master (control-name) rejects enrollment request from unit (unit-name) for the reason: the vpn mode capabilities are not compatible with the master configuration`

If licensing is not properly configured on a cluster member for Distributed VPN:	`ERROR: Master requested cluster vpn-mode change to distributed. Unable to change mode due to missing Carrier License.`

If the time stamp or member ID is invalid in the SPI of a received IKEv2 packet:	`Expired SPI received`
	or
	`Corrupted SPI detected`

If the cluster is unable to create a backup session:	`Failed to create the backup for an IKEv2 session.`
IKEv2 Initial Contact (IC) processing error:	`IKEv2 Negotiation aborted due to ERROR: Stale backup session found on backup`
Redistribution problems:	`Failed to send session redistribution message to member-name` `Failed to receive session move response from member-name (master only)`
If the topology changes during redistribution of the sessions:	`Cluster topology change detected. VPN session redistribution aborted.`

You may be encountering one of the following situations:

- Site-to-site VPN sessions are being distributed to only one of the chassis in a cluster when the Nexus 7K switch is configured with a layer 4 port as a load-balancing algorithm using the **port-channel load-balance src-dst l4port** command. An example of the cluster session allocation looks like below:

```
SSP-Cluster/slave(cfg-cluster)# show cluster vpn-sessiondb distribution
Member 0 (unit-1-3): active: 0
Member 1 (unit-2-2): active: 13295; backups at: 0(2536), 2(2769), 3(2495), 4(2835),
5(2660)
Member 2 (unit-2-3): active: 12174; backups at: 0(2074), 1(2687), 3(2207), 4(3084),
5(2122)
Member 3 (unit-2-1): active: 13416; backups at: 0(2419), 1(3013), 2(2712), 4(2771),
5(2501)
Member 4 (unit-1-1): active: 0
Member 5 (unit-1-2): active: 0
```

  Since site-to-site IKEv2 VPN uses port 500 for both source and destination ports, IKE packets are only sent to one of the links in the port channel connected between the Nexus 7K and the chassis.

Change the Nexus 7K load balancing algorithm to IP and Layer 4 port using the **port-channel load-balance src-dst ip-l4port**. Then the IKE packets are sent to all the links and thus all nodes.

For a more immediate adjustment, on the control node of the cluster, execute: **cluster redistribute vpn-sessiondb** to redistribute active VPN sessions to the cluster nodes of the other chassis.

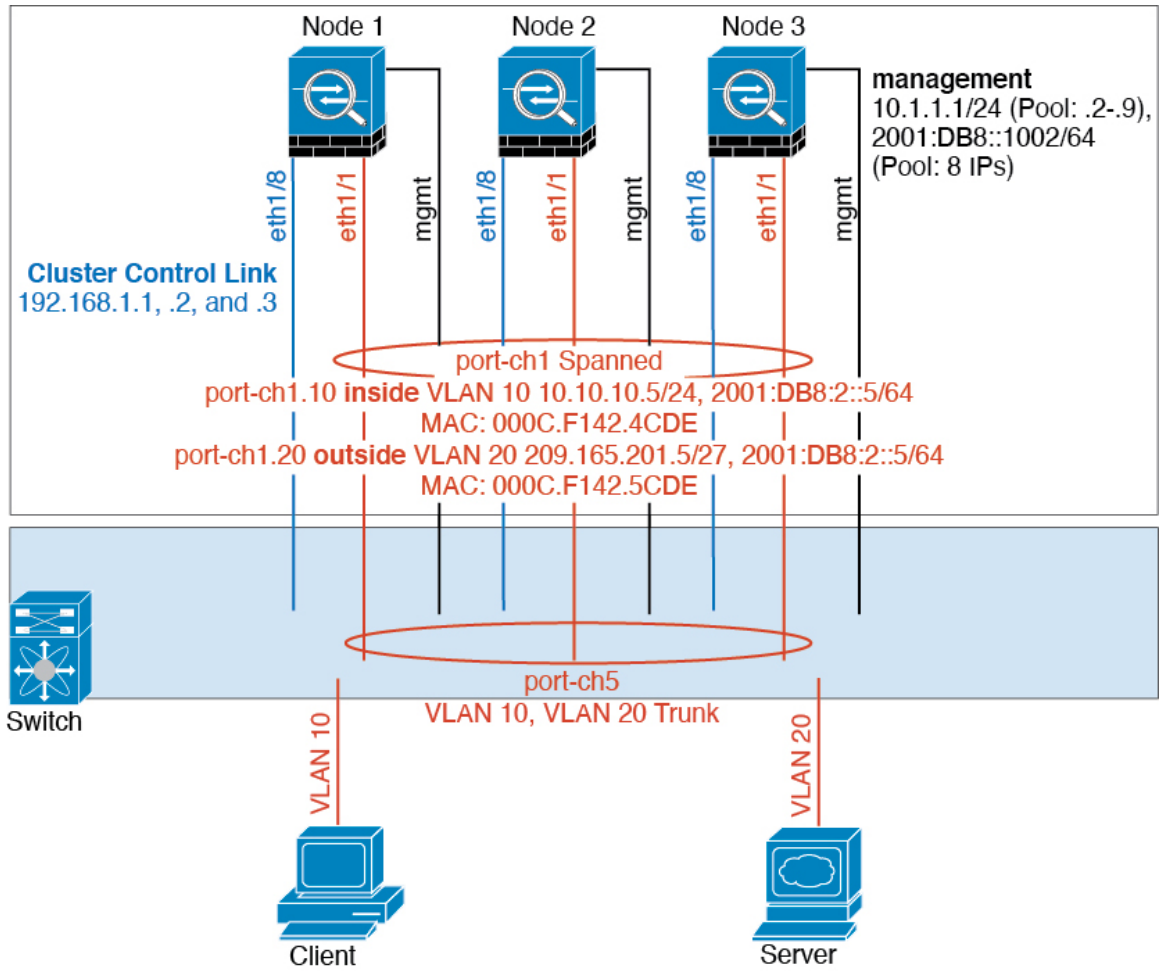
Examples for ASA Clustering

These examples include typical deployments.

Firewall on a stick

Data traffic from different security domains are associated with different VLANs. For example, VLAN 10 for the inside network and VLAN 20 for the outside network. Each ASA has a single physical port connected to the external switch or router. Trunking is enabled so that all packets on the physical link are 802.1q encapsulated. The ASA is the firewall between VLAN 10 and VLAN 20.

When using Spanned EtherChannels, all data links are grouped into one EtherChannel on the switch side. If the ASA becomes unavailable, the switch will rebalance traffic between the remaining units.



Interface mode on each unit

```
cluster interface-mode spanned force
```

Unit 1 control unit bootstrap configuration

```
interface ethernet1/8
no shutdown
description CCL

cluster group cluster1
local-unit asa1
cluster-interface ethernet1/8 ip 192.168.1.1 255.255.255.0
priority 1
key chuntheunavoidable
enable noconfirm
```

Unit 2 data unit bootstrap configuration

```
interface ethernet1/8
no shutdown
description CCL

cluster group cluster1
local-unit asa2
cluster-interface ethernet1/8 ip 192.168.1.2 255.255.255.0
priority 2
key chuntheunavoidable
enable as-slave
```

Unit 3 data unit bootstrap configuration

```
interface ethernet1/8
no shutdown
description CCL

cluster group cluster1
local-unit asa3
cluster-interface ethernet1/8 ip 192.168.1.3 255.255.255.0
priority 3
key chuntheunavoidable
enable as-slave
```

Control unit interface configuration

```
ip local pool mgmt 10.1.1.2-10.1.1.9
ipv6 local pool mgmtipv6 2001:DB8::1002/64 8

interface management 1/1
nameif management
ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
ipv6 address 2001:DB8::1001/32 cluster-pool mgmtipv6
security-level 100
management-only
no shutdown

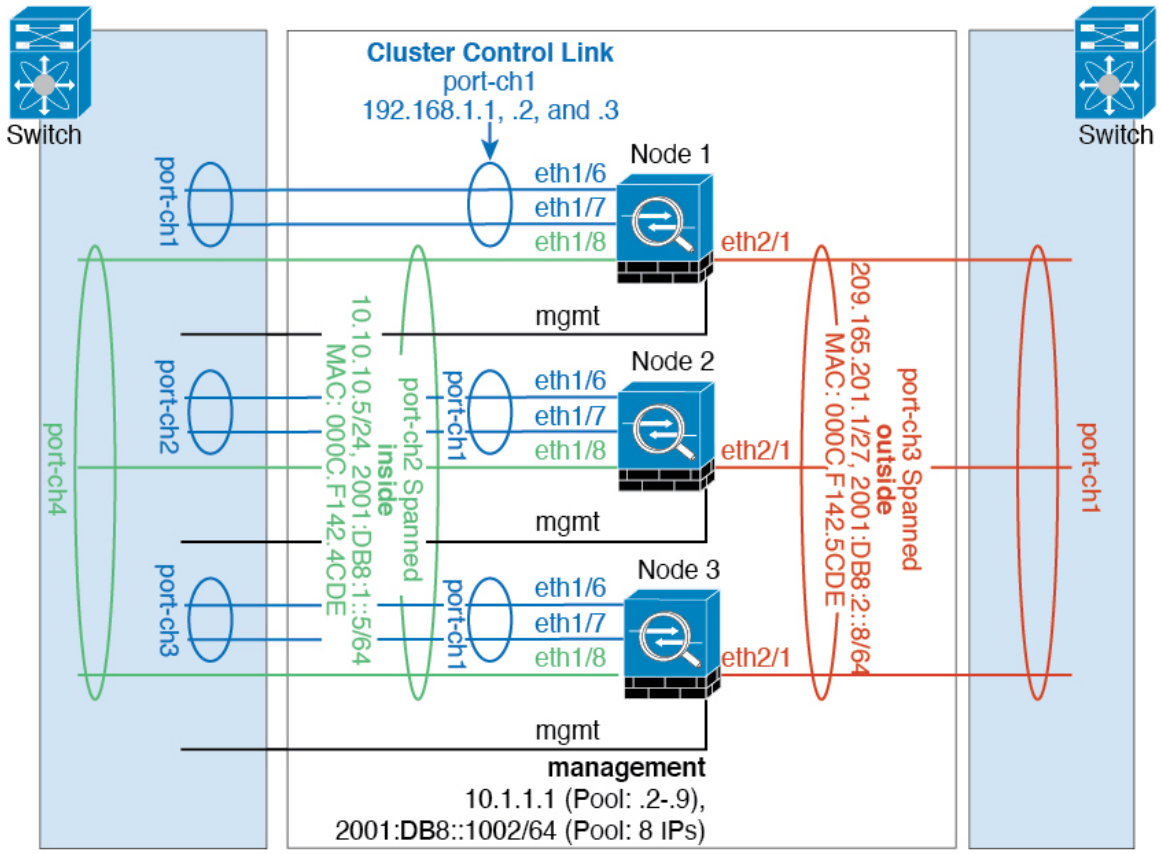
interface ethernet1/1
channel-group 1 mode active
no shutdown

interface port-channel 1

interface port-channel 1.10
vlan 10
nameif inside
ip address 10.10.10.5 255.255.255.0
ipv6 address 2001:DB8:1::5/64
mac-address 000C.F142.4CDE

interface port-channel 1.20
vlan 20
nameif outside
ip address 209.165.201.1 255.255.255.224
ipv6 address 2001:DB8:2::8/64
mac-address 000C.F142.5CDE
```

Traffic Segregation



As shown in the diagram above, there is one Spanned EtherChannel on the left side that connects to the inside switch, and the other on the right side to outside switch. You can also create VLAN subinterfaces on each EtherChannel if desired.

Interface mode on each unit

```
cluster interface-mode spanned force
```

Unit 1 control unit bootstrap configuration

```
interface ethernet 1/6
 channel-group 1 mode on
 no shutdown

interface ethernet 1/7
 channel-group 1 mode on
 no shutdown

interface port-channel 1
 description CCL

cluster group cluster1
 local-unit asa1
 cluster-interface port-channel1 ip 192.168.1.1 255.255.255.0
```

```
 priority 1
 key chuntheunavoidable
 enable noconfirm
```

Unit 2 data unit bootstrap configuration

```
interface ethernet 1/6
 channel-group 1 mode on
 no shutdown

interface ethernet 1/7
 channel-group 1 mode on
 no shutdown

interface port-channel 1
 description CCL

cluster group cluster1
 local-unit asa2
 cluster-interface port-channel1 ip 192.168.1.2 255.255.255.0
 priority 2
 key chuntheunavoidable
 enable as-slave
```

Unit 3 data unit bootstrap configuration

```
interface ethernet 1/6
 channel-group 1 mode on
 no shutdown

interface ethernet 1/7
 channel-group 1 mode on
 no shutdown

interface port-channel 1
 description CCL

cluster group cluster1
 local-unit asa3
 cluster-interface port-channel1 ip 192.168.1.3 255.255.255.0
 priority 3
 key chuntheunavoidable
 enable as-slave
```

Control unit interface configuration

```
ip local pool mgmt 10.1.1.2-10.1.1.9
ipv6 local pool mgmtipv6 2001:DB8::1002/64 8

interface management 1/1
 nameif management
 ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
 ipv6 address 2001:DB8::1001/32 cluster-pool mgmtipv6
 security-level 100
 management-only
 no shutdown

interface ethernet 1/8
 channel-group 2 mode active
 no shutdown
```

```
interface port-channel 2
 nameif inside
 ip address 10.10.10.5 255.255.255.0
 ipv6 address 2001:DB8:1::5/64
 mac-address 000C.F142.4CDE

interface ethernet 2/1
 channel-group 3 mode active
 no shutdown

interface port-channel 3
 nameif outside
 ip address 209.165.201.1 255.255.255.224
 ipv6 address 2001:DB8:2::8/64
 mac-address 000C.F142.5CDE
```

OTV Configuration for Routed Mode Inter-Site Clustering

The success of inter-site clustering for routed mode with Spanned EtherChannels depends on the proper configuration and monitoring of OTV. OTV plays a major role by forwarding the packets across the DCI. OTV forwards unicast packets across the DCI only when it learns the MAC address in its forwarding table. If the MAC address is not learned in the OTV forwarding table, it will drop the unicast packets.

Sample OTV Configuration

```
//Sample OTV config:
//3151 – Inside VLAN, 3152 – Outside VLAN, 202 – CCL VLAN
//aaaa.1111.1234 – ASA inside interface global vMAC
//0050.56A8.3D22 – Server MAC

feature ospf
feature otv

mac access-list ALL_MACs
  10 permit any any
mac access-list HSRP_VMAC
  10 permit aaaa.1111.1234 0000.0000.0000 any
  20 permit aaaa.2222.1234 0000.0000.0000 any
  30 permit any aaaa.1111.1234 0000.0000.0000
  40 permit any aaaa.2222.1234 0000.0000.0000
vlan access-map Local 10
  match mac address HSRP_VMAC
  action drop
vlan access-map Local 20
  match mac address ALL_MACs
  action forward
vlan filter Local vlan-list 3151-3152


//To block global MAC with ARP inspection:
arp access-list HSRP_VMAC_ARP
  10 deny aaaa.1111.1234 0000.0000.0000 any
  20 deny aaaa.2222.1234 0000.0000.0000 any
  30 deny any aaaa.1111.1234 0000.0000.0000
  40 deny any aaaa.2222.1234 0000.0000.0000
  50 permit ip any mac
ip arp inspection filter HSRP_VMAC_ARP 3151-3152
```

```
no ip igmp snooping optimise-multicast-flood
vlan 1,202,1111,2222,3151-3152

otv site-vlan 2222
mac-list GMAC_DENY seq 10 deny aaaa.aaaa.aaaa ffff.ffff.ffff
mac-list GMAC_DENY seq 20 deny aaaa.bbbb.bbbb ffff.ffff.ffff
mac-list GMAC_DENY seq 30 permit 0000.0000.0000 0000.0000.0000
route-map stop-GMAC permit 10
  match mac-list GMAC_DENY

interface Overlay1
  otv join-interface Ethernet8/1
  otv control-group 239.1.1.1
  otv data-group 232.1.1.0/28
  otv extend-vlan 202, 3151
  otv arp-nd timeout 60
  no shutdown

interface Ethernet8/1
  description uplink_to_OTV_cloud
  mtu 9198
  ip address 10.4.0.18/24
  ip igmp version 3
  no shutdown

interface Ethernet8/2

interface Ethernet8/3
  description back_to_default_vdc_e6/39
  switchport
    switchport mode trunk
    switchport trunk allowed vlan 202,2222,3151-3152
  mac packet-classify
  no shutdown

otv-isis default
  vpn Overlay1
    redistribute filter route-map stop-GMAC
otv site-identifier 0x2
//OTV flood not required for ARP inspection:
otv flood mac 0050.56A8.3D22 vlan 3151
```

OTV Filter Modifications Required Because of Site Failure

If a site goes down, the filters need to be removed from OTV because you do not want to block the global MAC address anymore. There are some additional configurations required.

You need to add a static entry for the ASA global MAC address on the OTV switch in the site that is functional. This entry will let the OTV at the other end add these entries on the overlay interface. This step is required because if the server and client already have the ARP entry for the ASA, which is the case for existing connections, then they will not send the ARP again. Therefore, OTV will not get a chance to learn the ASA global MAC address in its forwarding table. Because OTV does not have the global MAC address in its forwarding table, and per OTV design it will not flood unicast packets over the overlay interface, then it will drop the unicast packets to the global MAC address from the server, and the existing connections will break.

```
//OTV filter configs when one of the sites is down

mac-list GMAC_A seq 10 permit 0000.0000.0000 0000.0000.0000
route-map a-GMAC permit 10
  match mac-list GMAC_A
```

```
otv-isis default
  vpn Overlay1
    redistribute filter route-map a-GMAC

no vlan filter Local vlan-list 3151

//For ARP inspection, allow global MAC:
arp access-list HSRP_VMAC_ARP_Allow
  50 permit ip any mac
ip arp inspection filter HSRP_VMAC_ARP_Allow 3151-3152

mac address-table static aaaa.1111.1234 vlan 3151 interface Ethernet8/3
//Static entry required only in the OTV in the functioning Site
```

When the other site is restored, you need to add the filters back again and remove this static entry on the OTV. It is very important to clear the dynamic MAC address table on both the OTVs to clear the overlay entry for the global MAC address.

MAC Address Table Clearing

When a site goes down, and a static entry for the global MAC address is added to OTV, you need to let the other OTV learn the global MAC address on the overlay interface. After the other site comes up, these entries should be cleared. Make sure to clear the mac address table to make sure OTV does not have these entries in its forwarding table.

```
cluster-N7k6-OTV# show mac address-table
Legend:
* - primary entry, G - Gateway MAC, (R) - Routed MAC, O - Overlay MAC
age - seconds since last seen,+ - primary entry using vPC Peer-Link,
(T) - True, (F) - False
VLAN MAC Address Type age Secure NTFY Ports/SWID.SSID.LID
---------+-----------------+--------+---------+------+----+------------------
G -    d867.d900.2e42 static  -  F F sup-eth1(R)
O 202  885a.92f6.44a5 dynamic -  F F Overlay1
* 202  885a.92f6.4b8f dynamic 5  F F Eth8/3
O 3151 0050.5660.9412 dynamic -  F F Overlay1
* 3151 aaaa.1111.1234 dynamic 50 F F Eth8/3
```

OTV ARP Cache Monitoring

OTV maintains an ARP cache to proxy ARP for IP addresses that it learned across the OTV interface.

```
cluster-N7k6-OTV# show otv arp-nd-cache
OTV ARP/ND L3->L2 Address Mapping Cache

Overlay Interface Overlay1
VLAN MAC Address Layer-3 Address Age Expires In
3151 0050.5660.9412 10.0.0.2 1w0d 00:00:31
cluster-N7k6-OTV#
```

Examples for Inter-Site Clustering

The following examples show supported cluster deployments.

Spanned EtherChannel Routed Mode Example with Site-Specific MAC and IP Addresses

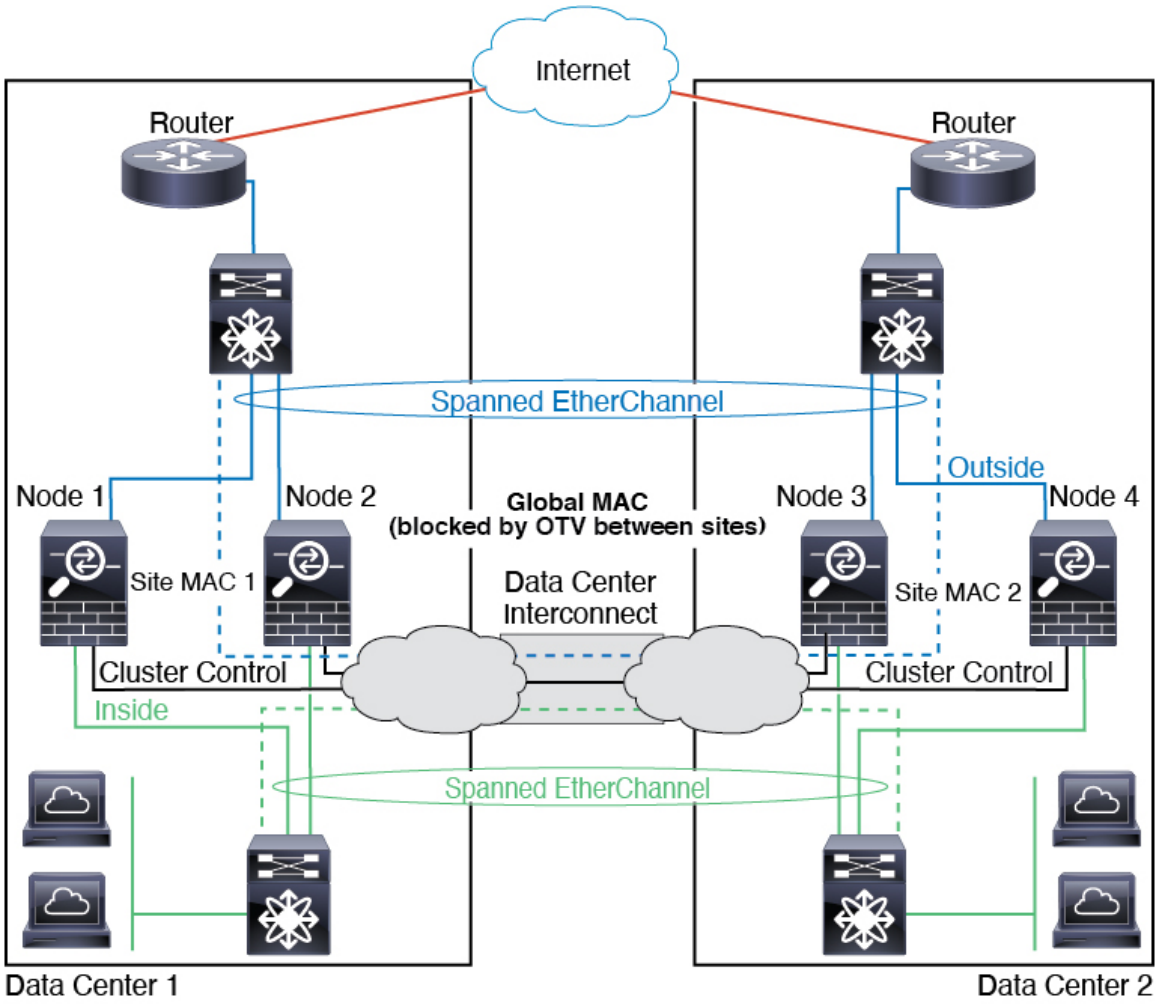
The following example shows 2 cluster members at each of 2 data centers placed between the gateway router and an inside network at each site (East-West insertion). The cluster members are connected by the cluster control link over the DCI. The cluster members at each site connect to the local switches using spanned EtherChannels for both the inside and outside networks. Each EtherChannel is spanned across all chassis in the cluster.

The data VLANs are extended between the sites using Overlay Transport Virtualization (OTV) (or something similar). You must add filters blocking the global MAC address to prevent traffic from traversing the DCI to the other site when the traffic is destined for the cluster. If the cluster nodes at one site become unreachable, you must remove the filters so traffic can be sent to the other site's cluster nodes. You should use VACLs to filter the global MAC address.For some switches, such as Nexus with the F3-series line card, you must also use ARP inspection to block ARP packets from the global MAC address. ARP inspection requires you to set both the site MAC address and the site IP address on the ASA. If you only configure the site MAC address be sure to disable ARP inspection.

The cluster acts as the gateway for the inside networks. The global virtual MAC, which is shared across all cluster nodes, is used only to receive packets. Outgoing packets use a site-specific MAC address from each DC cluster. This feature prevents the switches from learning the same global MAC address from both sites on two different ports, which causes MAC flapping; instead, they only learn the site MAC address.

In this scenario:

- All egress packets sent from the cluster use the site MAC address and are localized at the data center.
- All ingress packets to the cluster are sent using the global MAC address, so they can be received by any of the nodes at both sites; filters at the OTV localize the traffic within the data center.



Spanned EtherChannel Transparent Mode North-South Inter-Site Example

The following example shows 2 cluster members at each of 2 data centers placed between inside and outside routers (North-South insertion). The cluster members are connected by the cluster control link over the DCI. The cluster members at each site connect to the local switches using spanned EtherChannels for the inside and outside. Each EtherChannel is spanned across all chassis in the cluster.

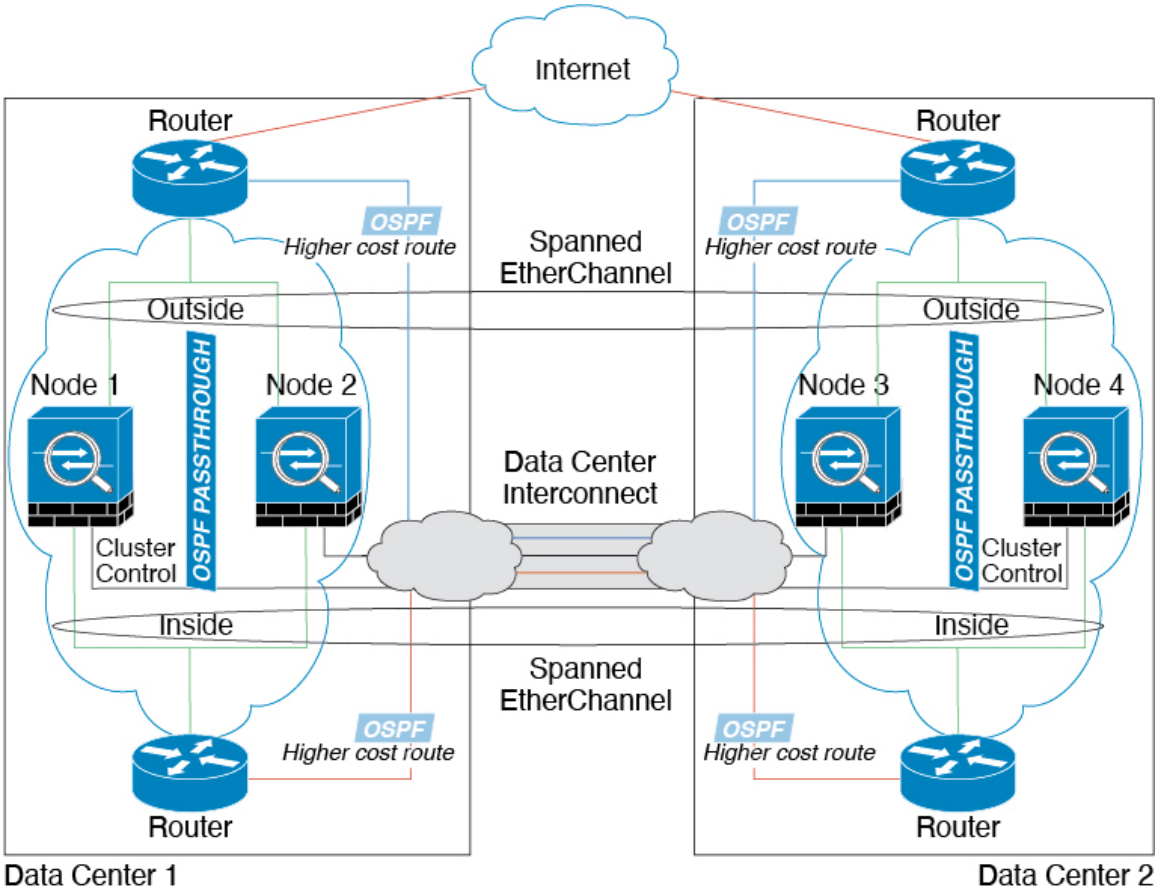
The inside and outside routers at each data center use OSPF, which is passed through the transparent ASAs. Unlike MACs, router IPs are unique on all routers. By assigning a higher cost route across the DCI, traffic stays within each data center unless all cluster members at a given site go down. The lower cost route through the ASAs must traverse the same bridge group at each site for the cluster to maintain asymmetric connections. In the event of a failure of all cluster members at one site, traffic goes from each router over the DCI to the cluster members at the other site.

The implementation of the switches at each site can include:

- Inter-site VSS, vPC, StackWise, or StackWise Virtual—In this scenario, you install one switch at Data Center 1, and the other at Data Center 2. One option is for the cluster nodes at each Data Center to only connect to the local switch, while the redundant switch traffic goes across the DCI. In this case, connections are for the most part kept local to each datacenter. You can optionally connect each node to both switches

across the DCI if the DCI can handle the extra traffic. In this case, traffic is distributed across the data centers, so it is essential for the DCI to be very robust.

- Local VSS, vPC, StackWise, or StackWise Virtual at each site—For better switch redundancy, you can install 2 separate redundant switch pairs at each site. In this case, although the cluster nodes still have a spanned EtherChannel with Data Center 1 chassis connected only to both local switches, and Data Center 2 chassis connected to those local switches, the spanned EtherChannel is essentially "split." Each local redundant switch system sees the spanned EtherChannel as a site-local EtherChannel.

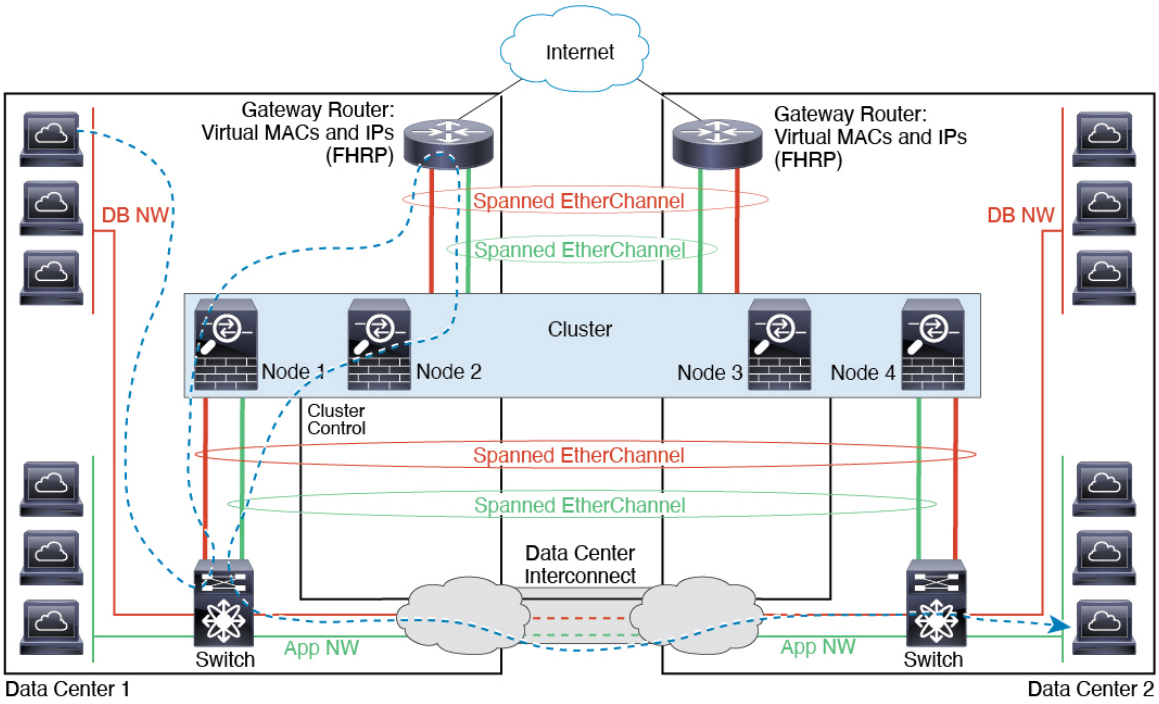


Spanned EtherChannel Transparent Mode East-West Inter-Site Example

The following example shows 2 cluster members at each of 2 data centers placed between the gateway router and two inside networks at each site, the App network and the DB network (East-West insertion). The cluster members are connected by the cluster control link over the DCI. The cluster members at each site connect to the local switches using spanned EtherChannels for both the App and DB networks on the inside and outside. Each EtherChannel is spanned across all chassis in the cluster.

The gateway router at each site uses an FHRP such as HSRP to provide the same destination virtual MAC and IP addresses at each site. A good practice to avoid unintended MAC address flapping is to statically add the gateway routers real MAC addresses to the ASA MAC address table using the **mac-address-table static** *outside_interface mac_address* command. Without these entries, if the gateway at site 1 communicates with the gateway at site 2, that traffic might pass through the ASA and attempt to reach site 2 from the inside interface and cause problems. The data VLANs are extended between the sites using Overlay Transport

Virtualization (OTV) (or something similar). You must add filters to prevent traffic from traversing the DCI to the other site when the traffic is destined for the gateway router. If the gateway router at one site becomes unreachable, you must remove the filters so traffic can be sent to the other site's gateway router.



Reference for Clustering

This section includes more information about how clustering operates.

ASA Features and Clustering

Some ASA features are not supported with ASA clustering, and some are only supported on the control node. Other features might have caveats for proper usage.

Unsupported Features with Clustering

These features cannot be configured with clustering enabled, and the commands will be rejected.

- Unified Communication features that rely on TLS Proxy
- Remote access VPN (SSL VPN and IPsec VPN)
- Virtual Tunnel Interfaces (VTIs)
- IS-IS routing
- The following application inspections:
  - CTIQBE

    - H323, H225, and RAS
    - IPsec passthrough
    - MGCP
    - MMP
    - RTSP
    - SCCP (Skinny)
    - WAAS
    - WCCP
- Botnet Traffic Filter
- Auto Update Server
- DHCP client, server, and proxy. DHCP relay is supported.
- VPN load balancing
- Failover
- Integrated Routing and Bridging
- Dead Connection Detection (DCD)
- FIPS mode

Centralized Features for Clustering

These features are only supported on the control node, and are not scaled for the cluster.



Note Traffic for centralized features is forwarded from member nodes to the control node over the cluster control link.

If you use the rebalancing feature, traffic for centralized features may be rebalanced to non-control nodes before the traffic is classified as a centralized feature; if this occurs, the traffic is then sent back to the control node.

For centralized features, if the control node fails, all connections are dropped, and you have to re-establish the connections on the new control node.

- These application inspections are supported:
    - DCERPC and ESMTP
    - IM
    - NetBIOS and PPTP
    - RADIUS
    - RSH

  - SNMP
  - SQLNET and SUNRPC
  - TFTP
  - XDMCP
- Static route monitoring
- Authentication and Authorization for network access. Accounting is decentralized.
- Filtering Services
- Site-to-site VPN

  In centralized mode, VPN connections are established with the control node of the cluster only. This is the default mode for VPN clustering. Site-to-site VPN can also be deployed in Distributed VPN Mode, where S2S IKEv2 VPN connections are distributed across nodes.
- IGMP multicast control plane protocol processing (data plane forwarding is distributed across the cluster)
- PIM multicast control plane protocol processing (data plane forwarding is distributed across the cluster)
- Dynamic routing

Features Applied to Individual Units

These features are applied to each ASA node, instead of the cluster as a whole or to the control node.

- QoS—The QoS policy is synced across the cluster as part of configuration replication. However, the policy is enforced on each node independently. For example, if you configure policing on output, then the conform rate and conform burst values are enforced on traffic exiting a particular ASA. In a cluster with 3 nodes and with traffic evenly distributed, the conform rate actually becomes 3 times the rate for the cluster.
- Threat detection—Threat detection works on each node independently; for example, the top statistics is node-specific. Port scanning detection, for example, does not work because scanning traffic will be load-balanced between all nodes, and one node will not see all traffic.
- Resource management—Resource management in multiple context mode is enforced separately on each node based on local usage.
- LISP traffic—LISP traffic on UDP port 4342 is inspected by each receiving node, but is not assigned a director. Each node adds to the EID table that is shared across the cluster, but the LISP traffic itself does not participate in cluster state sharing.

AAA for Network Access and Clustering

AAA for network access consists of three components: authentication, authorization, and accounting. Authentication and authorization are implemented as centralized features on the clustering control node with replication of the data structures to the cluster data nodes. If a control node is elected, the new control node will have all the information it needs to continue uninterrupted operation of the established authenticated users and their associated authorizations. Idle and absolute timeouts for user authentications are preserved when a control node change occurs.

Accounting is implemented as a distributed feature in a cluster. Accounting is done on a per-flow basis, so the cluster node owning a flow will send accounting start and stop messages to the AAA server when accounting is configured for a flow.

Connection Settings

Connection limits are enforced cluster-wide (see the **set connection conn-max**, **set connection embryonic-conn-max**, **set connection per-client-embryonic-max**, and **set connection per-client-max** commands). Each node has an estimate of the cluster-wide counter values based on broadcast messages. Due to efficiency considerations, the configured connection limit across the cluster might not be enforced exactly at the limit number. Each node may overestimate or underestimate the cluster-wide counter value at any given time. However, the information will get updated over time in a load-balanced cluster.

FTP and Clustering

- If FTP data channel and control channel flows are owned by different cluster members, then the data channel owner will periodically send idle timeout updates to the control channel owner and update the idle timeout value. However, if the control flow owner is reloaded, and the control flow is re-hosted, the parent/child flow relationship will not longer be maintained; the control flow idle timeout will not be updated.
- If you use AAA for FTP access, then the control channel flow is centralized on the control node.

ICMP Inspection

The flow of ICMP and ICMP error packets through the cluster varies depending on whether ICMP/ICMP error inspection is enabled. Without ICMP inspection, ICMP is a one-direction flow, and there is no director flow support. With ICMP inspection, the ICMP flow becomes two-directional and is backed up by a director/backup flow. One difference for an inspected ICMP flow is in the director handling of a forwarded packet: the director will forward the ICMP echo reply packet to the flow owner instead of returning the packet to the forwarder.

Multicast Routing and Clustering

The control unit handles all multicast routing packets and data packets until fast-path forwarding is established. After the connection is established, each data unit can forward multicast data packets.

NAT and Clustering

NAT can affect the overall throughput of the cluster. Inbound and outbound NAT packets can be sent to different ASAs in the cluster, because the load balancing algorithm relies on IP addresses and ports, and NAT causes inbound and outbound packets to have different IP addresses and/or ports. When a packet arrives at the ASA that is not the NAT owner, it is forwarded over the cluster control link to the owner, causing large amounts of traffic on the cluster control link. Note that the receiving node does not create a forwarding flow to the owner, because the NAT owner may not end up creating a connection for the packet depending on the results of security and policy checks.

If you still want to use NAT in clustering, then consider the following guidelines:

- PAT with Port Block Allocation—See the following guidelines for this feature:

  - Maximum-per-host limit is not a cluster-wide limit, and is enforced on each node individually. Thus, in a 3-node cluster with the maximum-per-host limit configured as 1, if the traffic from a host is load-balanced across all 3 nodes, then it can get allocated 3 blocks with 1 in each node.
  - Port blocks created on the backup node from the backup pools are not accounted for when enforcing the maximum-per-host limit.
  - On-the-fly PAT rule modifications, where the PAT pool is modified with a completely new range of IP addresses, will result in xlate backup creation failures for the xlate backup requests that were still in transit while the new pool became effective. This behavior is not specific to the port block allocation feature, and is a transient PAT pool issue seen only in cluster deployments where the pool is distributed and traffic is load-balanced across the cluster nodes.
  - When operating in a cluster, you cannot simply change the block allocation size. The new size is effective only after you reload each device in the cluster. To avoid having to reload each device, we recommend that you delete all block allocation rules and clear all xlates related to those rules. You can then change the block size and recreate the block allocation rules.

- NAT pool address distribution for dynamic PAT—When you configure a PAT pool, the cluster divides each IP address in the pool into port blocks. By default, each block is 512 ports, but if you configure port block allocation rules, your block setting is used instead. These blocks are distributed evenly among the nodes in the cluster, so that each node has one or more blocks for each IP address in the PAT pool. Thus, you could have as few as one IP address in a PAT pool for a cluster, if that is sufficient for the number of PAT'ed connections you expect. Port blocks cover the 1024-65535 port range, unless you configure the option to include the reserved ports, 1-1023, on the PAT pool NAT rule.
- Reusing a PAT pool in multiple rules—To use the same PAT pool in multiple rules, you must be careful about the interface selection in the rules. You must either use specific interfaces in all rules, or "any" in all rules. You cannot mix specific interfaces and "any" across the rules, or the system might not be able to match return traffic to the right node in the cluster. Using unique PAT pools per rule is the most reliable option.
- No round-robin—Round-robin for a PAT pool is not supported with clustering.
- No extended PAT—Extended PAT is not supported with clustering.
- Dynamic NAT xlates managed by the control node—The control node maintains and replicates the xlate table to data nodes. When a data node receives a connection that requires dynamic NAT, and the xlate is not in the table, it requests the xlate from the control node. The data node owns the connection.
- Stale xlates—The xlate idle time on the connection owner does not get updated. Thus, the idle time might exceed the idle timeout. An idle timer value higher than the configured timeout with a refcnt of 0 is an indication of a stale xlate.
- Per-session PAT feature—Although not exclusive to clustering, the per-session PAT feature improves the scalability of PAT and, for clustering, allows each data node to own PAT connections; by contrast, multi-session PAT connections have to be forwarded to and owned by the control node. By default, all TCP traffic and UDP DNS traffic use a per-session PAT xlate, whereas ICMP and all other UDP traffic uses multi-session. You can configure per-session NAT rules to change these defaults for TCP and UDP, but you cannot configure per-session PAT for ICMP. For example, with an increasing use of Quic protocol over UDP/443 as a best performance alternative compared to HTTPS TLS over TCP/443, you should enable per-session PAT for UDP/443. For traffic that benefits from multi-session PAT, such as H.323, SIP, or Skinny, you can disable per-session PAT for the associated TCP ports (the UDP ports for those H.323 and SIP are already multi-session by default). For more information about per-session PAT, see the firewall configuration guide.

- No static PAT for the following inspections—
  - FTP
  - PPTP
  - RSH
  - SQLNET
  - TFTP
  - XDMCP
  - SIP
- If you have an extremely large number of NAT rules, over ten thousand, you should enable the transactional commit model using the **asp rule-engine transactional-commit nat** command in the device CLI. Otherwise, the node might not be able to join the cluster.

Guidelines for NAT and clustering

Review these guidelines for NAT and clustering:

General guidelines

- NAT pool address distribution for dynamic PAT—When you configure a PAT pool, the cluster divides each IP address in the pool into port blocks. By default, each block is 512 ports, but if you configure port block allocation rules, your block setting is used instead. These blocks are distributed evenly among the nodes in the cluster, so that each node has one or more blocks for each IP address in the PAT pool. Thus, you could have as few as one IP address in a PAT pool for a cluster, if that is sufficient for the number of PAT'ed connections you expect. Port blocks cover the 1024-65535 port range, unless you configure the option to include the reserved ports, 1-1023, on the PAT pool NAT rule.
- Reusing a PAT pool in multiple rules—To use the same PAT pool in multiple rules, you must be careful about the interface selection in the rules. You must either use specific interfaces in all rules, or "any" in all rules. You cannot mix specific interfaces and "any" across the rules, or the system might not be able to match return traffic to the right node in the cluster. Using unique PAT pools per rule is the most reliable option.
- Dynamic NAT xlates managed by the control node—The control node maintains and replicates the xlate table to data nodes. When a data node receives a connection that requires dynamic NAT, and the xlate is not in the table, it requests the xlate from the control node. The data node owns the connection.
- If you have an extremely large number of NAT rules, over ten thousand, you should enable the transactional commit model using the **asp rule-engine transactional-commit nat** command in the device CLI. Otherwise, the node might not be able to join the cluster.

Limitations for NAT and clustering

Review these limitations for NAT and clustering:

General limitations

- No static PAT for these inspections:

- FTP
- PPTP
- RSH
- SQLNET
- TFTP
- XDMCP
- SIP

- Round-robin for a PAT pool is not supported with clustering.
- Extended PAT is not supported with clustering.
- Stale xlates—The xlate idle time on the connection owner does not get updated. Thus, the idle time might exceed the idle timeout. An idle timer value higher than the configured timeout with a refcnt of 0 is an indication of a stale xlate.

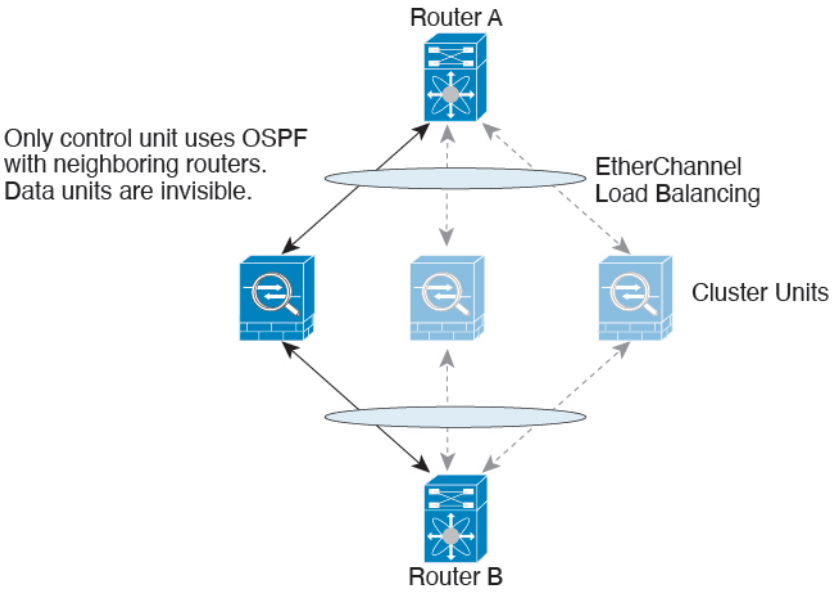
PAT with port block allocation limitations

- Maximum-per-host limit is not a cluster-wide limit, and is enforced on each node individually. Thus, in a 3-node cluster with the maximum-per-host limit configured as 1, if the traffic from a host is load-balanced across all 3 nodes, then it can get allocated 3 blocks with 1 in each node.
- Port blocks created on the backup node from the backup pools are not accounted for when enforcing the maximum-per-host limit.
- On-the-fly PAT rule modifications, where the PAT pool is modified with a completely new range of IP addresses, will result in xlate backup creation failures for the xlate backup requests that were still in transit while the new pool became effective. This behavior is not specific to the port block allocation feature, and is a transient PAT pool issue seen only in cluster deployments where the pool is distributed and traffic is load-balanced across the cluster nodes.
- When operating in a cluster, you cannot simply change the block allocation size. The new size is effective only after you reload each device in the cluster. To avoid having to reload each device, we recommend that you delete all block allocation rules and clear all xlates related to those rules. You can then change the block size and recreate the block allocation rules.

Dynamic Routing and Clustering

The routing process only runs on the control unit, and routes are learned through the control unit and replicated to secondaries. If a routing packet arrives at a data unit, it is redirected to the control unit.

Figure 48: Dynamic Routing



After the data units learn the routes from the control unit, each unit makes forwarding decisions independently.

The OSPF LSA database is not synchronized from the control unit to data units. If there is a control unit switchover, the neighboring router will detect a restart; the switchover is not transparent. The OSPF process picks an IP address as its router ID. Although not required, you can assign a static router ID to ensure a consistent router ID is used across the cluster. See the OSPF Non-Stop Forwarding feature to address the interruption.

SCTP and Clustering

An SCTP association can be created on any node (due to load balancing); its multi-homing connections must reside on the same node.

SIP Inspection and Clustering

A control flow can be created on any node (due to load balancing); its child data flows must reside on the same node.

TLS Proxy configuration is not supported.

SNMP and Clustering

An SNMP agent polls each individual ASA by its Local IP address. You cannot poll consolidated data for the cluster.

You should always use the Local address, and not the Main cluster IP address for SNMP polling. If the SNMP agent polls the Main cluster IP address, if a new control node is elected, the poll to the new control node will fail.

When using SNMPv3 with clustering, if you add a new cluster node after the initial cluster formation, then SNMPv3 users are not replicated to the new node.You must re-add them on the control node to force the users to replicate to the new node, or directly on the data node.

STUN and Clustering

STUN inspection is supported in failover and cluster modes, as pinholes are replicated. However, the transaction ID is not replicated among nodes. In the case where a node fails after receiving a STUN Request and another node received the STUN Response, the STUN Response will be dropped.

Syslog and NetFlow and Clustering

- Syslog—Each node in the cluster generates its own syslog messages. You can configure logging so that each node uses either the same or a different device ID in the syslog message header field. For example, the hostname configuration is replicated and shared by all nodes in the cluster. If you configure logging to use the hostname as the device ID, syslog messages generated by all nodes look as if they come from a single node. If you configure logging to use the local-node name that is assigned in the cluster bootstrap configuration as the device ID, syslog messages look as if they come from different nodes.
- NetFlow—Each node in the cluster generates its own NetFlow stream. The NetFlow collector can only treat each ASA as a separate NetFlow exporter.

Cisco TrustSec and Clustering

Only the control node learns security group tag (SGT) information. The control node then populates the SGT to data nodes, and data nodes can make a match decision for SGT based on the security policy.

VPN and Clustering on the Secure Firewall eXtensible Operating System (FXOS) Chassis

An ASA FXOS Cluster supports one of two mutually exclusive modes for S2S VPN, centralized or distributed:

- Centralized VPN Mode. The default mode. In centralized mode, VPN connections are established with the control unit of the cluster only.

  VPN functionality is limited to the control unit and does not take advantage of the cluster high availability capabilities. If the control unit fails, all existing VPN connections are lost, and VPN connected users see a disruption in service. When a new control unit is elected, you must reestablish the VPN connections.

  When you connect a VPN tunnel to a Spanned interface address, connections are automatically forwarded to the control unit. VPN-related keys and certificates are replicated to all units.

- Distributed VPN Mode. In this mode, S2S IPsec IKEv2 VPN connections are distributed across members of an ASA cluster providing scalability. Distributing VPN connections across the members of a cluster allows both the capacity and throughput of the cluster to be fully utilized, significantly scaling VPN support beyond Centralized VPN capabilities.



Note Centralized VPN clustering mode supports S2S IKEv1 and S2S IKEv2.

Distributed VPN clustering mode supports S2S IKEv2 only.

Distributed VPN clustering mode is supported on the Firepower 9300 only.

Remote access VPN is not supported in centralized or distributed VPN clustering mode.

Performance Scaling Factor

When you combine multiple units into a cluster, you can expect the total cluster performance to be approximately 80% of the maximum combined throughput.

For example, for TCP throughput, the Firepower 9300 with 3 SM-40 modules can handle approximately 135 Gbps of real world firewall traffic when running alone. For 2 chassis, the maximum combined throughput will be approximately 80% of 270 Gbps (2 chassis x 135 Gbps): 216 Gbps.

Control Unit Election

Members of the cluster communicate over the cluster control link to elect a control unit as follows:

1. When you deploy the cluster, each unit broadcasts an election request every 3 seconds.
2. Any other units with a higher priority respond to the election request; the priority is set when you deploy the cluster and is not configurable.
3. If after 45 seconds, a unit does not receive a response from another unit with a higher priority, then it becomes the control unit.



Note If multiple units tie for the highest priority, the cluster unit name and then the serial number is used to determine the control unit.

4. If a unit later joins the cluster with a higher priority, it does not automatically become the control unit; the existing control unit always remains as the control unit unless it stops responding, at which point a new control unit is elected.
5. In a "split brain" scenario when there are temporarily multiple control units, then the unit with highest priority retains the role while the other units return to data unit roles.



Note You can manually force a unit to become the control unit. For centralized features, if you force a control unit change, then all connections are dropped, and you have to re-establish the connections on the new control unit.

High Availability Within the Cluster

Clustering provides high availability by monitoring chassis, unit, and interface health and by replicating connection states between units.

Chassis-Application Monitoring

Chassis-application health monitoring is always enabled. The Firepower 4100/9300 chassis supervisor checks the ASA application periodically (every second). If the ASA is up and cannot communicate with the Firepower 4100/9300 chassis supervisor for 3 seconds, the ASA generates a syslog message and leaves the cluster.

If the Firepower 4100/9300 chassis supervisor cannot communicate with the application after 45 seconds, it reloads the ASA. If the ASA cannot communicate with the supervisor, it removes itself from the cluster.

Unit Health Monitoring

Each unit periodically sends a broadcast keepaliveheartbeat packet over the cluster control link. If the control node does not receive any keepaliveheartbeat packets or other packets from a data node within the configurable timeout period, then the control node removes the data node from the cluster. If the data nodes do not receive packets from the control node, then a new control node is elected from the remaining node.

If nodes cannot reach each other over the cluster control link because of a network failure and not because a node has actually failed, then the cluster may go into a "split brain" scenario where isolated data nodes will elect their own control nodes. For example, if a router fails between two cluster locations, then the original control node at location 1 will remove the location 2 data nodes from the cluster. Meanwhile, the nodes at location 2 will elect their own control node and form their own cluster. Note that asymmetric traffic may fail in this scenario. After the cluster control link is restored, then the control node that has the higher priority will keep the control node's role. See Control Unit Election, on page 509 for more information.

Interface Monitoring

Each node monitors the link status of all hardware interfaces in use, and reports status changes to the control node. For clustering on multiple chassis, Spanned EtherChannels use the cluster Link Aggregation Control Protocol (cLACP). Each chassis monitors the link status and the cLACP protocol messages to determine if the port is still active in the EtherChannel, and informs the ASA application if the interface is down. When you enable health monitoring, all physical interfaces are monitored by default (including the main EtherChannel for EtherChannel interfaces). Only named interfaces that are in an Up state can be monitored. For example, all member ports of an EtherChannel must fail before a *named* EtherChannel is removed from the cluster (depending on your minimum port bundling setting). You can optionally disable monitoring per interface.

If a monitored interface fails on a particular node, but it is active on other nodes, then the node is removed from the cluster. The amount of time before the ASA removes a node from the cluster depends on whether the node is an established member or is joining the cluster. The ASA does not monitor interfaces for the first 90 seconds that a node joins the cluster. Interface status changes during this time will not cause the ASA to be removed from the cluster. For an established member, the node is removed after 500 ms.

For clustering on multiple chassis, if you add or delete an EtherChannel from the cluster, interface health-monitoring is suspended for 95 seconds to ensure that you have time to make the changes on each chassis.

Decorator Application Monitoring

When you install a decorator application on an interface, such as the Radware DefensePro application, then both the ASA and the decorator application must be operational to remain in the cluster. The unit does not join the cluster until both applications are operational. Once in the cluster, the unit monitors the decorator application health every 3 seconds. If the decorator application is down, the unit is removed from the cluster.

Status After Failure

When a node in the cluster fails, the connections hosted by that node are seamlessly transferred to other nodes; state information for traffic flows is shared over the control node's cluster control link.

If the control node fails, then another member of the cluster with the highest priority (lowest number) becomes the control node.

The ASA automatically tries to rejoin the cluster, depending on the failure event.



Note When the ASA becomes inactive and fails to automatically rejoin the cluster, all data interfaces are shut down; only the management-only interface can send and receive traffic. The management interface remains up using the IP address the node received from the cluster IP pool. However if you reload, and the node is still inactive in the cluster, the management interface is disabled. You must use the console port for any further configuration.

Rejoining the Cluster

After a cluster member is removed from the cluster, how it can rejoin the cluster depends on why it was removed:

- Failed cluster control link when initially joining—After you resolve the problem with the cluster control link, you must manually rejoin the cluster by re-enabling clustering at the ASA console port by entering **cluster group** *name*, and then **enable**.
- Failed cluster control link after joining the cluster—The ASA automatically tries to rejoin every 5 minutes, indefinitely. This behavior is configurable.
- Failed data interface—The ASA automatically tries to rejoin at 5 minutes, then at 10 minutes, and finally at 20 minutes. If the join is not successful after 20 minutes, then the ASA disables clustering. After you resolve the problem with the data interface, you have to manually enable clustering at the ASA console port by entering **cluster group** *name*, and then **enable**.This behavior is configurable.
- Failed unit—If the unit was removed from the cluster because of a unit health check failure, then rejoining the cluster depends on the source of the failure. For example, a temporary power failure means the unit will rejoin the cluster when it starts up again as long as the cluster control link is up. The unit attempts to rejoin the cluster every 5 seconds.
- Failed Chassis-Application Communication—When the ASA detects that the chassis-application health has recovered, the ASA tries to rejoin the cluster immediately.
- Failed decorator application—The ASA rejoins the cluster when it senses that the decorator application is back up.
- Internal error—Internal failures include: application sync timeout; inconsistent application statuses; and so on. A unit will attempt to rejoin the cluster automatically at the following intervals: 5 minutes, 10 minutes, and then 20 minutes. This behavior is configurable.

Data Path Connection State Replication

Every connection has one owner and at least one backup owner in the cluster. The backup owner does not take over the connection in the event of a failure; instead, it stores TCP/UDP state information, so that the connection can be seamlessly transferred to a new owner in case of a failure. The backup owner is usually also the director.

Some traffic requires state information above the TCP or UDP layer. See the following table for clustering support or lack of support for this kind of traffic.

Table 19: Features Replicated Across the Cluster

Traffic	State Support	Notes
Up time	Yes	Keeps track of the system up time.
ARP Table	Yes	—
MAC address table	Yes	—
User Identity	Yes	Includes AAA rules (uauth).
IPv6 Neighbor database	Yes	—
Dynamic routing	Yes	—
SNMP Engine ID	**No**	—
Distributed VPN (Site-to-Site) for Firepower 4100/9300	Yes	Backup session becomes the active session, then a new backup session is created.

How the Cluster Manages Connections

Connections can be load-balanced to multiple nodes of the cluster. Connection roles determine how connections are handled in both normal operation and in a high availability situation.

Connection Roles

See the following roles defined for each connection:

- Owner—Usually, the node that initially receives the connection. The owner maintains the TCP state and processes packets. A connection has only one owner. If the original owner fails, then when new nodes receive packets from the connection, the director chooses a new owner from those nodes.
- Backup owner—The node that stores TCP/UDP state information received from the owner, so that the connection can be seamlessly transferred to a new owner in case of a failure. The backup owner does not take over the connection in the event of a failure. If the owner becomes unavailable, then the first node to receive packets from the connection (based on load balancing) contacts the backup owner for the relevant state information so it can become the new owner.

  As long as the director (see below) is not the same node as the owner, then the director is also the backup owner. If the owner chooses itself as the director, then a separate backup owner is chosen.

  For clustering on the Firepower 9300, which can include up to 3 cluster nodes in one chassis, if the backup owner is on the same chassis as the owner, then an additional backup owner will be chosen from another chassis to protect flows from a chassis failure.

  If you enable director localization for inter-site clustering, then there are two backup owner roles: the local backup and the global backup. The owner always chooses a local backup at the same site as itself (based on site ID). The global backup can be at any site, and might even be the same node as the local backup. The owner sends connection state information to both backups.

  If you enable site redundancy, and the backup owner is at the same site as the owner, then an additional backup owner will be chosen from another site to protect flows from a site failure. Chassis backup and site backup are independent, so in some cases a flow will have both a chassis backup and a site backup.

- Director—The node that handles owner lookup requests from forwarders. When the owner receives a new connection, it chooses a director based on a hash of the source/destination IP address and ports (see below for ICMP hash details), and sends a message to the director to register the new connection. If packets arrive at any node other than the owner, the node queries the director about which node is the owner so it can forward the packets. A connection has only one director. If a director fails, the owner chooses a new director.

  As long as the director is not the same node as the owner, then the director is also the backup owner (see above). If the owner chooses itself as the director, then a separate backup owner is chosen.

  If you enable director localization for inter-site clustering, the cluster uses two director roles: the local director and the global director. For a new-flow packet, the receiving member selects the local director from members at the receiving member's site. The global director can be at any site and might be the same member as the local director. If the original owner fails and localization applies, the local director chooses a new connection owner at the local director's site.

  ICMP/ICMPv6 hash details:

  - For Echo packets, the source port is the ICMP identifier, and the destination port is 0.
  - For Reply packets, the source port is 0, and the destination port is the ICMP identifier.
  - For other packets, both source and destination ports are 0.

- Forwarder—A node that forwards packets to the owner. If a forwarder receives a packet for a connection it does not own, it queries the director for the owner, and then establishes a flow to the owner for any other packets it receives for this connection. The director can also be a forwarder. If you enable director localization, the forwarder queries the local director when localization applies. The forwarder queries the global director if the local director does not know the owner, if the connection is owned at another site, or if localization is bypassed. A cross-site director query is expected behavior in these cases. Note that if a forwarder receives the SYN-ACK packet, it can derive the owner directly from a SYN cookie in the packet, so it does not need to query the director. (If you disable TCP sequence randomization, the SYN cookie is not used; a query to the director is required.) For short-lived flows such as DNS and ICMP, instead of querying, the forwarder immediately sends the packet to the director, which then sends them to the owner. A connection can have multiple forwarders; the most efficient throughput is achieved by a good load-balancing method where there are no forwarders and all packets of a connection are received by the owner.



Note We do not recommend disabling TCP sequence randomization when using clustering. There is a small chance that some TCP sessions won't be established, because the SYN/ACK packet might be dropped.

- Fragment Owner—For fragmented packets, cluster nodes that receive a fragment determine a fragment owner using a hash of the fragment source IP address, destination IP address, and the packet ID. All fragments are then forwarded to the fragment owner over the cluster control link. Fragments may be load-balanced to different cluster nodes, because only the first fragment includes the 5-tuple used in the switch load balance hash. Other fragments do not contain the source and destination ports and may be load-balanced to other cluster nodes. The fragment owner temporarily reassembles the packet so it can determine the director based on a hash of the source/destination IP address and ports. If it is a new connection, the fragment owner will register to be the connection owner. If it is an existing connection, the fragment owner forwards all fragments to the provided connection owner over the cluster control link. The connection owner will then reassemble all fragments.

Port Address Translation Connections

When a connection uses Port Address Translation (PAT), then the PAT type (per-session or multi-session) influences which member of the cluster becomes the owner of a new connection:

- Per-session PAT—The owner is the node that receives the initial packet in the connection.

  By default, TCP and DNS UDP traffic use per-session PAT.

- Multi-session PAT—The owner is always the control node. If a multi-session PAT connection is initially received by a data node, then the data node forwards the connection to the control node.

  By default, UDP (except for DNS UDP) and ICMP traffic use multi-session PAT, so these connections are always owned by the control node.

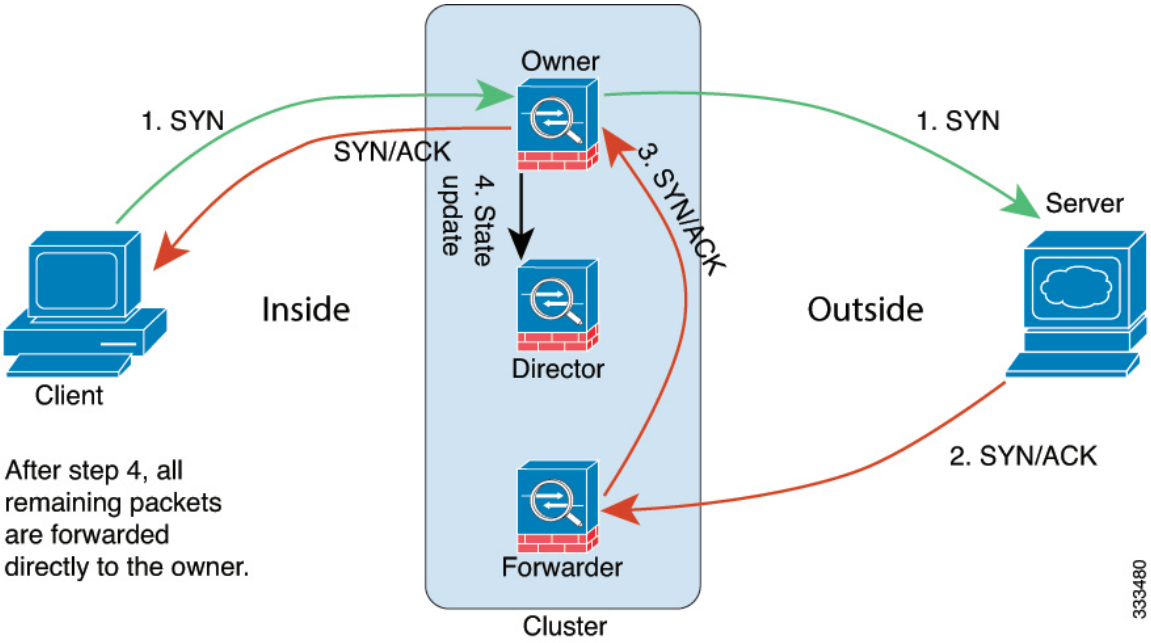
You can change the per-session PAT defaults for TCP and UDP so connections for these protocols are handled per-session or multi-session depending on the configuration. For ICMP, you cannot change from the default multi-session PAT. For more information about per-session PAT, see the firewall configuration guide.

New Connection Ownership

When a new connection is directed to a node of the cluster via load balancing, that node owns both directions of the connection. If any connection packets arrive at a different node, they are forwarded to the owner node over the cluster control link. If a reverse flow arrives at a different node, it is redirected back to the original node.

Sample Data Flow for TCP

The following example shows the establishment of a new connection.



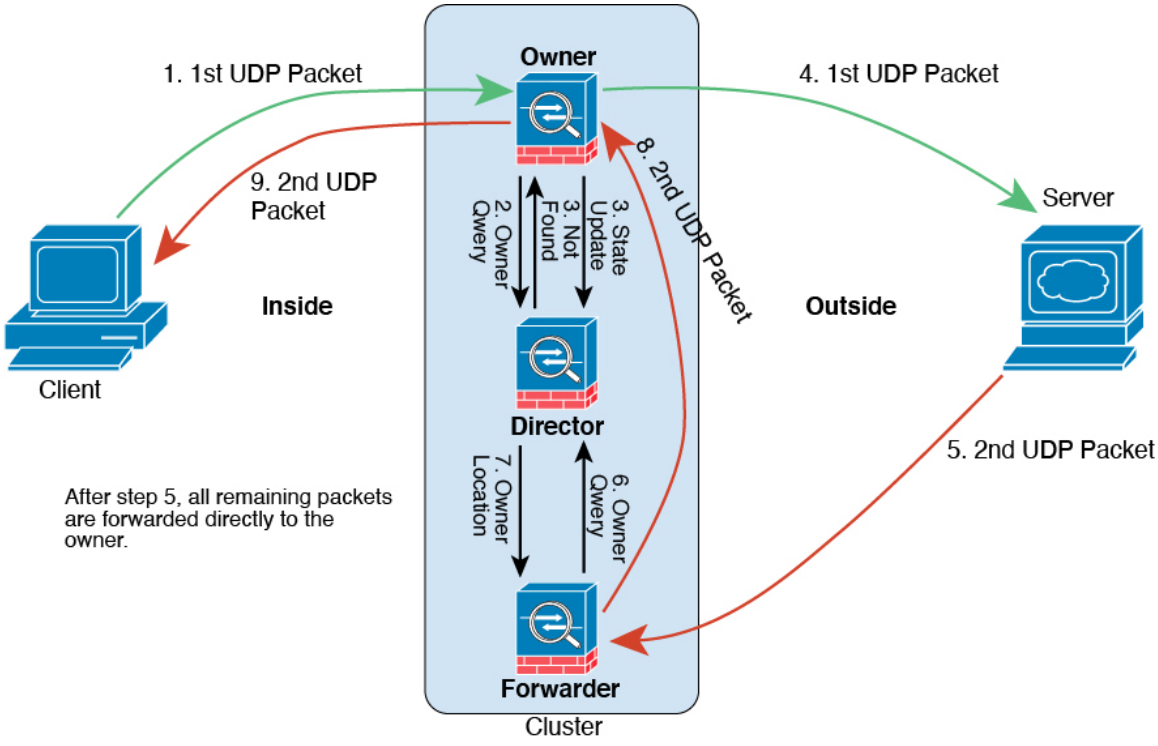
1. The SYN packet originates from the client and is delivered to one ASA (based on the load balancing method), which becomes the owner. The owner creates a flow, encodes owner information into a SYN cookie, and forwards the packet to the server.

2. The SYN-ACK packet originates from the server and is delivered to a different ASA (based on the load balancing method). This ASA is the forwarder.
3. Because the forwarder does not own the connection, it decodes owner information from the SYN cookie, creates a forwarding flow to the owner, and forwards the SYN-ACK to the owner.
4. The owner sends a state update to the director, and forwards the SYN-ACK to the client.
5. The director receives the state update from the owner, creates a flow to the owner, and records the TCP state information as well as the owner. The director acts as the backup owner for the connection.
6. Any subsequent packets delivered to the forwarder will be forwarded to the owner.
7. If packets are delivered to any additional nodes, it will query the director for the owner and establish a flow.
8. Any state change for the flow results in a state update from the owner to the director.

Sample Data Flow for ICMP and UDP

The following example shows the establishment of a new connection.

1. ***Figure 49: ICMP and UDP Data Flow***



The first UDP packet originates from the client and is delivered to one ASA (based on the load balancing method).

2. The node that received the first packet queries the director node that is chosen based on a hash of the source/destination IP address and ports.

3. The director finds no existing flow, creates a director flow and forwards the packet back to the previous node. In other words, the director has elected an owner for this flow.
4. The owner creates the flow, sends a state update to the director, and forwards the packet to the server.
5. The second UDP packet originates from the server and is delivered to the forwarder.
6. The forwarder queries the director for ownership information. For short-lived flows such as DNS, instead of querying, the forwarder immediately sends the packet to the director, which then sends it to the owner.
7. The director replies to the forwarder with ownership information.
8. The forwarder creates a forwarding flow to record owner information and forwards the packet to the owner.
9. The owner forwards the packet to the client.

Rebalancing New TCP Connections Across the Cluster

If the load balancing capabilities of the upstream or downstream routers result in unbalanced flow distribution, you can configure new connection rebalancing so nodes with higher new connections per second will redirect new TCP flows to other nodes. No existing flows will be moved to other nodes.

Because this command only rebalances based on connections per second, the total number of established connections on each node is not considered, and the total number of connections may not be equal.

Once a connection is offloaded to a different node, it becomes an asymmetric connection.

Do not configure connection rebalancing for inter-site topologies; you do not want new connections rebalanced to cluster members at a different site.

History for ASA Clustering on the Firepower 4100/9300

Feature Name	Version	Feature Information
Improved PAT port block allocation for clustering on the Firepower 4100/9300	9.16(1)	The improved PAT port block allocation ensures that the control unit keeps ports in reserve for joining nodes, and proactively reclaims unused ports. To best optimize the allocation, you can set the maximum nodes you plan to have in the cluster using the **cluster-member-limit** command. The control unit can then allocate port blocks to the planned number of nodes, and it will not have to reserve ports for extra nodes you don't plan to use. The default is 16 nodes. You can also monitor syslog 747046 to ensure that there are enough ports available for a new node. New/Modified commands: **cluster-member-limit**, **show nat pool cluster** [**summary**], **show nat pool ip detail**
show cluster history command improvements	9.16(1)	We have added additional outputs for the **show cluster history** command. New/Modified commands: **show cluster history brief**, **show cluster history latest**, **show cluster history reverse**, **show cluster history time**
MTU ping test from control node on node join	9.16(1)	When a node joins the cluster, the control node checks MTU compatibility by sending a ping to the data node with a packet size two-times the cluster control link MTU. If the ping fails, a notification is generated so you can fix the MTU mismatch on connecting switches and try again. Added/modified commands: **show cluster history**.

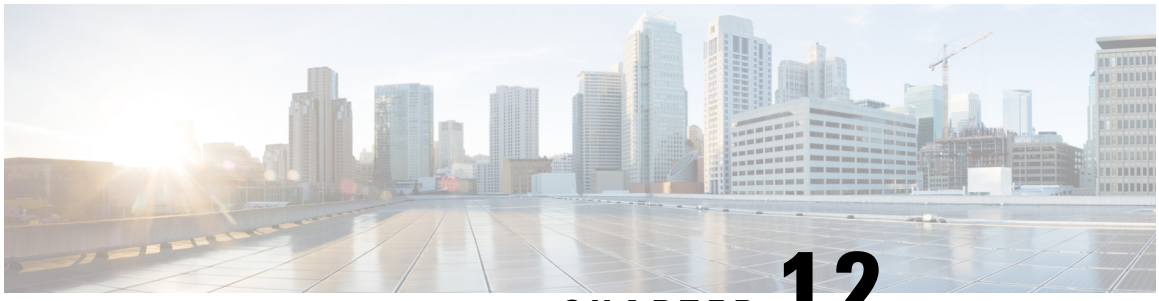
Feature Name	Version	Feature Information
Configuration sync to data units in parallel	9.14(1)	The control unit now syncs configuration changes with data units in parallel by default. Formerly, synching occurred sequentially. New/Modified commands: **config-replicate-parallel**
Messages for cluster join failure or eviction added to **show cluster history**	9.14(1)	New messages were added to the **show cluster history** command for when a cluster unit either fails to join the cluster or leaves the cluster. New/Modified commands: **show cluster history**
Initiator and responder information for Dead Connection Detection (DCD), and DCD support in a cluster.	9.13(1)	If you enable Dead Connection Detection (DCD), you can use the **show conn detail** command to get information about the initiator and responder. Dead Connection Detection allows you to maintain an inactive connection, and the **show conn** output tells you how often the endpoints have been probed. In addition, DCD is now supported in a cluster. New/Modified commands: **show conn** (output only).
Monitor the traffic load for a cluster	9.13(1)	You can now monitor the traffic load for cluster members, including total connection count, CPU and memory usage, and buffer drops. If the load is too high, you can choose to manually disable clustering on the unit if the remaining units can handle the load, or adjust the load balancing on the external switch. This feature is enabled by default. New/Modified commands: **debug cluster load-monitor**, **load-monitor**, **show cluster info load-monitor**
Accelerated cluster joining	9.13(1)	When a data unit has the same configuration as the control unit, it will skip syncing the configuration and will join faster. This feature is enabled by default. This feature is configured on each unit, and is not replicated from the control unit to the data unit. **Note** Some configuration commands are not compatible with accelerated cluster joining; if these commands are present on the unit, even if accelerated cluster joining is enabled, configuration syncing will always occur. You must remove the incompatible configuration for accelerated cluster joining to work. Use the **show cluster info unit-join-acceleration incompatible-config** to view incompatible configuration. New/Modified commands: **unit join-acceleration**, **show cluster info unit-join-acceleration incompatible-config**
Per-site gratuitous ARP for clustering	9.12(1)	The ASA now generates gratuitous ARP (GARP) packets to keep the switching infrastructure up to date: the highest priority member at each site periodically generates GARP traffic for the global MAC/IP addresses. When using per-site MAC and IP addresses, packets sourced from the cluster use a site-specific MAC address and IP address, while packets received by the cluster use a global MAC address and IP address. If traffic is not generated from the global MAC address periodically, you could experience a MAC address timeout on your switches for the global MAC address. After a timeout, traffic destined for the global MAC address will be flooded across the entire switching infrastructure, which can cause performance and security concerns. GARP is enabled by default when you set the site ID for each unit and the site MAC address for each Spanned EtherChannel. New/Modified commands: **site-periodic-garp interval**

Feature Name	Version	Feature Information
Parallel cluster joining of units per Firepower 9300 chassis	9.10(1)	For the Firepower 9300, this feature ensures that the security modules in a chassis join the cluster simultaneously, so that traffic is evenly distributed between the modules. If a module joins very much in advance of other modules, it can receive more traffic than desired, because the other modules cannot yet share the load. New/Modified commands: **unit parallel-join**
Cluster control link customizable IP Address for the Firepower 4100/9300	9.10(1)	By default, the cluster control link uses the 127.2.0.0/16 network. You can now set the network when you deploy the cluster in FXOS. The chassis auto-generates the cluster control link interface IP address for each unit based on the chassis ID and slot ID: 127.2.*chassis_id*.*slot_id*. However, some networking deployments do not allow 127.2.0.0/16 traffic to pass. Therefore, you can now set a custom /16 subnet for the cluster control link in FXOS except for loopback (127.0.0.0/8) and multicast (224.0.0.0/4) addresses. New/Modified FXOS commands: **set cluster-control-link network**
Cluster interface debounce time now applies to interfaces changing from a down state to an up state	9.10(1)	When an interface status update occurs, the ASA waits the number of milliseconds specified in the **health-check monitor-interface debounce-time** command or the ASDM **Configuration** > **Device Management** > **High Availability and Scalability** > **ASA Cluster** screen before marking the interface as failed and the unit is removed from the cluster. This feature now applies to interfaces changing from a down state to an up state. For example, in the case of an EtherChannel that transitions from a down state to an up state (for example, the switch reloaded, or the switch enabled an EtherChannel), a longer debounce time can prevent the interface from appearing to be failed on a cluster unit just because another cluster unit was faster at bundling the ports. We did not modify any commands.
Automatically rejoin the cluster after an internal failure	9.9(2)	Formerly, many error conditions caused a cluster unit to be removed from the cluster, and you were required to manually rejoin the cluster after resolving the issue. Now, a unit will attempt to rejoin the cluster automatically at the following intervals by default: 5 minutes, 10 minutes, and then 20 minutes. These values are configurable. Internal failures include: application sync timeout; inconsistent application statuses; and so on. New or Modified commands: **health-check system auto-rejoin, show cluster info auto-join**
Show transport related statistics for cluster reliable transport protocol messages	9.9(2)	You can now view per-unit cluster reliable transport buffer usage so you can identify packet drop issues when the buffer is full in the control plane. New or modified command: **show cluster info transport cp detail**
cluster remove unit command behavior matches **no enable** behavior	9.9(1)	The **cluster remove unit** command now removes a unit from the cluster until you manually reenable clustering or reload, similar to the **no enable** command. Previously, if you redeployed the bootstrap configuration from FXOS, clustering would be reenabled. Now, the disabled status persists even in the case of a bootstrap configuration redeployment. Reloading the ASA, however, will reenable clustering. New/Modified command: **cluster remove unit**
Improved chassis health check failure detection for the chassis	9.9(1)	You can now configure a lower holdtime for the chassis health check: 100 ms. The previous minimum was 300 ms. Note that the minimum combined time (*interval* x *retry-count*) cannot be less than 600 ms. New or modified command: **app-agent heartbeat interval**

Feature Name	Version	Feature Information
Inter-site redundancy for clustering	9.9(1)	Inter-site redundancy ensures that a backup owner for a traffic flow will always be at the other site from the owner. This feature guards against site failure. New or modified command: **site-redundancy, show asp cluster counter change, show asp table cluster chash-table, show conn flag**
Distributed Site-to-Site VPN with clustering on the Firepower 9300	9.9(1)	An ASA cluster on the Firepower 9300 supports Site-to-Site VPN in distributed mode. Distributed mode provides the ability to have many Site-to-Site IPsec IKEv2 VPN connections distributed across members of an ASA cluster, not just on the control unit (as in centralized mode). This significantly scales VPN support beyond Centralized VPN capabilities and provides high availability. Distributed S2S VPN runs on a cluster of up to two chassis, each containing up to three modules (six total cluster members), each module supporting up to 6K active sessions (12K total), for a maximum of approximately 36K active sessions (72K total). New or modified commands: **cluster redistribute vpn-sessiondb**, **show cluster vpn-sessiondb**, **vpn mode** , **show cluster resource usage**, **show vpn-sessiondb** , **show connection detail**, **show crypto ikev2**
Improved cluster unit health-check failure detection	9.8(1)	You can now configure a lower holdtime for the unit health check: .3 seconds minimum. The previous minimum was .8 seconds. This feature changes the unit health check messaging scheme to *heartbeats* in the data plane from *keepalives* in the control plane. Using heartbeats improves the reliability and the responsiveness of clustering by not being susceptible to control plane CPU hogging and scheduling delays. Note that configuring a lower holdtime increases cluster control link messaging activity. We suggest that you analyze your network before you configure a low holdtime; for example, make sure a ping from one unit to another over the cluster control link returns within the *holdtime*/3, because there will be three heartbeat messages during one holdtime interval. If you downgrade your ASA software after setting the hold time to .3 - .7, this setting will revert to the default of 3 seconds because the new setting is unsupported. We modified the following commands: **health-check holdtime, show asp drop cluster counter, show cluster info health details**
Configurable debounce time to mark an interface as failed for the Firepower 4100/9300 chassis	9.8(1)	You can now configure the debounce time before the ASA considers an interface to be failed, and the unit is removed from the cluster. This feature allows for faster detection of interface failures. Note that configuring a lower debounce time increases the chances of false-positives. When an interface status update occurs, the ASA waits the number of milliseconds specified before marking the interface as failed and the unit is removed from the cluster. The default debounce time is 500 ms, with a range of 300 ms to 9 seconds. New or modified command: **health-check monitor-interface debounce-time**
Inter-site clustering improvement for the ASA on the Firepower 4100/9300 chassis	9.7(1)	You can now configure the site ID for each Firepower 4100/9300 chassis when you deploy the ASA cluster. Previously, you had to configure the site ID within the ASA application; this new feature eases initial deployment. Note that you can no longer set the site ID within the ASA configuration. Also, for best compatibility with inter-site clustering, we recommend that you upgrade to ASA 9.7(1) and FXOS 2.1.1, which includes several improvements to stability and performance. We modified the following command: **site-id**

Feature Name	Version	Feature Information
Director localization: inter-site clustering improvement for data centers	9.7(1)	To improve performance and keep traffic within a site for inter-site clustering for data centers, you can enable director localization. New connections are typically load-balanced and owned by cluster members within a given site. However, the ASA assigns the director role to a member at *any* site. Director localization enables additional director roles: a local director at the same site as the owner, and a global director that can be at any site. Keeping the owner and director at the same site improves performance. Also, if the original owner fails, the local director chooses a new connection owner at the same site. The global director is used if a cluster member receives packets for a connection that is owned on a different site. We introduced or modified the following commands: **director-localization, show asp table cluster chash, show conn, show conn detail**
Support for 16 chassis for the Firepower 4100 series	9.6(2)	You can now add up to 16 chassis to the cluster for the Firepower 4100 series. We did not modify any commands.
Support for the Firepower 4100 series	9.6(1)	With FXOS 1.1.4, the ASA supports inter-chassis clustering on the Firepower 4100 series for up to 6 chassis. We did not modify any commands.
Support for site-specific IP addresses in Routed, Spanned EtherChannel mode	9.6(1)	For inter-site clustering in routed mode with Spanned EtherChannels, you can now configure site-specific IP addresess in addition to site-specific MAC addresses. The addition of site IP addresses allows you to use ARP inspection on the Overlay Transport Virtualization (OTV) devices to prevent ARP responses from the global MAC address from traveling over the Data Center Interconnect (DCI), which can cause routing problems. ARP inspection is required for some switches that cannot use VACLs to filter MAC addresses. We modified the following commands: **mac-address, show interface**
Inter-chassis clustering for 16 modules, and inter-site clustering for the Firepower 9300 ASA application	9.5(2.1)	With FXOS 1.1.3, you can now enable inter-chassis, and by extension inter-site clustering. You can include up to 16 modules. For example, you can use 1 module in 16 chassis, or 2 modules in 8 chassis, or any combination that provides a maximum of 16 modules. We did not modify any commands.
Site-specific MAC addresses for inter-site clustering support for Spanned EtherChannel in Routed firewall mode	9.5(2)	You can now use inter-site clustering for Spanned EtherChannels in routed mode. To avoid MAC address flapping, configure a site ID for each cluster member so that a site-specific MAC address for each interface can be shared among a site's units. We introduced or modified the following commands: **site-id, mac-address site-id, show cluster info, show interface**
ASA cluster customization of the auto-rejoin behavior when an interface or the cluster control link fails	9.5(2)	You can now customize the auto-rejoin behavior when an interface or the cluster control link fails. We introduced the following command: **health-check auto-rejoin**
The ASA cluster supports GTPv1 and GTPv2	9.5(2)	The ASA cluster now supports GTPv1 and GTPv2 inspection. We did not modify any commands.

Feature Name	Version	Feature Information
Cluster replication delay for TCP connections	9.5(2)	This feature helps eliminate the "unnecessary work" related to short-lived flows by delaying the director/backup flow creation. We introduced the following command: **cluster replication delay**
LISP Inspection for Inter-Site Flow Mobility	9.5(2)	Cisco Locator/ID Separation Protocol (LISP) architecture separates the device identity from its location into two different numbering spaces, making server migration transparent to clients. The ASA can inspect LISP traffic for location changes and then use this information for seamless clustering operation; the ASA cluster members inspect LISP traffic passing between the first hop router and the egress tunnel router (ETR) or ingress tunnel router (ITR), and then change the flow owner to be at the new site. We introduced or modified the following commands: **allowed-eid, clear cluster info flow-mobility counters, clear lisp eid, cluster flow-mobility lisp, debug cluster flow-mobility, debug lisp eid-notify-intercept, flow-mobility lisp, inspect lisp, policy-map type inspect lisp, site-id, show asp table classify domain inspect-lisp, show cluster info flow-mobility counters, show conn, show lisp eid, show service-policy, validate-key**
Carrier Grade NAT enhancements now supported in failover and ASA clustering	9.5(2)	For carrier-grade or large-scale PAT, you can allocate a block of ports for each host, rather than have NAT allocate one port translation at a time (see RFC 6888). This feature is now supported in failover and ASA cluster deployments. We modified the following command: **show local-host**
Configurable level for clustering trace entries	9.5(2)	By default, all levels of clustering events are included in the trace buffer, including many low level events. To limit the trace to higher level events, you can set the minimum trace level for the cluster. We introduced the following command: **trace-level**
Intra-chassis ASA Clustering for the Firepower 9300	9.4(1.150)	You can cluster up to 3 security modules within the Firepower 9300 chassis. All modules in the chassis must belong to the cluster. We introduced the following commands: **cluster replication delay, debug service-module, management-only individual, show cluster chassis**



CHAPTER 12

ASA Cluster for the ASAv for the Private Cloud

Clustering lets you group multiple ASAv's together as a single logical device. A cluster provides all the convenience of a single device (management, integration into a network) while achieving the increased throughput and redundancy of multiple devices. You can deploy the ASAv clusters using:

- KVM
- VMware

Note Only routed firewall mode is supported.

Note Some features are not supported when using clustering. See Unsupported Features with Clustering, on page 574.

- About ASAv Clustering, on page 523
- Licenses for ASAv Clustering, on page 530
- Requirements and Prerequisites for ASAv Clustering, on page 530
- Guidelines for ASAv Clustering, on page 530
- Configure the ASAv Clustering Using a Day0 Configuration, on page 531
- Configure ASAv Clustering after Deployment, on page 534
- Customize the Clustering Operation, on page 547
- Manage Cluster Nodes, on page 558
- Monitoring the ASAv Cluster, on page 562
- Examples for ASAv Clustering, on page 573
- Reference for Clustering, on page 574
- History for ASAv Clustering, on page 589

About ASAv Clustering

This section describes the clustering architecture and how it works.

How the Cluster Fits into Your Network

The cluster consists of multiple firewalls acting as a single device. To act as a cluster, the firewalls need the following infrastructure:

- Isolated network for intra-cluster communication, known as the *cluster control link*, using VXLAN interfaces. VXLANs, which act as Layer 2 virtual networks over Layer 3 physical networks, let the ASAv send broadcast/multicast messages over the cluster control link.
- Management access to each firewall for configuration and monitoring. The ASAv deployment includes a Management 0/0 interface that you will use to manage the cluster nodes.

When you place the cluster in your network, the upstream and downstream routers need to be able to load-balance the data coming to and from the cluster using Layer 3 Individual interfaces and one of the following methods:

- Policy-Based Routing—The upstream and downstream routers perform load balancing between nodes using route maps and ACLs.
- Equal-Cost Multi-Path Routing—The upstream and downstream routers perform load balancing between nodes using equal cost static or dynamic routes.



Note Layer 2 Spanned EtherChannels are not supported.

Cluster Nodes

Cluster nodes work together to accomplish the sharing of the security policy and traffic flows. This section describes the nature of each node role.

Bootstrap Configuration

On each device, you configure a minimal bootstrap configuration including the cluster name, cluster control link interface, and other cluster settings. The first node on which you enable clustering typically becomes the *control* node. When you enable clustering on subsequent nodes, they join the cluster as *data* nodes.

Control and data node roles

One member of the cluster is the control node. If multiple cluster nodes come online at the same time, the control node is determined by the priority setting in the bootstrap configuration; the priority is set between 1 and 100, where 1 is the highest priority. All other members are data nodes. Typically, when you first create a cluster, the first node you add becomes the control node simply because it is the only node in the cluster so far.

You must perform all configuration (aside from the bootstrap configuration) on the control node only; the configuration is then replicated to the data nodes. In the case of physical assets, such as interfaces, the configuration of the control node is mirrored on all data nodes. For example, if you configure Ethernet 1/2 as the inside interface and Ethernet 1/1 as the outside interface, then these interfaces are also used on the data nodes as inside and outside interfaces.

Some features do not scale in a cluster, and the control node handles all traffic for those features.

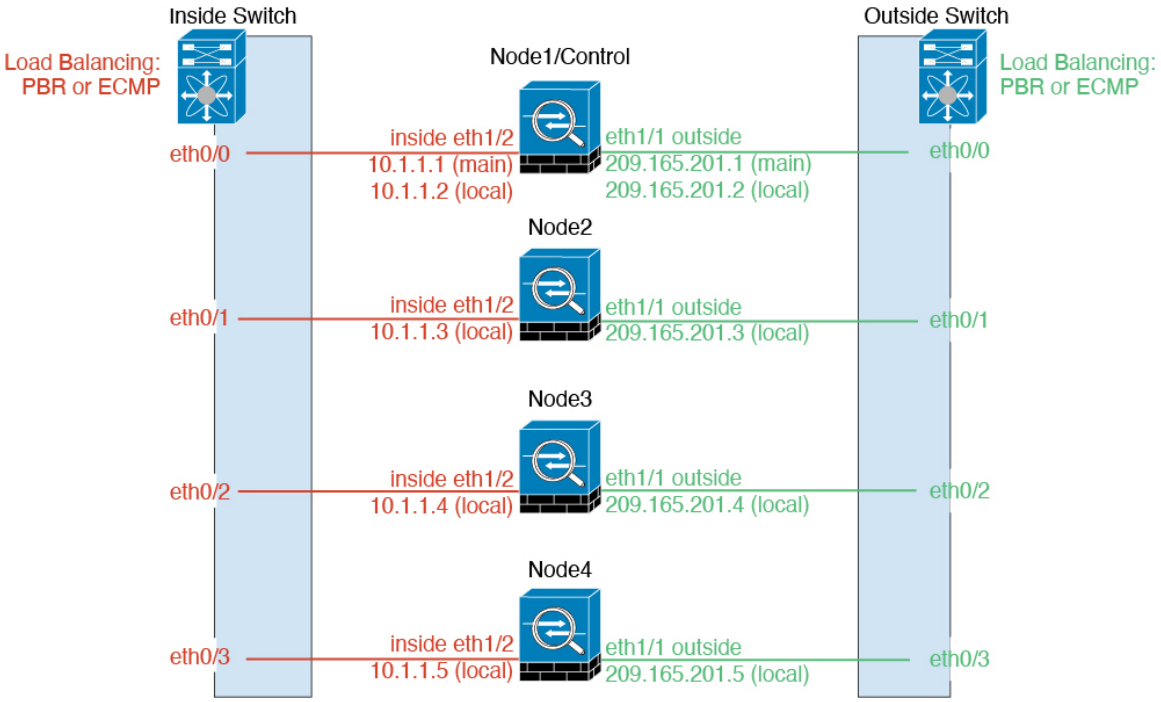
Individual Interfaces

You can configure cluster interfaces as *Individual interfaces*.

Individual interfaces are normal routed interfaces, each with their own *local IP address* used for routing. The *main cluster IP address* for each interface is a fixed address that always belongs to the control node. When the control node changes, the Main cluster IP address moves to the new control node, so management of the cluster continues seamlessly.

Because interface configuration must be configured only on the control node, you configure a pool of IP addresses to be used for a given interface on the cluster nodes, including one for the control node.

Load balancing must be configured separately on the upstream switch.



Note Layer 2 Spanned EtherChannels are not supported.

Policy-based routing

When using Individual interfaces, each ASA interface maintains its own IP address and MAC address. One method of load balancing is Policy-Based Routing (PBR).

We recommend this method if you are already using PBR, and want to take advantage of your existing infrastructure.

PBR makes routing decisions based on a route map and ACL. You must manually divide traffic between all ASAs in a cluster. Because PBR is static, it may not achieve the optimum load balancing result at all times. To achieve the best performance, we recommend that you configure the PBR policy so that forward and return packets of a connection are directed to the same ASA.

For example, if you have a Cisco router, redundancy can be achieved by using Cisco IOS PBR with Object Tracking. Cisco IOS Object Tracking monitors each ASA using ICMP ping. PBR can then enable or disable route maps based on reachability of a particular ASA.

Equal-cost multi-path routing

When using Individual interfaces, each ASA interface maintains its own IP address and MAC address. One method of load balancing is Equal-Cost Multi-Path (ECMP) routing.

We recommend this method if you are already using ECMP, and want to take advantage of your existing infrastructure.

ECMP routing considerations

ECMP routing can forward packets over multiple best paths that tie for top place in the routing metric. Like EtherChannel, a hash of source and destination IP addresses and/or source and destination ports can be used to send a packet to one of the next hops.

If you use static routes for ECMP routing, then the ASA failure can cause problems; the route continues to be used, and traffic to the failed ASA will be lost.

If you use static routes, be sure to use a static route monitoring feature such as Object Tracking. We recommend using dynamic routing protocols to add and remove routes, in which case, you must configure each ASA to participate in dynamic routing.

Cluster Control Link

Each node must dedicate one interface as a VXLAN (VTEP) interface for the cluster control link. For more information about VXLAN, see VXLAN Interfaces, on page 645.

VXLAN Tunnel Endpoint

VXLAN tunnel endpoint (VTEP) devices perform VXLAN encapsulation and decapsulation. Each VTEP has two interface types: one or more virtual interfaces called VXLAN Network Identifier (VNI) interfaces, and a regular interface called the VTEP source interface that tunnels the VNI interfaces between VTEPs. The VTEP source interface is attached to the transport IP network for VTEP-to-VTEP communication.

VTEP Source Interface

The VTEP source interface is a regular ASAv interface with which you plan to associate the VNI interface. You can configure one VTEP source interface to act as the cluster control link. The source interface is reserved for cluster control link use only. Each VTEP source interface has an IP address on the same subnet. This subnet should be isolated from all other traffic, and should include only the cluster control link interfaces.

VNI Interface

A VNI interface is similar to a VLAN interface: it is a virtual interface that keeps network traffic separated on a given physical interface by using tagging. You can only configure one VNI interface. Each VNI interface has an IP address on the same subnet.

Peer VTEPs

Unlike regular VXLAN for data interfaces, which allows a single VTEP peer, The ASAv clustering allows you to configure multiple peers.

Cluster Control Link Traffic Overview

Cluster control link traffic includes both control and data traffic.

Control traffic includes:

- Control node election.
- Configuration replication.
- Health monitoring.

Data traffic includes:

- State replication.
- Connection ownership queries and data packet forwarding.

Cluster Control Link Failure

If the cluster control link line protocol goes down for a unit, then clustering is disabled; data interfaces are shut down. After you fix the cluster control link, you must manually rejoin the cluster by re-enabling clustering.



Note When the ASAv becomes inactive, all data interfaces are shut down; only the management-only interface can send and receive traffic. The management interface remains up using the IP address the unit received from DHCP or the cluster IP pool. If you use a cluster IP pool, if you reload and the unit is still inactive in the cluster, then the management interface is not accessible (because it then uses the Main IP address, which is the same as the control node). You must use the console port (if available) for any further configuration.

Cluster control link MTU ping testing

Use Cluster control link MTU ping testing to

- verify MTU compatibility between cluster nodes during the join process
- use ping-based fragmentation tests to confirm the network path supports proper packet handling, and
- allow nodes to join even if tests fail while providing diagnostic messages for troubleshooting MTU mismatches.

When a node joins the cluster, the control node send a ping with a payload size of twice the MTU. This process tests the network's ability to handle packet fragmentation because the underlying IP layer will fragment packets that exceed the MTU limit.

A successful ping confirms that the network path supports proper fragmentation and that the cluster control link can reliably process traffic at the configured MTU size.

If the ping fails, view these messages:

- **show cluster history**—`Event: CCL MTU test to unit name failed`
- Console WARNING—`WARNING: Unit name is not reachable in CCL jumbo frame ICMP test, please check cluster interface and switch MTU configuration`

Even if the ping fails, the node is allowed to join the cluster. In this case, you need to resolve the MTU mismatch as soon as possible.

Configuration Replication

All nodes in the cluster share a single configuration. You can only make configuration changes on the control node (with the exception of the bootstrap configuration), and changes are automatically synced to all other nodes in the cluster.

ASAv Cluster Management

One of the benefits of using ASAv clustering is the ease of management. This section describes how to manage the cluster.

Management Network

We recommend connecting all nodes to a single management network. This network is separate from the cluster control link.

Management Interface

Use the Management 0/0 interface for management.



Note You cannot enable dynamic routing for the management interface. You must use a static route.

You can use either static addressing or DHCP for the management IP address.

If you use static addressing, you can use a Main cluster IP address that is a fixed address for the cluster that always belongs to the current control node. For each interface, you also configure a range of addresses so that each node, including the current control node, can use a Local address from the range. The Main cluster IP address provides consistent management access to an address; when a control node changes, the Main cluster IP address moves to the new control node, so management of the cluster continues seamlessly. The Local IP address is used for routing, and is also useful for troubleshooting. For example, you can manage the cluster by connecting to the Main cluster IP address, which is always attached to the current control node. To manage an individual member, you can connect to the Local IP address. For outbound management traffic such as TFTP or syslog, each node, including the control node, uses the Local IP address to connect to the server.

If you use DHCP, you do not use a pool of Local addresses or have a Main cluster IP address.



Note To-the-box traffic needs to be directed to the node's management IP address; to-the-box traffic is not forwarded over the cluster control link to any other node.

Control Node Management Vs. Data Node Management

All management and monitoring can take place on the control node. From the control node, you can check runtime statistics, resource usage, or other monitoring information of all nodes. You can also issue a command to all nodes in the cluster, and replicate the console messages from data nodes to the control node.

You can monitor data nodes directly if desired. Although also available from the control node, you can perform file management on data nodes (including backing up the configuration and updating images). The following functions are not available from the control node:

- Monitoring per-node cluster-specific statistics.
- Syslog monitoring per node (except for syslogs sent to the console when console replication is enabled).
- SNMP
- NetFlow

Crypto Key Replication

When you create a crypto key on the control node, the key is replicated to all data nodes. If you have an SSH session to the Main cluster IP address, you will be disconnected if the control node fails. The new control node uses the same key for SSH connections, so that you do not need to update the cached SSH host key when you reconnect to the new control node.

ASDM Connection Certificate IP Address Mismatch

By default, a self-signed certificate is used for the ASDM connection based on the Local IP address. If you connect to the Main cluster IP address using ASDM, then a warning message about a mismatched IP address might appear because the certificate uses the Local IP address, and not the Main cluster IP address. You can ignore the message and establish the ASDM connection. However, to avoid this type of warning, you can enroll a certificate that contains the Main cluster IP address and all the Local IP addresses from the IP address pool. You can then use this certificate for each cluster member. See https://www.cisco.com/c/en/us/td/docs/security/asdm/identity-cert/cert-install.html for more information.

Inter-Site Clustering

For inter-site installations, you can take advantage of ASAv clustering as long as you follow the recommended guidelines.

You can configure each cluster chassis to belong to a separate site ID. Site IDs are used to enable flow mobility using LISP inspection, director localization to improve performance and reduce round-trip time latency for inter-site clustering for data centers, and site redundancy for connections where a backup owner of a traffic flow is always at a different site from the owner.

See the following sections for more information about inter-site clustering:

- Sizing the Data Center Interconnect—Requirements and Prerequisites for ASAv Clustering, on page 530
- Inter-Site Guidelines—Guidelines for ASAv Clustering, on page 530
- Configure Cluster Flow Mobility—Configure Cluster Flow Mobility, on page 553
- Enable Director Localization—Enable Director Localization, on page 551
- Enable Site Redundancy—Enable Director Localization, on page 551
- Inter-Site Examples—Individual Interface Routed Mode North-South Inter-Site Example, on page 573

Licenses for ASAv Clustering

Each cluster node requires the same model license. We recommend using the same number of CPUs and memory for all nodes, or else peformance will be limited on all nodes to match the least capable member. The throughput level will be replicated from the control node to each data node so they match.

Note If you deregister the ASAv so that it is unlicensed, then it will revert to a severely rate-limited state if you reload the ASAv. An unlicensed, low performing cluster node will impact the performance of the entire cluster negatively. Be sure to keep all cluster nodes licensed, or remove any unlicensed nodes.

Requirements and Prerequisites for ASAv Clustering

Model Requirements

- ASAv30, ASAv50, ASAv100
- The following private cloud services:
  - KVM
  - VMware
- A maximum of 16 nodes in a cluster on *two* hosts in a 2x8 deployment configuration. We recommend you to deploy a maximum of *eight* ASAvs on each of the *two* hosts (2x8), which results in a cluster of 16 nodes.

ASAv Platform and Software Requirements

All nodes in a cluster:

- Must be the same model. We recommend using the same number of CPUs and memory for all nodes, or else peformance will be limited on all nodes to match the least capable node.
- Must run the identical software except at the time of an image upgrade. Hitless upgrade is supported. Mismatched software versions can lead to poor performance, so be sure to upgrade all nodes in the same maintenance window.
- New cluster members must use the same SSL encryption setting (the **ssl encryption** command) as the control node for initial cluster control link communication before configuration replication.

Guidelines for ASAv Clustering

Failover

Failover is not supported with clustering.

IPv6

The cluster control link is only supported using IPv4.

Additional Guidelines

- When significant topology changes occur (such as enabling or disabling an interface on the ASA or the switch, adding an additional switch to form a VSS or vPC) you should disable the health check feature and also disable interface monitoring for the disabled interfaces. When the topology change is complete, and the configuration change is synced to all nodes, you can re-enable the interface health check feature.
- When adding a node to an existing cluster, or when reloading a node, there will be a temporary, limited packet/connection drop; this is expected behavior. In some cases, the dropped packets can hang your connection; for example, dropping a FIN/ACK packet for an FTP connection will make the FTP client hang. In this case, you need to reestablish the FTP connection.
- We do not support VXLANs for data interfaces; only the cluster control link supports VXLAN.
- It takes time to replicate changes to all the nodes in a cluster. If you make a large change, for example, adding an access control rule that uses object groups (which, when deployed, are broken out into multiple rules), the time needed to complete the change can exceed the timeout for the cluster nodes to respond with a success message. If this happens, you might see a "failed to replicate command" message. You can ignore the message.

Defaults for ASAv Clustering

- The cluster health check feature is enabled by default with the holdtime of 3 seconds. Interface health monitoring is enabled on all interfaces by default.
- The cluster auto-rejoin feature for a failed cluster control link is unlimited attempts every 5 minutes.
- The cluster auto-rejoin feature for a failed data interface is 3 attempts every 5 minutes, with the increasing interval set to 2.
- Connection rebalancing is disabled by default. If you enable connection rebalancing, the default time between load information exchanges is 5 seconds.
- Connection replication delay of 5 seconds is enabled by default for HTTP traffic.

Configure the ASAv Clustering Using a Day0 Configuration

Control Node Day0 Configuration

The following Day0 configuration for the control node includes the bootstrap configuration followed by interface configuration that will be replicated to the data nodes. Bold text shows the values you need to change for the data node Day0 configuration.



Note This configuration only includes the cluster-centric configuration. Your Day0 configuration should also include other settings for licensing, SSH access, ASDM access and more. See the getting started guide for more information about Day0 configurations.

```
!BOOTSTRAP
! Cluster interface mode
cluster interface mode individual
!
! VXLAN peer group
object-group network cluster-peers
network-object host 10.6.6.51
network-object host 10.6.6.52
network-object host 10.6.6.53
network-object host 10.6.6.54
!
! Alternate object group representation
! object-network xyz
! range 10.6.6.51 10.6.6.54
! object-group network cluster-peers
! network-object object xyz
!
! Cluster control link physical interface (VXLAN tunnel endpoint (VTEP) src interface)
interface gigabitethernet 0/7
description CCL VTEP src ifc
nve-only cluster
nameif ccl
security-level 0
ip address 10.6.6.51 255.255.255.0
no shutdown
!
! VXLAN Network Identifier (VNI) interface
interface vni1
segment-id 1
vtep-nve 1
!
! Set the CCL MTU
mtu ccl 1654
!
! Network Virtualization Endpoint (NVE) association with VTEP src interface
nve 1
encapsulation vxlan
source-interface ccl
peer-group cluster-peers
!
! Management Interface Using DHCP
interface management 0/0
nameif management
ip address dhcp setroute
no shutdown
!
! Alternate Management Using Static IP
! ip local pool mgmt_pool 10.1.1.1 10.10.10.4
! interface management 0/0
! nameif management
! ip address 10.1.1.25 255.255.255.0 cluster-pool mgmt_pool
! no shutdown
!
! Cluster Config
cluster group cluster1
local-unit A
cluster-interface vni1 ip 10.2.2.1 255.255.255.0
priority 1
enable noconfirm
!
! INTERFACES
!
ip local pool inside_pool 10.10.10.11 10.10.10.14
```

```
ip local pool outside_pool 10.11.11.11 10.11.11.14
!
interface GigabitEthernet0/1
nameif inside
security-level 100
ip address 10.10.10.10 255.255.255.0 cluster-pool inside_pool
!
interface GigabitEthernet0/0
nameif outside
security-level 0
ip address 10.11.11.10 255.255.255.0 cluster-pool outside_pool
!
!JUMBO FRAME RESERVATION for CCL MTU
jumbo-frame reservation
```

Data Node Day0 Configuration

The following Day0 configuration for the data node includes only the bootstrap configuration. Bold text shows the values you need to change from the control node Day0 configuration.



Note This configuration only includes the cluster-centric configuration. Your Day0 configuration should also include other settings for licensing, SSH access, ASDM access and more. See the getting started guide for more information about Day0 configurations.

```
!BOOTSTRAP
! Cluster interface mode
cluster interface mode individual
!
! VXLAN peer group
object-group network cluster-peers
network-object host 10.6.6.51
network-object host 10.6.6.52
network-object host 10.6.6.53
network-object host 10.6.6.54
!
! Alternate object group representation
! object-network xyz
! range 10.6.6.51 10.6.6.54
! object-group network cluster-peers
! network-object object xyz
!
! Cluster control link physical interface (VXLAN tunnel endpoint (VTEP) src interface)
interface gigabitethernet 0/7
description CCL VTEP src ifc
nve-only cluster
nameif ccl
security-level 0
ip address 10.6.6.52 255.255.255.0
no shutdown
!
! VXLAN Network Identifier (VNI) interface
interface vni1
segment-id 1
vtep-nve 1
!
! Set the CCL MTU
mtu ccl 1654
!
```

```
! Network Virtualization Endpoint (NVE) association with VTEP src interface
nve 1
encapsulation vxlan
source-interface ccl
peer-group cluster-peers
!
! Management Interface Using DHCP
interface management 0/0
nameif management
ip address dhcp setroute
no shutdown
!
! Alternate Management Using Static IP
! ip local pool mgmt_pool 10.1.1.1 10.10.10.4
! interface management 0/0
! nameif management
! ip address 10.1.1.25 255.255.255.0 cluster-pool mgmt_pool
! no shutdown
!
! Cluster Config
cluster group cluster1
local-unit B
cluster-interface vni1 ip 10.2.2.2 255.255.255.0
priority 2
enable noconfirm
!
! INTERFACES
!
ip local pool inside_pool 10.10.10.11 10.10.10.14
ip local pool outside_pool 10.11.11.11 10.11.11.14
!
interface GigabitEthernet0/1
nameif inside
security-level 100
ip address 10.10.10.10 255.255.255.0 cluster-pool inside_pool
!
interface GigabitEthernet0/0
nameif outside
security-level 0
ip address 10.11.11.10 255.255.255.0 cluster-pool outside_pool
!
!JUMBO FRAME RESERVATION for CCL MTU
jumbo-frame reservation
```

Configure ASAv Clustering after Deployment

To configure clustering after you deploy your ASAvs, perform the following tasks.

Configure Interface Settings

Configure the cluster interface mode on each node as well as interfaces on the control node. The interface configuration will be replicated to data nodes when they join the cluster. Note that configuration of the cluster control link is covered in the bootstrap configuration procedure.

Configure the Cluster Interface Mode on Each Node

Before you enable clustering, you need to convert the firewall to use Individual interfaces. Because clustering limits the types of interfaces you can use, this process lets you check your existing configuration for incompatible interfaces and then prevents you from configuring any unsupported interfaces.

Before you begin

- You must set the mode separately on each ASAv that you want to add to the cluster.
- Connect to the ASAv CLI using either the console port (if available) or SSH (if configured). If neither of these options is available, you can use ASDM to configure clustering.

Procedure

Step 1 Show any incompatible configuration so that you can force the interface mode and fix your configuration later; the mode is not changed with this command:

cluster interface-mode individual check-details

Example:

```
ciscoasa(config)# cluster interface-mode individual check-details
```

Caution
After you set the interface mode, you can continue to connect to the interface using SSH; however, if you reload the ASA before you configure your management interface to comply with clustering requirements (for example, adding a cluster IP pool or getting the IP address from DHCP), you will not be able to reconnect because cluster-incompatible interface configuration is removed. In that case, you will have to connect to the console port, if available, to fix the interface configuration.

Step 2 Set the interface mode for clustering:

cluster interface-mode individual force

Example:

```
ciscoasa(config)# cluster interface-mode individual force
```

There is no default setting; you must explicitly choose the mode. If you have not set the mode, you cannot enable clustering.

The **force** option changes the mode without checking your configuration for incompatible settings. You need to manually fix any configuration issues after you change the mode. Because any interface configuration can only be fixed after you set the mode, we recommend using the **force** option so that you can at least start from the existing configuration. You can re-run the **check-details** option after you set the mode for more guidance.

Without the **force** option, if there is any incompatible configuration, you are prompted to clear your configuration and reload, thus requiring you to connect to the console port (if available) to reconfigure your management access. If your configuration is compatible (rare), the mode is changed and the configuration is preserved. If you do not want to clear your configuration, you can exit the command by typing **n**.

To remove the interface mode, enter the **no cluster interface-mode** command.

Configure Individual Interfaces

You must modify any interface that is currently configured with an IP address to be cluster-ready before you enable clustering. At a minimum, you may need to modify the management interface to which SSH is currently connected when you use a static IP address for management. For other interfaces, you can configure them before or after you enable clustering; we recommend pre-configuring all of your interfaces so that the complete configuration is synced to new cluster nodes.

This section describes how to configure interfaces to be Individual interfaces compatible with clustering. Individual interfaces are normal routed interfaces, each with their own IP address taken from a pool of IP addresses. The Main cluster IP address is a fixed address for the cluster that always belongs to the current control node. All data interfaces must be Individual interfaces.

For the Management interface, you can configure an IP address pool or you can use DHCP; only the Management interface supports getting an address from DHCP. To use DHCP, do not use this procedure; instead configure it as usual (see Configure General Routed Mode Interface Parameters, on page 673).

Before you begin

- (Optional) Configure subinterfaces.
- For the management interface, you can use a static address or you can use DHCP. If you are using static IP addresses and connecting remotely to the management interface using SSH, the current IP address of prospective data nodes are for temporary use.
  - Each member will be assigned an IP address from the cluster IP pool defined on the control node.
  - The cluster IP pool cannot include addresses already in use on the network, including prospective secondary IP addresses.

    For example:

    1. You configure the control node to use 10.1.1.1.
    2. Other nodes use 10.1.1.2, 10.1.1.3, and 10.1.1.4.
    3. When you configure the cluster IP pool on the control node, you cannot include the .2, .3, or .4 addresses in the pool, because they are in use.
    4. Instead, you need to use other IP addresses on the network, such as .5, .6, .7, and .8.

Note The pool needs as many addresses as there are members of the cluster, including the control node; the original .1 address is the main cluster IP address that belongs to the current control node.

    5. After you join the cluster, the old, temporary addresses are relinquished and can be used elsewhere.

Procedure

Step 1 Configure a pool of Local IP addresses (IPv4 and/or IPv6), one of which will be assigned to each cluster node for the interface:

(IPv4)

ip local pool *poolname first-address — last-address* [**mask** *mask*]

(IPv6)

ipv6 local pool *poolname ipv6-address*/*prefix-length number_of_addresses*

Example:

```
ciscoasa(config)# ip local pool ins 192.168.1.2-192.168.1.9
ciscoasa(config-if)# ipv6 local pool insipv6 2001:DB8:45:1003/64 8
```

Include at least as many addresses as there are nodes in the cluster. If you plan to expand the cluster, include additional addresses. The Main cluster IP address that belongs to the current control node is *not* a part of this pool; be sure to reserve an IP address on the same network for the Main cluster IP address.

You cannot determine the exact Local address assigned to each node in advance; to see the address used on each node, enter the **show ip**[**v6**] **local pool** *poolname* command. Each cluster member is assigned a member ID when it joins the cluster. The ID determines the Local IP used from the pool.

Step 2 Enter interface configuration mode:

interface *interface_id*

Example:

```
ciscoasa(config)# interface gigabitethernet 0/1
```

Step 3 Name the interface:

nameif *name*

Example:

```
ciscoasa(config-if)# nameif inside
```

The *name* is a text string up to 48 characters, and is not case-sensitive. You can change the name by reentering this command with a new value.

Step 4 Set the Main cluster IP address and identify the cluster pool:

(IPv4)

ip address *ip_address* [*mask*] **cluster-pool** *poolname*

(IPv6)

ipv6 address *ipv6-address*/*prefix-length* **cluster-pool** *poolname*

Example:

```
ciscoasa(config-if)# ip address 192.168.1.1 255.255.255.0 cluster-pool ins
ciscoasa(config-if)# ipv6 address 2001:DB8:45:1003::99/64 cluster-pool insipv6
```

This IP address must be on the same network as the cluster pool addresses, but not be part of the pool. You can configure an IPv4 and/or an IPv6 address.

DHCP, PPPoE, and IPv6 autoconfiguration are not supported; you must manually configure the IP addresses. Manually configuring the link-local address is also not supported.

Step 5 Set the security level, where *number* is an integer between 0 (lowest) and 100 (highest):

security-level *number*

Example:

```
ciscoasa(config-if)# security-level 100
```

Step 6 Enable the interface:

no shutdown

Examples

The following example configures the Management 0/0, GigabitEthernet 0/0, and GigabitEthernet 0/1 interfaces as Individual interfaces:

```
ip local pool mgmt 10.1.1.2-10.1.1.9
ipv6 local pool mgmtipv6 2001:DB8:45:1002/64 8

interface management 0/0
nameif management
ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
ipv6 address 2001:DB8:45:1001::99/64 cluster-pool mgmtipv6
security-level 100
no shutdown

ip local pool out 209.165.200.225-209.165.200.232
ipv6 local pool outipv6 2001:DB8:45:1002/64 8

interface gigabitethernet 0/0
nameif outside
ip address 209.165.200.233 255.255.255.224 cluster-pool out
ipv6 address 2001:DB8:45:1002::99/64 cluster-pool outipv6
security-level 0
no shutdown

ip local pool ins 192.168.1.2-192.168.1.9
ipv6 local pool insipv6 2001:DB8:45:1003/64 8

interface gigabitethernet 0/1
nameif inside
ip address 192.168.1.1 255.255.255.0 cluster-pool ins
ipv6 address 2001:DB8:45:1003::99/64 cluster-pool insipv6
security-level 100
no shutdown
```

Create the Bootstrap Configuration

Each node in the cluster requires a bootstrap configuration to join the cluster.

Configure Control Node Bootstrap Settings

Each node in the cluster requires a bootstrap configuration to join the cluster. Typically, the first node you configure to join the cluster will be the control node. After you enable clustering, after an election period, the cluster elects a control node. With only one node in the cluster initially, that node will become the control node. Subsequent nodes that you add to the cluster will be data nodes.

Before you begin

- Back up your configurations in case you later want to leave the cluster, and need to restore your configuration.
- With the exception of the cluster control link and the Management interface, which can optionally use DHCP, any interfaces in your configuration must be configured with a cluster IP pool before you enable clustering. If you have pre-existing interface configuration, you can either clear the interface configuration (**clear configure interface**), or convert your interfaces to cluster interfaces before you enable clustering.
- When you add a node to a running cluster, you may see temporary, limited packet/connection drops; this is expected behavior.
- Enable jumbo frame reservation for use with the cluster control link, so you can set the cluster control link MTU to the recommended value. See the **jumbo-frame reservation** command. Enabling jumbo frames causes the ASA to reload, so you must perform this step before continuing with this procedure.

Procedure

Step 1 Configure a VXLAN interface for the cluster control link interface before you join the cluster.

You will later identify this interface as the cluster control link when you enable clustering.

The cluster control link interface configuration is not replicated from the control node to data nodes; however, you must use the same configuration on each node. Because this configuration is not replicated, you must configure the cluster control link interfaces separately on each node.

a) Identify the VTEP peer IP addresses by creating a network object group.

See the "Objects for Access Control" chapter in the ASA firewall configuration guide for more information about network object groups.

The underlying IP network between VTEPs is independent of the cluster control link network that the VNI interfaces use. Each VTEP source interface has an IP address on the same subnet. This subnet should be isolated from all other traffic, and should include only the cluster control link interfaces.

Example:

The following example creates a network object group with hosts defined inline:

```
ciscoasa(config)# object-group network cluster-peers
ciscoasa(network-object-group)# network-object host 10.6.6.51
ciscoasa(network-object-group)# network-object host 10.6.6.52
```

```
ciscoasa(network-object-group)# network-object host 10.6.6.53
ciscoasa(network-object-group)# network-object host 10.6.6.54
```

The following example creates a network object group that refers to a standalone network object:

```
ciscoasa(config)# object network xyz
ciscoasa(config-network-object)# range 10.6.6.51 10.6.6.54

ciscoasa(config)# object-group network cluster-peers
ciscoasa(network-object-group)# network-object object xyz
```

b) Configure the VTEP source interface.

interface *interface_id*

nve-only cluster

nameif *name*

ip address *ip_address subnet_mask*

no shutdown

The IP address should be included as one of the peers in the network object group.

Example:

```
ciscoasa(config)# interface gigabitethernet 0/7
ciscoasa(config-if)# nve-only cluster
ciscoasa(config-if)# nameif ccl
ciscoasa(config-if)# ip address 10.6.6.51 255.255.255.0
ciscoasa(config-if)# no shutdown
```

c) Associate the VTEP source interface with the NVE instance.

nve 1

source-interface *interface-name*

peer-group *network_object_name*

You can only specify one NVE instance, with the ID 1.

The **encapsulation vxlan** command is added by default for the NVE instance; you do not need to explicitly add it.

Example:

```
ciscoasa(config)# nve 1
ciscoasa(cfg-nve)# source-interface ccl
ciscoasa(cfg-nve)# peer-group cluster-peers
```

d) Specify the maximum transmission unit for the VTEP source interface to be at least 154 bytes higher than the highest MTU of the data interfaces.

mtu *interface_name bytes*

Because the cluster control link traffic includes data packet forwarding, the cluster control link needs to accommodate the entire size of a data packet plus cluster traffic overhead (100 bytes) and VXLAN overhead (54 bytes). Set the MTU between 1554 and 9198 bytes, but not between 2561 and 8362. Due

to block pool handling, this MTU size is not optimal for system operation. The default MTU is 1554 bytes. We suggest setting the cluster control link MTU to 1654 when data interfaces are set to 1500; this value requires jumbo frame reservation (see the **jumbo-frame reservation** command).

For example, when using jumbo frames, because the maximum MTU is 9198 bytes, then the highest data interface MTU can be 9044, while the cluster control link can be set to 9198.

This command is replicated to data nodes, but we recommend you configure this setting along with the bootstrap settings.

Example:

```
ciscoasa(config)# mtu ccl 1654
```

e) (Optional) Set the VXLAN UDP port.

vxlan *port number*

By default, the VTEP source interface accepts VXLAN traffic to UDP port 4789. If your network uses a non-standard port, you can change it.

Example:

```
ciscoasa(config)# vxlan port 5678
```

f) Create the VNI interface.

interface vni *vni_num*

segment-id *id*

vtep-nve 1

Example:

```
ciscoasa(config)# interface vni 1
ciscoasa(config-if)# segment-id 1000
ciscoasa(config-if)# vtep-nve 1
```

- Set the VNI number between 1 and 10000. This ID is only an internal interface identifier.
- Set the segment ID between 1 and 16777215. The segment ID is used for VXLAN tagging.

Do *not* configure a name for the interface or any other parameters.

Step 2 Name the cluster and enter cluster configuration mode:

cluster group *name*

Example:

```
ciscoasa(config)# cluster group pod1
```

The name must be an ASCII string from 1 to 38 characters. You can only configure one cluster group per node. All members of the cluster must use the same name.

Step 3 Name this member of the cluster:

local-unit *node_name*

Use a unique ASCII string from 1 to 38 characters. Each node must have a unique name. A node with a duplicated name will not be allowed in the cluster.

Example:

```
ciscoasa(cfg-cluster)# local-unit node1
```

Step 4 Specify the cluster control link VNI interface:

cluster-interface *vni_interface_id* **ip** *ip_address mask*

Example:

```
ciscoasa(cfg-cluster)# cluster-interface vni1 ip 192.168.1.1 255.255.255.0
INFO: Non-cluster interface config is cleared on VNI1
```

Specify an IPv4 address for the IP address; IPv6 is not supported for this interface. For each node, specify a different IP address on the same network. The VNI network is the encrypted virtual network that runs on top of the physical VTEP network.

Step 5 Set the priority of this node for control node elections:

priority *priority_number*

Example:

```
ciscoasa(cfg-cluster)# priority 1
```

The priority is between 1 and 100, where 1 is the highest priority.

Step 6 (Optional) Set an authentication key for control traffic on the cluster control link:

key *shared_secret*

Example:

```
ciscoasa(cfg-cluster)# key chuntheunavoidable
```

The shared secret is an ASCII string from 1 to 63 characters. The shared secret is used to generate the key. This command does not affect datapath traffic, including connection state update and forwarded packets, which are always sent in the clear.

Step 7 Enable clustering:

enable [**noconfirm**]

Example:

```
ciscoasa(cfg-cluster)# enable
INFO: Clustering is not compatible with following commands:
policy-map global_policy
 class inspection_default
 inspect skinny
policy-map global_policy
 class inspection_default
  inspect sip
```

```
 Would you like to remove these commands? [Y]es/[N]o:Y

INFO: Removing incompatible commands from running configuration...
Cryptochecksum (changed): f16b7fc2 a742727e e40bc0b0 cd169999
INFO: Done
```

When you enter the **enable** command, the ASA scans the running configuration for incompatible commands for features that are not supported with clustering, including commands that may be present in the default configuration. You are prompted to delete the incompatible commands. If you respond **No**, then clustering is not enabled. Use the **noconfirm** keyword to bypass the confirmation and delete incompatible commands automatically.

For the first node enabled, a control node election occurs. Because the first node should be the only member of the cluster so far, it will become the control node. Do not perform any configuration changes during this period.

To disable clustering, enter the **no enable** command.

Note
If you disable clustering, all data interfaces are shut down, and only the management interface is active.

Examples

The following example configures the management, inside, and outside interfaces and the VXLAN cluster control link, and then enables clustering for the ASA called “node1,” which will become the control node because it is added to the cluster first:

```
ip local pool mgmt 10.1.1.2-10.1.1.9
ipv6 local pool mgmtipv6 2001:DB8:45:1002/64 8

interface management 0/0
    nameif management
    ip address 10.1.1.1 255.255.255.0 cluster-pool mgmt
    ipv6 address 2001:DB8:45:1001::99/64 cluster-pool mgmtipv6
    security-level 100
    no shutdown

ip local pool out 209.165.200.225-209.165.200.232
ipv6 local pool outipv6 2001:DB8:45:1002/64 8

interface gigabitethernet 0/0
    nameif outside
    ip address 209.165.200.233 255.255.255.224 cluster-pool out
    ipv6 address 2001:DB8:45:1002::99/64 cluster-pool outipv6
    security-level 0
    no shutdown

ip local pool ins 192.168.1.2-192.168.1.9
ipv6 local pool insipv6 2001:DB8:45:1003/64 8

interface gigabitethernet 0/1
    nameif inside
    ip address 192.168.1.1 255.255.255.0 cluster-pool ins
    ipv6 address 2001:DB8:45:1003::99/64 cluster-pool insipv6
    security-level 100
    no shutdown
```

```
object-group network cluster-peers
    network-object host 10.6.6.51
    network-object host 10.6.6.52
    network-object host 10.6.6.53
    network-object host 10.6.6.54

interface gigabitethernet 0/7
    nve-only cluster
    nameif ccl
    ip address 10.6.6.51 255.255.255.0
    no shutdown

nve 1
    source-interface ccl
    peer-group cluster-peers

mtu ccl 1654

interface vni 1
    segment-id 1000
    vtep-nve 1

cluster group pod1
    local-unit node1
    cluster-interface vni1 ip 192.168.1.1 255.255.255.0
    priority 1
    key 67impala
    enable noconfirm
```

Configure Data Node Bootstrap Settings

Perform the following procedure to configure the data nodes.

Before you begin

- Back up your configurations in case you later want to leave the cluster, and need to restore your configuration.
- With the exception of the cluster control link and the Management interface, which can optionally use DHCP, any interfaces in your configuration must be configured with a cluster IP pool before you enable clustering. If you have pre-existing interface configuration, you can either clear the interface configuration (**clear configure interface**), or convert your interfaces to cluster interfaces before you enable clustering.
- When you add a node to a running cluster, you may see temporary, limited packet/connection drops; this is expected behavior.
- Enable jumbo frame reservation for use with the cluster control link, so you can set the cluster control link MTU to the recommended value. See the **jumbo-frame reservation** command. Enabling jumbo frames causes the ASA to reload, so you must perform this step before continuing with this procedure.

Procedure

Step 1 Configure the same cluster control link interface as you configured for the control node. Be sure to supply a different IP address for the VTEP source interface (shown in **bold**).

Example:

```
ciscoasa(config)# object-group network cluster-peers
ciscoasa(network-object-group)# network-object host 10.6.6.51
ciscoasa(network-object-group)# network-object host 10.6.6.52
ciscoasa(network-object-group)# network-object host 10.6.6.53
ciscoasa(network-object-group)# network-object host 10.6.6.54
ciscoasa(config)# interface gigabitethernet 0/7
ciscoasa(config-if)# nve-only cluster
ciscoasa(config-if)# nameif ccl
ciscoasa(config-if)# ip address 10.6.6.52 255.255.255.0
ciscoasa(config-if)# no shutdown
ciscoasa(config)# nve 1
ciscoasa(cfg-nve)# source-interface ccl
ciscoasa(cfg-nve)# peer-group cluster-peers
ciscoasa(config)# mtu ccl 1654
ciscoasa(config)# interface vni 1
ciscoasa(config-if)# segment-id 1000
ciscoasa(config-if)# vtep-nve 1
```

Step 2 Identify the same cluster name that you configured for the control node:

Example:

```
ciscoasa(config)# cluster group pod1
```

Step 3 Name this member of the cluster with a unique string:

local-unit *node_name*

Example:

```
ciscoasa(cfg-cluster)# local-unit node2
```

Specify an ASCII string from 1 to 38 characters.

Each node must have a unique name. A node with a duplicated name will be not be allowed in the cluster.

Step 4 Specify the same cluster control link interface that you configured for the control node, but specify a different IP address on the same network for each node:

cluster-interface *vni_interface_id* **ip** *ip_address mask*

Example:

```
ciscoasa(cfg-cluster)# cluster-interface vni1 ip 192.168.1.2 255.255.255.0
INFO: Non-cluster interface config is cleared on VNI1
```

Specify an IPv4 address for the IP address; IPv6 is not supported for this interface. This interface cannot have a **nameif** configured.

Step 5 If you use inter-site clustering, set the site ID for this node so it uses a site-specific MAC address:

site-id *number*

Example:

```
ciscoasa(cfg-cluster)# site-id 2
```

The **number** is between 1 and 8.

Step 6 Set the priority of this node for control node elections, typically to a higher value than the control node:

priority *priority_number*

Example:

```
ciscoasa(cfg-cluster)# priority 2
```

Set the priority between 1 and 100, where 1 is the highest priority.

Step 7 Set the same authentication key that you set for the control node:

Example:

```
ciscoasa(cfg-cluster)# key chuntheunavoidable
```

Step 8 Enable clustering:

enable as-slave

You can avoid any configuration incompatibilities (primarily the existence of any interfaces not yet configured for clustering) by using the **enable as-slave** command. This command ensures the data node joins the cluster with no possibility of becoming the control node in any current election. Its configuration is overwritten with the one synced from the control node.

To disable clustering, enter the **no enable** command.

Note
If you disable clustering, all data interfaces are shut down, and only the management interface is active.

Examples

The following example includes the configuration for a data node, node2:

```
object-group network cluster-peers
    network-object host 10.6.6.51
    network-object host 10.6.6.52
    network-object host 10.6.6.53
    network-object host 10.6.6.54

interface gigabitethernet 0/7
    nve-only cluster
    nameif ccl
    ip address 10.6.6.52 255.255.255.0
    no shutdown

nve 1
    source-interface ccl
    peer-group cluster-peers
```

```
mtu ccl 1654

interface vni 1
    segment-id 1000
    vtep-nve 1

cluster group pod1
    local-unit node2
    cluster-interface vni1 ip 192.168.1.2 255.255.255.0
    priority 2
    key 67impala
    enable noconfirm
```

Customize the Clustering Operation

You can customize clustering health monitoring, TCP connection replication delay, flow mobility and other optimizations, either as part of the Day 0 configuration or after you deploy the cluster.

Perform these procedures on the control node.

Configure Basic ASA Cluster Parameters

You can customize cluster settings on the control node.

Procedure

Step 1 Enter cluster configuration mode:

cluster group *name*

Step 2 (Optional) Enable console replication from data nodes to the control node:

console-replicate

This feature is disabled by default. The ASA prints out some messages directly to the console for certain critical events. If you enable console replication, data nodes send the console messages to the control node so that you only need to monitor one console port for the cluster.

Step 3 Set the minimum trace level for clustering events:

trace-level *level*

Set the minimum level as desired:

- **critical**—Critical events (severity=1)
- **warning**—Warnings (severity=2)
- **informational**—Informational events (severity=3)
- **debug**—Debugging events (severity=4)

Configure Health Monitoring and Auto-Rejoin Settings

This procedure configures node and interface health monitoring.

You might want to disable health monitoring of non-essential interfaces, for example, the management interface. Health monitoring is not performed on VLAN subinterfaces. You cannot configure monitoring for the cluster control link; it is always monitored.

Procedure

Step 1 Enter cluster configuration mode.

cluster group *name*

Example:

```
ciscoasa(config)# cluster group test
ciscoasa(cfg-cluster)#
```

Step 2 Customize the cluster node health check feature.

health-check [**holdtime** *timeout*]

To determine node health, the ASA cluster nodes send heartbeat messages on the cluster control link to other nodes. If a node does not receive any heartbeat messages from a peer node within the holdtime period, the peer node is considered unresponsive or dead.

- **holdtime** *timeout*—Determines the amount of time between node heartbeat status messages, between .3 and 45 seconds; The default is 3 seconds.

When any topology changes occur (such as adding or removing a data interface, enabling or disabling an interface on the ASA or the switch) you should disable the health check feature and also disable interface monitoring for the disabled interfaces (**no health-check monitor-interface**). When the topology change is complete, and the configuration change is synced to all nodes, you can re-enable the health check feature.

Example:

```
ciscoasa(cfg-cluster)# health-check holdtime 5
```

Step 3 Disable the interface health check on an interface.

no health-check monitor-interface *interface_id*

The interface health check monitors for link failures. The amount of time before the ASA removes a member from the cluster depends on whether the node is an established member or is joining the cluster. Health check is enabled by default for all interfaces. You can disable it per interface using the **no** form of this command. You might want to disable health monitoring of non-essential interfaces, for example, the management interface.

- *interface_id*—Disables monitoring of an interface. Health monitoring is not performed on VLAN subinterfaces. You cannot configure monitoring for the cluster control link; it is always monitored.

When any topology changes occur (such as adding or removing a data interface, enabling or disabling an interface on the ASA or the switch) you should disable the health check feature (**no health-check**) and also disable interface monitoring for the disabled interfaces. When the topology change is complete, and the configuration change is synced to all nodes, you can re-enable the health check feature.

Example:

```
ciscoasa(cfg-cluster)# no health-check monitor-interface management1/1
```

Step 4 Customize the auto-rejoin cluster settings after a health check failure.

health-check {**data-interface** | **cluster-interface** | **system**} **auto-rejoin** [**unlimited** | *auto_rejoin_max*] *auto_rejoin_interval auto_rejoin_interval_variation*

- **system**—Specifies the auto-rejoin settings for internal errors. Internal failures include: application sync timeout; inconsistent application statuses; and so on.
- **unlimited**—(Default for the **cluster-interface**) Does not limit the number of rejoin attempts.
- *auto-rejoin-max*—Sets the number of rejoin attempts, between 0 and 65535. **0** disables auto-rejoining. The default for the **data-interface** and **system** is 3.
- *auto_rejoin_interval*—Defines the interval duration in minutes between rejoin attempts, between 2 and 60. The default value is 5 minutes. The maximum total time that the node attempts to rejoin the cluster is limited to 14400 minutes (10 days) from the time of last failure.
- *auto_rejoin_interval_variation*—Defines if the interval duration increases. Set the value between 1 and 3: **1** (no change); **2** (2 x the previous duration), or **3** (3 x the previous duration). For example, if you set the interval duration to 5 minutes, and set the variation to 2, then the first attempt is after 5 minutes; the 2nd attempt is 10 minutes (2 x 5); the 3rd attempt 20 minutes (2 x 10), and so on. The default value is **1** for the cluster-interface and **2** for the data-interface and system.

Example:

```
ciscoasa(cfg-cluster)# health-check data-interface auto-rejoin 10 3 3
```

Step 5 Configure the debounce time before the ASA considers an interface to be failed and the node is removed from the cluster.

health-check monitor-interface debounce-time *ms*

Example:

```
ciscoasa(cfg-cluster)# health-check monitor-interface debounce-time 300
```

Set the debounce time between 300 and 9000 ms. The default is 500 ms. Lower values allow for faster detection of interface failures. Note that configuring a lower debounce time increases the chances of false-positives. When an interface status update occurs, the ASA waits the number of milliseconds specified before marking the interface as failed and the node is removed from the cluster.

Step 6 (Optional) Configure traffic load monitoring.

load-monitor [**frequency** *seconds*] [**intervals** *intervals*]

- **frequency** *seconds*—Sets the time in seconds between monitoring messages, between 10 and 360 seconds. The default is 20 seconds.
- **intervals** *intervals*—Sets the number of intervals for which the ASA maintains data, between 1 and 60. The default is 30.

You can monitor the traffic load for cluster members, including total connection count, CPU and memory usage, and buffer drops. If the load is too high, you can choose to manually disable clustering on the node if the remaining nodes can handle the load, or adjust the load balancing on the external switch. This feature is enabled by default. You can periodically monitor the traffic load. If the load is too high, you can choose to manually disable clustering on the node.

Use the **show cluster info load-monitor** command to view the traffic load.

Example:

```
ciscoasa(cfg-cluster)# load-monitor frequency 50 intervals 25
ciscoasa(cfg-cluster)# show cluster info load-monitor
ID  Unit Name
0  B
1  A_1
Information from all units with 50 second interval:
Unit     Connections    Buffer Drops   Memory Used   CPU Used
Average from last 1 interval:
   0           0            0            14              25
   1           0            0            16              20
Average from last 25 interval:
   0           0            0            12              28
   1           0            0            13              27
```

Example

The following example configures the health-check holdtime to .3 seconds; disables monitoring on the Management 0/0 interface; sets the auto-rejoin for data interfaces to 4 attempts starting at 2 minutes, increasing the duration by 3 x the previous interval; and sets the auto-rejoin for the cluster control link to 6 attempts every 2 minutes.

```
ciscoasa(config)# cluster group test
ciscoasa(cfg-cluster)# health-check holdtime .3
ciscoasa(cfg-cluster)# no health-check monitor-interface management0/0
ciscoasa(cfg-cluster)# health-check data-interface auto-rejoin 4 2 3
ciscoasa(cfg-cluster)# health-check cluster-interface auto-rejoin 6 2 1
```

Configure Connection Rebalancing and the Cluster TCP Replication Delay

You can configure connection rebalancing. If the load balancing capabilities of the upstream or downstream routers result in unbalanced flow distribution, you can configure overloaded nodes to redirect new TCP flows to other nodes. No existing flows will be moved to other nodes.

Enable the cluster replication delay for TCP connections to help eliminate the “unnecessary work” related to short-lived flows by delaying the director/backup flow creation. Note that if a node fails before the

director/backup flow is created, then those flows cannot be recovered. Similarly, if traffic is rebalanced to a different node before the flow is created, then the flow cannot be recovered. You should not enable the TCP replication delay for traffic on which you disable TCP randomization.

Procedure

Step 1 Enable the cluster replication delay for TCP connections:

cluster replication delay *seconds* {**http | match tcp {host** *ip_address* | *ip_address mask* | **any | any4 | any6} [{eq | lt | gt}** *port*] {**host** *ip_address* | *ip_address mask* | **any | any4 | any6} [{eq | lt | gt}** *port*]}

Example:

```
ciscoasa(config)# cluster replication delay 15 match tcp any any eq ftp
ciscoasa(config)# cluster replication delay 15 http
```

Set the *seconds* between 1 and 15. The **http** delay is enabled by default for 5 seconds.

Step 2 Enter cluster configuration mode:

cluster group *name*

Step 3 (Optional) Enable connection rebalancing for TCP traffic:

conn-rebalance [**frequency** *seconds*]

Example:

```
ciscoasa(cfg-cluster)# conn-rebalance frequency 60
```

This command is disabled by default. If enabled, ASAs exchange information about the connections per second periodically, and offload new connections from devices with more connections per second to less loaded devices. Existing connections are never moved. Moreover, because this command only rebalances based on connections per second, the total number of established connections on each node is not considered, and the total number of connections may not be equal. The frequency, between 1 and 360 seconds, specifies how often the load information is exchanged. The default is 5 seconds.

Once a connection is offloaded to a different node, it becomes an asymmetric connection.

Do not configure connection rebalancing for inter-site topologies; you do not want connections rebalanced to cluster members at a different site.

Configure Inter-Site Features

For inter-site clustering, you can customize your configuration to enhance redundancy and stability.

Enable Director Localization

To improve performance and reduce round-trip time (RTT) for inter-site clustering, you can enable director localization. Director localization adds two director roles: a local director and a global director.

When a cluster member receives a new-flow packet and localization applies, the receiving member uses its own site ID to select a local director from members at the same site. The global director can be at any site. Director localization optimizes same-site flow setup for common cases, but it does not guarantee that every director query or owner selection remains at the ingress site.

The cluster uses the global director instead of the local director in these conditions:

- The local-director hash and global-director hash resolve to the same member.
- The receiving member is selected as the local director.
- The flow has a NAT or PAT translation.
- The flow is SCTP traffic.
- The flow is a TCP-proxy server-certificate probe connection.
- The flow is a reassembled fragment chain.
- The local data interface is not ready.



Note Cross-site director-query output in **packet-tracer** does not by itself indicate a cluster failure. If packets are dropped, verify routing on each cluster member. Cluster members run independent routing processes, and routing tables are not synchronized over the cluster control link.

Before you begin

- Set the site ID for the cluster member in the bootstrap configuration.
- The following traffic types do not support localization: NAT or PAT traffic; SCTP-inspected traffic; Fragmentation owner query.

Procedure

Step 1 Enter cluster configuration mode.

cluster group *name*

Example:

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)#
```

Step 2 Enable director localization.

director-localization

Enable Site Redundancy

To protect flows from a site failure, you can enable site redundancy. If the connection backup owner is at the same site as the owner, then an additional backup owner will be chosen from another site to protect flows from a site failure.

Before you begin

- Set the site ID for the cluster member in the bootstrap configuration.

Procedure

Step 1 Enter cluster configuration mode.

cluster group *name*

Example:

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)#
```

Step 2 Enable site redundancy.

site-redundancy

Configure Cluster Flow Mobility

You can inspect LISP traffic to enable flow mobility when a server moves between sites.

About LISP Inspection

You can inspect LISP traffic to enable flow mobility between sites.

About LISP

Data center virtual machine mobility such as VMware VMotion enables servers to migrate between data centers while maintaining connections to clients. To support such data center server mobility, routers need to be able to update the ingress route towards the server when it moves. Cisco Locator/ID Separation Protocol (LISP) architecture separates the device identity, or endpoint identifier (EID), from its location, or routing locator (RLOC), into two different numbering spaces, making server migration transparent to clients. For example, when a server moves to a new site and a client sends traffic to the server, the router redirects traffic to the new location.

LISP requires routers and servers in certain roles, such as the LISP egress tunnel router (ETR), ingress tunnel router (ITR), first hop routers, map resolver (MR), and map server (MS). When the first hop router for the server senses that the server is connected to a different router, it updates all of the other routers and databases so that the ITR connected to the client can intercept, encapsulate, and send traffic to the new server location.

ASA LISP Support

The ASA does not run LISP itself; it can, however, inspect LISP traffic for location changes and then use this information for seamless clustering operation. Without LISP integration, when a server moves to a new site, traffic comes to an ASA cluster member at the new site instead of to the original flow owner. The new ASA forwards traffic to the ASA at the old site, and then the old ASA has to send traffic back to the new site to reach the server. This traffic flow is sub-optimal and is known as "tromboning" or "hair-pinning."

With LISP integration, the ASA cluster members can inspect LISP traffic passing between the first hop router and the ETR or ITR, and can then change the flow owner to be at the new site.

LISP Guidelines

- The ASA cluster members must reside between the first hop router and the ITR or ETR for the site. The ASA cluster itself cannot be the first hop router for an extended segment.
- Only fully-distributed flows are supported; centralized flows, semi-distributed flows, or flows belonging to individual nodes are not moved to new owners. Semi-distributed flows include applications, such as SIP, where all child flows are owned by the same ASA that owns the parent flow.
- The cluster only moves Layer 3 and 4 flow states; some application data might be lost.
- For short-lived flows or non-business-critical flows, moving the owner may not be worthwhile. You can control the types of traffic that are supported with this feature when you configure the inspection policy, and should limit flow mobility to essential traffic.

ASA LISP Implementation

This feature includes several inter-related configurations (all of which are described in this chapter):

1. (Optional) Limit inspected EIDs based on the host or server IP address—The first hop router might send EID-notify messages for hosts or networks the ASA cluster is not involved with, so you can limit the EIDs to only those servers or networks relevant to your cluster. For example, if the cluster is only involved with 2 sites, but LISP is running on 3 sites, you should only include EIDs for the 2 sites involved with the cluster.
2. LISP traffic inspection—The ASA inspects LISP traffic on UDP port 4342 for the EID-notify message sent between the first hop router and the ITR or ETR. The ASA maintains an EID table that correlates the EID and the site ID. For example, you should inspect LISP traffic with a source IP address of the first hop router and a destination address of the ITR or ETR. Note that LISP traffic is not assigned a director, and LISP traffic itself does not participate in cluster state sharing.
3. Service Policy to enable flow mobility on specified traffic—You should enable flow mobility on business-critical traffic. For example, you can limit flow mobility to only HTTPS traffic, and/or to traffic to specific servers.
4. Site IDs—The ASA uses the site ID for each cluster node to determine the new owner.
5. Cluster-level configuration to enable flow mobility—You must also enable flow mobility at the cluster level. This on/off toggle lets you easily enable or disable flow mobility for a particular class of traffic or applications.

Configure LISP Inspection

You can inspect LISP traffic to enable flow mobility when a server moves between sites.

Before you begin

- Assign each cluster unit to a site ID according to Configure Control Node Bootstrap Settings, on page 539 and Configure Data Node Bootstrap Settings, on page 544.
- LISP traffic is not included in the default-inspection-traffic class, so you must configure a separate class for LISP traffic as part of this procedure.

Procedure

Step 1 (Optional) Configure a LISP inspection map to limit inspected EIDs based on IP address, and to configure the LISP pre-shared key:

a) Create an extended ACL; only the destination IP address is matched to the EID embedded address:

access list *eid_acl_name* **extended permit ip** *source_address mask destination_address mask*

Both IPv4 and IPv6 ACLs are accepted. See the command reference for exact **access-list extended** syntax.

b) Create the LISP inspection map, and enter parameters mode:

policy-map type inspect lisp *inspect_map_name*

parameters

c) Define the allowed EIDs by identifying the ACL you created:

allowed-eid access-list *eid_acl_name*

The first hop router or ITR/ETR might send EID-notify messages for hosts or networks that the ASA cluster is not involved with, so you can limit the EIDs to only those servers or networks relevant to your cluster. For example, if the cluster is only involved with 2 sites, but LISP is running on 3 sites, you should only include EIDs for the 2 sites involved with the cluster.

d) If necessary, enter the pre-shared key:

validate-key *key*

Example:

```
ciscoasa(config)# access-list TRACKED_EID_LISP extended permit ip any 10.10.10.0 255.255.255.0
ciscoasa(config)# policy-map type inspect lisp LISP_EID_INSPECT
ciscoasa(config-pmap)# parameters
ciscoasa(config-pmap-p)# allowed-eid access-list TRACKED_EID_LISP
ciscoasa(config-pmap-p)# validate-key MadMaxShinyandChrome
```

Step 2 Configure LISP inspection for UDP traffic between the first hop router and the ITR or ETR on port 4342:

a) Configure the extended ACL to identify LISP traffic:

access list *inspect_acl_name* **extended permit udp** *source_address mask destination_address mask* **eq 4342**

You *must* specify UDP port 4342. Both IPv4 and IPv6 ACLs are accepted. See the command reference for exact **access-list extended** syntax.

b) Create a class map for the ACL:

class-map *inspect_class_name*

match access-list *inspect_acl_name*

c) Specify the policy map, the class map, enable inspection using the optional LISP inspection map, and apply the service policy to an interface (if new):

policy-map *policy_map_name*

class *inspect_class_name*

inspect lisp [*inspect_map_name*]

service-policy *policy_map_name* {**global** | **interface** *ifc_name*}

If you have an existing service policy, specify the existing policy map name. By default, the ASA includes a global policy called **global_policy**, so for a global policy, specify that name. You can also create one service policy per interface if you do not want to apply the policy globally. LISP inspection is applied to traffic bidirectionally so you do not need to apply the service policy on both the source and destination interfaces; all traffic that enters or exits the interface to which you apply the policy map is affected if the traffic matches the class map for both directions.

Example:

```
ciscoasa(config)# access-list LISP_ACL extended permit udp host 192.168.50.89 host
192.168.10.8 eq 4342
ciscoasa(config)# class-map LISP_CLASS
ciscoasa(config-cmap)# match access-list LISP_ACL
ciscoasa(config-cmap)# policy-map INSIDE_POLICY
ciscoasa(config-pmap)# class LISP_CLASS
ciscoasa(config-pmap-c)# inspect lisp LISP_EID_INSPECT
ciscoasa(config)# service-policy INSIDE_POLICY interface inside
```

The ASA inspects LISP traffic for the EID-notify message sent between the first hop router and the ITR or ETR. The ASA maintains an EID table that correlates the EID and the site ID.

Step 3 Enable Flow Mobility for a traffic class:

a) Configure the extended ACL to identify business critical traffic that you want to re-assign to the most optimal site when servers change sites:

access list *flow_acl_name* **extended permit udp** *source_address mask destination_address mask* **eq** *port*

Both IPv4 and IPv6 ACLs are accepted. See the command reference for exact **access-list extended** syntax. You should enable flow mobility on business-critical traffic. For example, you can limit flow mobility to only HTTPS traffic, and/or to traffic to specific servers.

b) Create a class map for the ACL:

class-map *flow_map_name*

match access-list *flow_acl_name*

c) Specify the same policy map on which you enabled LISP inspection, the flow class map, and enable flow mobility:

policy-map *policy_map_name*

class *flow_map_name*

cluster flow-mobility lisp

Example:

```
ciscoasa(config)# access-list IMPORTANT-FLOWS extended permit tcp any 10.10.10.0 255.255.255.0
 eq https
ciscoasa(config)# class-map IMPORTANT-FLOWS-MAP
ciscoasa(config)# match access-list IMPORTANT-FLOWS
ciscoasa(config-cmap)# policy-map INSIDE_POLICY
ciscoasa(config-pmap)# class IMPORTANT-FLOWS-MAP
ciscoasa(config-pmap-c)# cluster flow-mobility lisp
```

Step 4 Enter cluster group configuration mode, and enable flow mobility for the cluster:

cluster group *name*

flow-mobility lisp

This on/off toggle lets you easily enable or disable flow mobility.

Examples

The following example:

- Limits EIDs to those on the 10.10.10.0/24 network
- Inspects LISP traffic (UDP 4342) between a LISP router at 192.168.50.89 (on inside) and an ITR or ETR router (on another ASA interface) at 192.168.10.8
- Enables flow mobility for all inside traffic going to a server on 10.10.10.0/24 using HTTPS.
- Enables flow mobility for the cluster.

```
access-list TRACKED_EID_LISP extended permit ip any 10.10.10.0 255.255.255.0
policy-map type inspect lisp LISP_EID_INSPECT
   parameters
      allowed-eid access-list TRACKED_EID_LISP
      validate-key MadMaxShinyandChrome
!
access-list LISP_ACL extended permit udp host 192.168.50.89 host 192.168.10.8 eq 4342
class-map LISP_CLASS
   match access-list LISP_ACL
policy-map INSIDE_POLICY
   class LISP_CLASS
      inspect lisp LISP_EID_INSPECT
service-policy INSIDE_POLICY interface inside
!
access-list IMPORTANT-FLOWS extended permit tcp any 10.10.10.0 255.255.255.0 eq https
class-map IMPORTANT-FLOWS-MAP
   match access-list IMPORTANT-FLOWS
policy-map INSIDE_POLICY
   class IMPORTANT-FLOWS-MAP
      cluster flow-mobility lisp
!
cluster group cluster1
   flow-mobility lisp
```

Manage Cluster Nodes

After you deploy the cluster, you can change the configuration and manage cluster nodes.

Become an Inactive Node

To become an inactive member of the cluster, disable clustering on the node while leaving the clustering configuration intact.



Note When an ASA becomes inactive (either manually or through a health check failure), all data interfaces are shut down; only the management-only interface can send and receive traffic. To resume traffic flow, re-enable clustering; or you can remove the node altogether from the cluster. The management interface remains up using the IP address the node received from the cluster IP pool. However if you reload, and the node is still inactive in the cluster (for example, you saved the configuration with clustering disabled), then the management interface is disabled. You must use the console port for any further configuration.

Procedure

Step 1 Enter cluster configuration mode:

cluster group *name*

Example:

```
ciscoasa(config)# cluster group pod1
```

Step 2 Disable clustering:

no enable

If this node was the control node, a new control election takes place, and a different member becomes the control node.

The cluster configuration is maintained, so that you can enable clustering again later.

Deactivate a Data Node from the Control Node

To deactivate a member other than the node you are logged into, perform the following steps.

Note When an ASA becomes inactive, all data interfaces are shut down; only the management-only interface can send and receive traffic. To resume traffic flow, re-enable clustering. The management interface remains up using the IP address the node received from the cluster IP pool. However if you reload, and the node is still inactive in the cluster (for example, if you saved the configuration with clustering disabled), the management interface is disabled. You must use the console port for any further configuration.

Procedure

Remove the node from the cluster.

cluster remove unit *node_name*

The bootstrap configuration remains intact, as well as the last configuration synched from the control node, so that you can later re-add the node without losing your configuration. If you enter this command on a data node to remove the control node, a new control node is elected.

To view member names, enter **cluster remove unit ?**, or enter the **show cluster info** command.

Example:

```
ciscoasa(config)# cluster remove unit ?

Current active units in the cluster:
asa2

ciscoasa(config)# cluster remove unit asa2
WARNING: Clustering will be disabled on unit asa2. To bring it back
to the cluster please logon to that unit and re-enable clustering
```

Rejoin the Cluster

If a node was removed from the cluster, for example for a failed interface or if you manually deactivated a member, you must manually rejoin the cluster.

Procedure

Step 1 At the console, enter cluster configuration mode:

cluster group *name*

Example:

```
ciscoasa(config)# cluster group pod1
```

Step 2 Enable clustering.

enable

Leave the Cluster

If you want to leave the cluster altogether, you need to remove the entire cluster bootstrap configuration. Because the current configuration on each node is the same (synced from the active unit), leaving the cluster also means either restoring a pre-clustering configuration from backup, or clearing your configuration and starting over to avoid IP address conflicts.

Procedure

Step 1 For a data node, disable clustering:

cluster group *cluster_name*
no enable

Example:

```
ciscoasa(config)# cluster group cluster1
ciscoasa(cfg-cluster)# no enable
```

You cannot make configuration changes while clustering is enabled on a data node.

Step 2 Clear the cluster configuration:

clear configure cluster

The ASA shuts down all interfaces including the management interface and cluster control link.

Step 3 Disable cluster interface mode:

no cluster interface-mode

The mode is not stored in the configuration and must be reset manually.

Step 4 If you have a backup configuration, copy the backup configuration to the running configuration:

copy *backup_cfg* **running-config**

Example:

```
ciscoasa(config)# copy backup_cluster.cfg running-config

Source filename [backup_cluster.cfg]?

Destination filename [running-config]?
ciscoasa(config)#
```

Step 5 Save the configuration to startup:

write memory

Step 6 If you do not have a backup configuration, reconfigure management access. Be sure to change the interface IP addresses, and restore the correct hostname, for example.

Change the Control Node



Caution The best method to change the control node is to disable clustering on the control node, wait for a new control election, and then re-enable clustering. If you must specify the exact node you want to become the control node, use the procedure in this section. Note, however, that for centralized features, if you force a control node change using this procedure, then all connections are dropped, and you have to re-establish the connections on the new control node.

To change the control node, perform the following steps.

Procedure

Set a new node as the control node:

cluster master unit*node_name*

Example:

```
ciscoasa(config)# cluster master unit asa2
```

You will need to reconnect to the Main cluster IP address.

To view member names, enter **cluster master unit ?** (to see all names except the current node), or enter the **show cluster info** command.

Execute a Command Cluster-Wide

To send a command to all nodes in the cluster, or to a specific node, perform the following steps. Sending a **show** command to all nodes collects all output and displays it on the console of the current node. Other commands, such as **capture** and **copy**, can also take advantage of cluster-wide execution.

Procedure

Send a command to all nodes, or if you specify the node name, a specific node:

cluster exec [**unit** *node_name*] *command*

Example:

```
ciscoasa# cluster exec show xlate
```

To view node names, enter **cluster exec unit ?** (to see all names except the current node), or enter the **show cluster info** command.

Examples

To copy the same capture file from all nodes in the cluster at the same time to a TFTP server, enter the following command on the control node:

```
ciscoasa# cluster exec copy /pcap capture: tftp://10.1.1.56/capture1.pcap
```

Multiple PCAP files, one from each node, are copied to the TFTP server. The destination capture file name is automatically attached with the node name, such as capture1_asa1.pcap, capture1_asa2.pcap, and so on. In this example, asa1 and asa2 are cluster node names.

Monitoring the ASAv Cluster

You can monitor and troubleshoot cluster status and connections.

Monitoring Cluster Status

See the following commands for monitoring cluster status:

- **show cluster info** [**health** [**details**]]

  With no keywords, the **show cluster info** command shows the status of all members of the cluster.

  The **show cluster info health** command shows the current health of interfaces, nodes, and the cluster overall. The **details** keyword shows the number heartbeat message failures.

  See the following output for the **show cluster info** command:

```
ciscoasa# show cluster info
Cluster stbu: On
  This is "C" in state SLAVE
      ID        : 0
      Site ID   : 1
            Version   : 9.4(1)
      Serial No.: P3000000025
      CCL IP    : 10.0.0.3
      CCL MAC   : 000b.fcf8.c192
      Last join : 17:08:59 UTC Sep 26 2011
      Last leave: N/A
Other members in the cluster:
  Unit "D" in state SLAVE
      ID        : 1
      Site ID   : 1
            Version   : 9.4(1)
      Serial No.: P3000000001
```

```
    CCL IP    : 10.0.0.4
    CCL MAC   : 000b.fcf8.c162
    Last join : 19:13:11 UTC Sep 23 2011
    Last leave: N/A
Unit "A" in state MASTER
    ID        : 2
    Site ID   : 2
          Version   : 9.4(1)
    Serial No.: JAB0815R0JY
    CCL IP    : 10.0.0.1
    CCL MAC   : 000f.f775.541e
    Last join : 19:13:20 UTC Sep 23 2011
    Last leave: N/A
Unit "B" in state SLAVE
    ID        : 3
    Site ID   : 2
          Version   : 9.4(1)
    Serial No.: P3000000191
    CCL IP    : 10.0.0.2
    CCL MAC   : 000b.fcf8.c61e
    Last join : 19:13:50 UTC Sep 23 2011
    Last leave: 19:13:36 UTC Sep 23 2011
```

- **show cluster info auto-join**

  Shows whether the cluster node will automatically rejoin the cluster after a time delay and if the failure conditions (such as waiting for the license, chassis health check failure, and so on) are cleared. If the node is permanently disabled, or if the node is already in the cluster, then this command will not show any output.

  See the following outputs for the **show cluster info auto-join** command:

```
ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster in 253 seconds.
Quit reason: Received control message DISABLE

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Master has application down that slave has up.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Chassis-blade health check failed.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Service chain application became down.

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit will try to join cluster when quit reason is cleared.
Quit reason: Unit is kicked out from cluster because of Application health check failure.


ciscoasa(cfg-cluster)# show cluster info auto-join
Unit join is pending (waiting for the smart license entitlement: ent1)

ciscoasa(cfg-cluster)# show cluster info auto-join
Unit join is pending (waiting for the smart license export control flag)
```

- **show cluster info transport** {**asp** | **cp** [**detail**]}

Shows transport related statistics for the following:

- **asp** —Data plane transport statistics.
- **cp** —Control plane transport statistics.

  If you enter the **detail** keyword, you can view cluster reliable transport protocol usage so you can identify packet drop issues when the buffer is full in the control plane. See the following output for the **show cluster info transport cp detail** command:

```
ciscoasa# show cluster info transport cp detail
Member ID to name mapping:
  0 - unit-1-1   2 - unit-4-1   3 - unit-2-1

Legend:
  U    - unreliable messages
  UE   - unreliable messages error
  SN   - sequence number
  ESN  - expecting sequence number
  R    - reliable messages
  RE   - reliable messages error
  RDC  - reliable message deliveries confirmed
  RA   - reliable ack packets received
  RFR  - reliable fast retransmits
  RTR  - reliable timer-based retransmits
  RDP  - reliable message dropped
  RDPR - reliable message drops reported
  RI   - reliable message with old sequence number
  RO   - reliable message with out of order sequence number
  ROW  - reliable message with out of window sequence number
  ROB  - out of order reliable messages buffered
  RAS  - reliable ack packets sent

This unit as a sender
--------------------------
        all       0         2         3
  U     123301    3867966   3230662   3850381
  UE    0         0         0         0
  SN    1656a4ce  acb26fe   5f839f76  7b680831
  R     733840    1042168   852285    867311
  RE    0         0         0         0
  RDC   699789    934969    740874    756490
  RA    385525    281198    204021    205384
  RFR   27626     56397     0         0
  RTR   34051     107199    111411    110821
  RDP   0         0         0         0
  RDPR  0         0         0         0

This unit as a receiver of broadcast messages
-----------------------------------------------
        0         2         3
  U     111847    121862    120029
  R     7503      665700    749288
  ESN   5d75b4b3  6d81d23   365ddd50
  RI    630       34278     40291
  RO    0         582       850
  ROW   0         566       850
  ROB   0         16        0
  RAS   1571      123289    142256

This unit as a receiver of unicast messages
-----------------------------------------------
        0         2         3
```

```
  U     1          3308122   4370233
  R     513846     879979    1009492
  ESN   4458903a   6d841a84  7b4e7fa7
  RI    66024      108924    102114
  RO    0          0         0
  ROW   0          0         0
  ROB   0          0         0
  RAS   130258     218924    228303

Gated Tx Buffered Message Statistics
------------------------------------
    current sequence number: 0

    total:                   0
    current:                 0
    high watermark:          0

    delivered:               0
    deliver failures:        0

    buffer full drops:       0
    message truncate drops:  0

    gate close ref count:    0

    num of supported clients:45

MRT Tx of broadcast messages
============================
Message high watermark: 3%
  Total messages buffered at high watermark: 5677
  [Per-client message usage at high watermark]
  -------------------------------------------------------------
  Client name                          Total messages  Percentage
  Cluster Redirect Client                        4153         73%
  Route Cluster Client                            419          7%
  RRI Cluster Client                             1105         19%

Current MRT buffer usage: 0%
  Total messages buffered in real-time: 1
  [Per-client message usage in real-time]
  Legend:
        F - MRT messages sending when buffer is full
        L - MRT messages sending when cluster node leave
        R - MRT messages sending in Rx thread
  ------------------------------------------------------------------------------
  Client name                          Total messages  Percentage    F   L   R
  VPN Clustering HA Client                          1        100%    0   0   0

MRT Tx of unitcast messages(to member_id:0)
===========================================
Message high watermark: 31%
  Total messages buffered at high watermark: 4059
  [Per-client message usage at high watermark]
  -------------------------------------------------------------
  Client name                          Total messages  Percentage
  Cluster Redirect Client                        3731         91%
  RRI Cluster Client                              328          8%

Current MRT buffer usage: 29%
  Total messages buffered in real-time: 3924
  [Per-client message usage in real-time]
  Legend:
        F - MRT messages sending when buffer is full
```

```
        L - MRT messages sending when cluster node leave
        R - MRT messages sending in Rx thread
  ------------------------------------------------------------------------------
  Client name                          Total messages  Percentage    F   L   R
  Cluster Redirect Client                        3607         91%    0   0   0
  RRI Cluster Client                              317          8%    0   0   0

MRT Tx of unitcast messages(to member_id:2)
===========================================
Message high watermark: 14%
  Total messages buffered at high watermark: 578
  [Per-client message usage at high watermark]
  ---------------------------------------------------------------
  Client name                          Total messages  Percentage
  VPN Clustering HA Client                       578        100%

Current MRT buffer usage: 0%
  Total messages buffered in real-time: 0

MRT Tx of unitcast messages(to member_id:3)
===========================================
Message high watermark: 12%
  Total messages buffered at high watermark: 573
  [Per-client message usage at high watermark]
  ---------------------------------------------------------------
  Client name                          Total messages  Percentage
  VPN Clustering HA Client                       572         99%
  Cluster VPN Unique ID Client                     1          0%

Current MRT buffer usage: 0%
  Total messages buffered in real-time: 0
```

- **show cluster history**

  Shows the cluster history, as well as error messages about why a cluster node failed to join or why a node left the cluster.

Capturing Packets Cluster-Wide

See the following command for capturing packets in a cluster:

cluster exec capture

To support cluster-wide troubleshooting, you can enable capture of cluster-specific traffic on the control node using the **cluster exec capture** command, which is then automatically enabled on all of the data nodes in the cluster.

Monitoring Cluster Resources

See the following command for monitoring cluster resources:

show cluster {**cpu** | **memory** | **resource**} [*options*]

Displays aggregated data for the entire cluster. The *options* available depends on the data type.

Monitoring Cluster Traffic

See the following commands for monitoring cluster traffic:

- **show conn [detail], cluster exec show conn**

  The **show conn** command shows whether a flow is a director, backup, or forwarder flow. Use the **cluster exec show conn** command on any node to view all connections. This command can show how traffic for a single flow arrives at different ASAs in the cluster. The throughput of the cluster is dependent on the efficiency and configuration of load balancing. This command provides an easy way to view how traffic for a connection is flowing through the cluster, and can help you understand how a load balancer might affect the performance of a flow.

  The **show conn detail** command also shows which flows are subject to flow mobility.

  The following is sample output for the **show conn detail** command:

```
ciscoasa/ASA2/slave# show conn detail
12 in use, 13 most used
Cluster stub connections: 0 in use, 46 most used
Flags: A - awaiting inside ACK to SYN, a - awaiting outside ACK to SYN,
       B - initial SYN from outside, b - TCP state-bypass or nailed,
       C - CTIQBE media, c - cluster centralized,
       D - DNS, d - dump, E - outside back connection, e - semi-distributed,
       F - outside FIN, f - inside FIN,
       G - group, g - MGCP, H - H.323, h - H.225.0, I - inbound data,
       i - incomplete, J - GTP, j - GTP data, K - GTP t3-response
       k - Skinny media, L - LISP triggered flow owner mobility,
       M - SMTP data, m - SIP media, n - GUP
       O - outbound data, o - offloaded,
       P - inside back connection,
       Q - Diameter, q - SQL*Net data,
       R - outside acknowledged FIN,
       R - UDP SUNRPC, r - inside acknowledged FIN, S - awaiting inside SYN,
       s - awaiting outside SYN, T - SIP, t - SIP transient, U - up,
       V - VPN orphan, W - WAAS,
       w - secondary domain backup,
       X - inspected by service module,
       x - per session, Y - director stub flow, y - backup stub flow,
       Z - Scansafe redirection, z - forwarding stub flow
ESP outside: 10.1.227.1/53744 NP Identity Ifc: 10.1.226.1/30604, , flags c, idle 0s,
uptime
1m21s, timeout 30s, bytes 7544, cluster sent/rcvd bytes 0/0, owners (0,255) Traffic
received
at interface outside Locally received: 7544 (93 byte/s) Traffic received at interface
NP
Identity Ifc Locally received: 0 (0 byte/s) UDP outside: 10.1.227.1/500 NP Identity
Ifc:
10.1.226.1/500, flags -c, idle 1m22s, uptime 1m22s, timeout 2m0s, bytes 1580, cluster
sent/rcvd bytes 0/0, cluster sent/rcvd total bytes 0/0, owners (0,255) Traffic received
 at
interface outside Locally received: 864 (10 byte/s) Traffic received at interface NP
Identity
Ifc Locally received: 716 (8 byte/s)
```

  To troubleshoot the connection flow, first see connections on all nodes by entering the **cluster exec show conn** command on any node. Look for flows that have the following flags: director (Y), backup (y), and forwarder (z). The following example shows an SSH connection from 172.18.124.187:22 to 192.168.103.131:44727 on all three ASAs; ASA 1 has the z flag showing it is a forwarder for the connection, ASA3 has the Y flag showing it is the director for the connection, and ASA2 has no special

flags showing it is the owner. In the outbound direction, the packets for this connection enter the inside interface on ASA2 and exit the outside interface. In the inbound direction, the packets for this connection enter the outside interface on ASA 1 and ASA3, are forwarded over the cluster control link to ASA2, and then exit the inside interface on ASA2.

```
ciscoasa/ASA1/master# cluster exec show conn
ASA1(LOCAL):**********************************************************
18 in use, 22 most used
Cluster stub connections: 0 in use, 5 most used
TCP outside  172.18.124.187:22 inside  192.168.103.131:44727, idle 0:00:00, bytes
37240828, flags z


ASA2:**********************************************************************
12 in use, 13 most used
Cluster stub connections: 0 in use, 46 most used
TCP outside  172.18.124.187:22 inside  192.168.103.131:44727, idle 0:00:00, bytes
37240828, flags UIO


ASA3:**********************************************************************
10 in use, 12 most used
Cluster stub connections: 2 in use, 29 most used
TCP outside  172.18.124.187:22 inside  192.168.103.131:44727, idle 0:00:03, bytes 0,
flags  Y
```

- **show cluster info** [**conn-distribution** | **packet-distribution** | **loadbalance** | **flow-mobility counters**]

  The **show cluster info conn-distribution** and **show cluster info packet-distribution** commands show traffic distribution across all cluster nodes. These commands can help you to evaluate and adjust the external load balancer.

  The **show cluster info loadbalance** command shows connection rebalance statistics.

  The **show cluster info flow-mobility counters** command shows EID movement and flow owner movement information. See the following output for **show cluster info flow-mobility counters**:

```
ciscoasa# show cluster info flow-mobility counters
EID movement notification received  : 4
EID movement notification processed : 4
Flow owner moving requested         : 2
```

- **show cluster info load-monitor** [**details**]

  The **show cluster info load-monitor** command shows the traffic load for cluster members for the last interval and also the average over total number of intervals configured (30 by default). Use the **details** keyword to view the value for each measure at each interval.

```
ciscoasa(cfg-cluster)# show cluster info load-monitor
ID  Unit Name
0  B
1  A_1
Information from all units with 20 second interval:
Unit     Connections    Buffer Drops   Memory Used   CPU Used
Average from last 1 interval:
   0           0           0           14             25
   1           0           0           16             20
Average from last 30 interval:
   0           0           0           12             28
```

```
   1           0           0           13              27

ciscoasa(cfg-cluster)# show cluster info load-monitor details

ID  Unit Name

 0  B

 1  A_1

Information from all units with 20 second interval


Connection count captured over 30 intervals:

Unit ID  0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

Unit ID  1

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0


Buffer drops captured over 30 intervals:

Unit ID  0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

         0          0          0          0          0          0

Unit ID  1

         0          0          0          0          0          0

         0          0          0          0          0          0
```

```
          0          0          0          0          0          0
          0          0          0          0          0          0
          0          0          0          0          0          0

Memory usage(%) captured over 30 intervals:
Unit ID  0
         25         25         30         30         30         35
         25         25         35         30         30         30
         25         25         30         25         25         35
         30         30         30         25         25         25
         25         20         30         30         30         30
Unit ID  1
         30         25         35         25         30         30
         25         25         35         25         30         35
         30         30         35         30         30         30
         25         20         30         25         25         30
         20         30         35         30         30         35

CPU usage(%) captured over 30 intervals:
Unit ID  0
         25         25         30         30         30         35
         25         25         35         30         30         30
         25         25         30         25         25         35
         30         30         30         25         25         25
         25         20         30         30         30         30
Unit ID  1
         30         25         35         25         30         30
         25         25         35         25         30         35
         30         30         35         30         30         30
         25         20         30         25         25         30
         20         30         35         30         30         35
```

- **show cluster** {**access-list** | **conn** | **traffic** | **user-identity** | **xlate**} [*options*]

Displays aggregated data for the entire cluster. The *options* available depends on the data type.

See the following output for the **show cluster access-list** command:

```
ciscoasa# show cluster access-list
hitcnt display order: cluster-wide aggregated result, unit-A, unit-B,  unit-C, unit-D
access-list cached ACL log flows: total 0, denied 0 (deny-flow-max 4096) alert-interval
 300
access-list 101; 122 elements; name hash: 0xe7d586b5
access-list 101 line 1 extended permit tcp 192.168.143.0 255.255.255.0 any eq www
(hitcnt=0, 0, 0, 0, 0) 0x207a2b7d
access-list 101 line 2 extended permit tcp any 192.168.143.0 255.255.255.0 (hitcnt=0,
0, 0, 0, 0) 0xfe4f4947
access-list  101 line 3 extended permit tcp host 192.168.1.183 host 192.168.43.238
(hitcnt=1, 0, 0, 0, 1) 0x7b521307
access-list 101 line 4 extended permit tcp host 192.168.1.116 host 192.168.43.238
(hitcnt=0, 0, 0, 0, 0) 0x5795c069
access-list 101 line 5 extended permit tcp host 192.168.1.177 host 192.168.43.238
(hitcnt=1, 0, 0, 1, 0) 0x51bde7ee
access list 101 line 6 extended permit tcp host 192.168.1.177 host 192.168.43.13
(hitcnt=0, 0, 0, 0, 0) 0x1e68697c
access-list 101 line 7 extended permit tcp host 192.168.1.177 host 192.168.43.132
(hitcnt=2, 0, 0, 1, 1) 0xc1ce5c49
access-list 101 line 8 extended permit tcp host 192.168.1.177 host 192.168.43.192
(hitcnt=3, 0, 1, 1, 1) 0xb6f59512
access-list 101 line 9 extended permit tcp host 192.168.1.177 host 192.168.43.44
(hitcnt=0, 0, 0, 0, 0) 0xdc104200
access-list 101 line 10 extended permit tcp host 192.168.1.112 host 192.168.43.44
(hitcnt=429, 109, 107, 109, 104)
0xce4f281d
access-list 101 line 11 extended permit tcp host 192.168.1.170 host 192.168.43.238
(hitcnt=3, 1, 0, 0, 2) 0x4143a818
access-list 101 line 12 extended permit tcp host 192.168.1.170 host 192.168.43.169
(hitcnt=2, 0, 1, 0, 1) 0xb18dfea4
access-list 101 line 13 extended permit tcp host 192.168.1.170 host 192.168.43.229
(hitcnt=1, 1, 0, 0, 0) 0x21557d71
access-list 101 line 14 extended permit tcp host 192.168.1.170 host 192.168.43.106
(hitcnt=0, 0, 0, 0, 0) 0x7316e016
access-list 101 line 15 extended permit tcp host 192.168.1.170 host 192.168.43.196
(hitcnt=0, 0, 0, 0, 0) 0x013fd5b8
access-list 101 line 16 extended permit tcp host 192.168.1.170 host 192.168.43.75
(hitcnt=0, 0, 0, 0, 0) 0x2c7dba0d
```

To display the aggregated count of in-use connections for all nodes, enter:

```
ciscoasa# show cluster conn count
Usage Summary In Cluster:*****************************************
  200 in use (cluster-wide aggregated)
     cl2(LOCAL):************************************************************
  100 in use, 100 most used

  cl1:************************************************************************
  100 in use, 100 most used
```

- **show asp cluster counter**

  This command is useful for datapath troubleshooting.

Monitoring Cluster Routing

See the following commands for cluster routing:

- **show route cluster**
- **debug route cluster**

  Shows cluster information for routing.

- **show lisp eid**

  Shows the ASA EID table showing EIDs and site IDs.

  See the following output from the **cluster exec show lisp eid** command.

```
ciscoasa# cluster exec show lisp eid
L1(LOCAL):*********************************************************
     LISP EID       Site ID
    33.44.33.105        2
    33.44.33.201        2
    11.22.11.1             4
    11.22.11.2             4
L2:*********************************************************************
     LISP EID       Site ID
    33.44.33.105        2
    33.44.33.201        2
    11.22.11.1   4
    11.22.11.2  4
```

- **show asp table classify domain inspect-lisp**

  This command is useful for troubleshooting.

Configuring Logging for Clustering

See the following command for configuring logging for clustering:

logging device-id

Each node in the cluster generates syslog messages independently. You can use the **logging device-id** command to generate syslog messages with identical or different device IDs to make messages appear to come from the same or different nodes in the cluster.

Monitoring Cluster Interfaces

See the following commands for monitoring cluster interfaces:

- **show cluster interface-mode**

  Shows the cluster interface mode.

Debugging Clustering

See the following commands for debugging clustering:

- **debug cluster [ccp | datapath | fsm | general | hc | license | rpc | transport]**

  Shows debug messages for clustering.

- **debug cluster flow-mobility**

  Shows events related to clustering flow mobility.

- **debug lisp eid-notify-intercept**

  Shows events when the eid-notify message is intercepted.

- **show cluster info trace**

  The **show cluster info trace** command shows the debug information for further troubleshooting.

  See the following output for the **show cluster info trace** command:

```
ciscoasa# show cluster info trace
 Feb 02 14:19:47.456 [DBUG]Receive CCP message: CCP_MSG_LOAD_BALANCE
 Feb 02 14:19:47.456 [DBUG]Receive CCP message: CCP_MSG_LOAD_BALANCE
 Feb 02 14:19:47.456 [DBUG]Send CCP message to all: CCP_MSG_KEEPALIVE from 80-1 at
MASTER
```

  For example, if you see the following messages showing that two nodes with the same **local-unit** name are acting as the control node, it could mean that either two nodes have the same **local-unit** name (check your configuration), or a node is receiving its own broadcast messages (check your network).

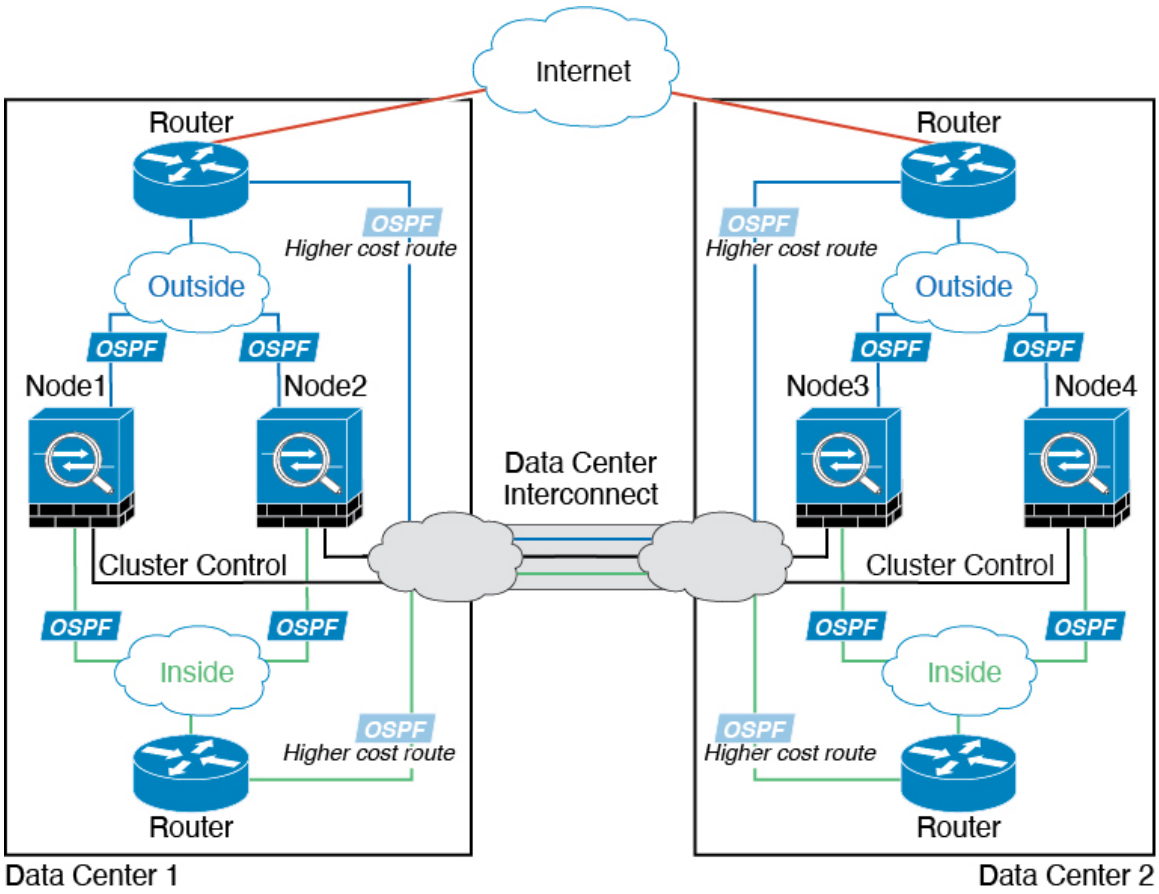
```
ciscoasa# show cluster info trace
 May 23 07:27:23.113 [CRIT]Received datapath event 'multi masters' with parameter 1.
 May 23 07:27:23.113 [CRIT]Found both unit-9-1 and unit-9-1 as master units. Master
role retained by unit-9-1, unit-9-1 will leave then join as a Slave
 May 23 07:27:23.113 [DBUG]Send event (DISABLE, RESTART | INTERNAL-EVENT, 5000 msecs,
Detected another Master, leave and re-join as Slave) to FSM. Current state MASTER
 May 23 07:27:23.113 [INFO]State machine changed from state MASTER to DISABLED
```

Examples for ASAv Clustering

These examples include all cluster-related ASA configuration for typical deployments.

Individual Interface Routed Mode North-South Inter-Site Example

The following example shows 2 ASA cluster nodes at each of 2 data centers placed between inside and outside routers (North-South insertion). The cluster nodes are connected by the cluster control link over the DCI. The inside and outside routers at each data center use OSPF and PBR or ECMP to load balance the traffic between cluster members. By assigning a higher cost route across the DCI, traffic stays within each data center unless all ASA cluster nodes at a given site go down. In the event of a failure of all cluster nodes at one site, traffic goes from each router over the DCI to the ASA cluster nodes at the other site.



Reference for Clustering

This section includes more information about how clustering operates.

ASA Features and Clustering

Some ASA features are not supported with ASA clustering, and some are only supported on the control node. Other features might have caveats for proper usage.

Unsupported Features with Clustering

These features cannot be configured with clustering enabled, and the commands will be rejected.

- Unified Communication features that rely on TLS Proxy
- Remote access VPN (SSL VPN and IPsec VPN)
- Virtual Tunnel Interfaces (VTIs)
- The following application inspections:
  - CTIQBE

- H323, H225, and RAS
- IPsec passthrough
- MGCP
- MMP
- RTSP
- SCCP (Skinny)
- WAAS
- WCCP

- Botnet Traffic Filter
- Auto Update Server
- DHCP client, server, and proxy. DHCP relay is supported.
- VPN load balancing
- Failover
- Integrated Routing and Bridging
- FIPS mode

Centralized Features for Clustering

These features are only supported on the control node, and are not scaled for the cluster.



Note Traffic for centralized features is forwarded from member nodes to the control node over the cluster control link.

If you use the rebalancing feature, traffic for centralized features may be rebalanced to non-control nodes before the traffic is classified as a centralized feature; if this occurs, the traffic is then sent back to the control node.

For centralized features, if the control node fails, all connections are dropped, and you have to re-establish the connections on the new control node.

- These application inspections are supported:
  - DCERPC and ESMTP
  - IM
  - NetBIOS and PPTP
  - RADIUS
  - RSH
  - SNMP

  - SQLNET and SUNRPC
  - TFTP
  - XDMCP
- Static route monitoring
- Authentication and Authorization for network access. Accounting is decentralized.
- Filtering Services
- Site-to-site VPN
- Multicast routing

Features Applied to Individual Nodes

These features are applied to each ASA node, instead of the cluster as a whole or to the control node.

- QoS—The QoS policy is synced across the cluster as part of configuration replication. However, the policy is enforced on each node independently. For example, if you configure policing on output, then the conform rate and conform burst values are enforced on traffic exiting a particular ASA. In a cluster with 3 nodes and with traffic evenly distributed, the conform rate actually becomes 3 times the rate for the cluster.
- Threat detection—Threat detection works on each node independently; for example, the top statistics is node-specific. Port scanning detection, for example, does not work because scanning traffic will be load-balanced between all nodes, and one node will not see all traffic.
- Resource management—Resource management in multiple context mode is enforced separately on each node based on local usage.
- LISP traffic—LISP traffic on UDP port 4342 is inspected by each receiving node, but is not assigned a director. Each node adds to the EID table that is shared across the cluster, but the LISP traffic itself does not participate in cluster state sharing.

AAA for Network Access and Clustering

AAA for network access consists of three components: authentication, authorization, and accounting. Authentication and authorization are implemented as centralized features on the clustering control node with replication of the data structures to the cluster data nodes. If a control node is elected, the new control node will have all the information it needs to continue uninterrupted operation of the established authenticated users and their associated authorizations. Idle and absolute timeouts for user authentications are preserved when a control node change occurs.

Accounting is implemented as a distributed feature in a cluster. Accounting is done on a per-flow basis, so the cluster node owning a flow will send accounting start and stop messages to the AAA server when accounting is configured for a flow.

Connection Settings and Clustering

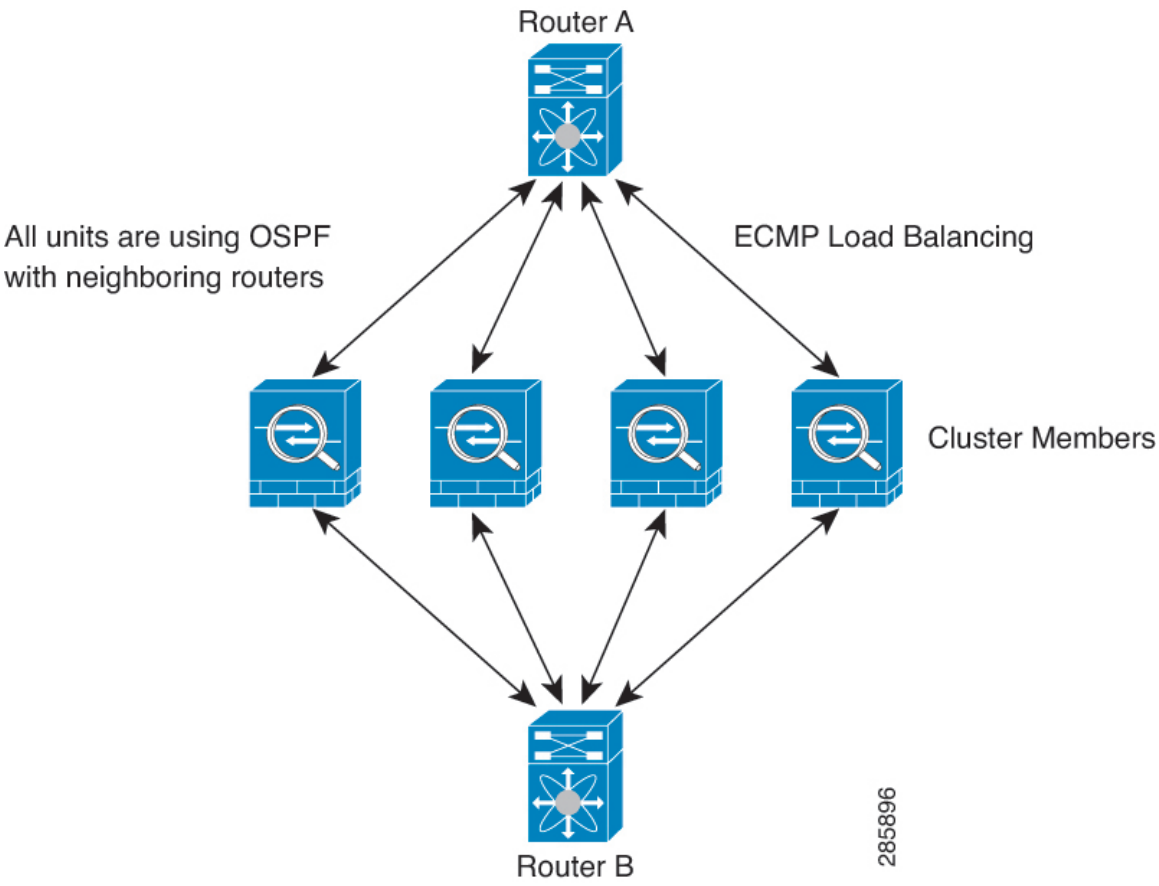
Connection limits are enforced cluster-wide (see the **set connection conn-max**, **set connection embryonic-conn-max**, **set connection per-client-embryonic-max**, and **set connection per-client-max** commands). Each node has an estimate of the cluster-wide counter values based on broadcast messages. Due

to efficiency considerations, the configured connection limit across the cluster might not be enforced exactly at the limit number. Each node may overestimate or underestimate the cluster-wide counter value at any given time. However, the information will get updated over time in a load-balanced cluster.

Dynamic Routing and Clustering

In Individual interface mode, each node runs the routing protocol as a standalone router, and routes are learned by each node independently.

Figure 50: Dynamic Routing in Individual Interface Mode



In the above diagram, Router A learns that there are 4 equal-cost paths to Router B, each through a node. ECMP is used to load balance traffic between the 4 paths. Each node picks a different router ID when talking to external routers.

You must configure a cluster pool for the router ID so that each node has a separate router ID.

EIGRP does not form neighbor relationships with cluster peers in individual interface mode.

Note If the cluster has multiple adjacencies to the same router for redundancy purposes, asymmetric routing can lead to unacceptable traffic loss. To avoid asymmetric routing, group all of these node interfaces into the same traffic zone. See Configure a Traffic Zone, on page 726.

FTP and Clustering

- If FTP data channel and control channel flows are owned by different cluster members, then the data channel owner will periodically send idle timeout updates to the control channel owner and update the idle timeout value. However, if the control flow owner is reloaded, and the control flow is re-hosted, the parent/child flow relationship will not longer be maintained; the control flow idle timeout will not be updated.
- If you use AAA for FTP access, then the control channel flow is centralized on the control node.

ICMP Inspection and Clustering

The flow of ICMP and ICMP error packets through the cluster varies depending on whether ICMP/ICMP error inspection is enabled. Without ICMP inspection, ICMP is a one-direction flow, and there is no director flow support. With ICMP inspection, the ICMP flow becomes two-directional and is backed up by a director/backup flow. One difference for an inspected ICMP flow is in the director handling of a forwarded packet: the director will forward the ICMP echo reply packet to the flow owner instead of returning the packet to the forwarder.

Multicast Routing and Clustering

In Individual interface mode, units do not act independently with multicast. All data and routing packets are processed and forwarded by the control unit, thus avoiding packet replication.

NAT and Clustering

NAT can affect the overall throughput of the cluster. Inbound and outbound NAT packets can be sent to different ASAs in the cluster, because the load balancing algorithm relies on IP addresses and ports, and NAT causes inbound and outbound packets to have different IP addresses and/or ports. When a packet arrives at the ASA that is not the NAT owner, it is forwarded over the cluster control link to the owner, causing large amounts of traffic on the cluster control link. Note that the receiving node does not create a forwarding flow to the owner, because the NAT owner may not end up creating a connection for the packet depending on the results of security and policy checks.

If you still want to use NAT in clustering, then consider the following guidelines:

- No Proxy ARP—For Individual interfaces, a proxy ARP reply is never sent for mapped addresses. This prevents the adjacent router from maintaining a peer relationship with an ASA that may no longer be in the cluster. The upstream router needs a static route or PBR with Object Tracking for the mapped addresses that points to the Main cluster IP address. This is not an issue for a Spanned EtherChannel, because there is only one IP address associated with the cluster interface.
- No interface PAT on an Individual interface—Interface PAT is not supported for Individual interfaces.
- PAT with Port Block Allocation—See the following guidelines for this feature:
  - Maximum-per-host limit is not a cluster-wide limit, and is enforced on each node individually. Thus, in a 3-node cluster with the maximum-per-host limit configured as 1, if the traffic from a host is load-balanced across all 3 nodes, then it can get allocated 3 blocks with 1 in each node.
  - Port blocks created on the backup node from the backup pools are not accounted for when enforcing the maximum-per-host limit.

  - On-the-fly PAT rule modifications, where the PAT pool is modified with a completely new range of IP addresses, will result in xlate backup creation failures for the xlate backup requests that were still in transit while the new pool became effective. This behavior is not specific to the port block allocation feature, and is a transient PAT pool issue seen only in cluster deployments where the pool is distributed and traffic is load-balanced across the cluster nodes.
  - When operating in a cluster, you cannot simply change the block allocation size. The new size is effective only after you reload each device in the cluster. To avoid having to reload each device, we recommend that you delete all block allocation rules and clear all xlates related to those rules. You can then change the block size and recreate the block allocation rules.

- NAT pool address distribution for dynamic PAT—When you configure a PAT pool, the cluster divides each IP address in the pool into port blocks. By default, each block is 512 ports, but if you configure port block allocation rules, your block setting is used instead. These blocks are distributed evenly among the nodes in the cluster, so that each node has one or more blocks for each IP address in the PAT pool. Thus, you could have as few as one IP address in a PAT pool for a cluster, if that is sufficient for the number of PAT'ed connections you expect. Port blocks cover the 1024-65535 port range, unless you configure the option to include the reserved ports, 1-1023, on the PAT pool NAT rule.
- Reusing a PAT pool in multiple rules—To use the same PAT pool in multiple rules, you must be careful about the interface selection in the rules. You must either use specific interfaces in all rules, or "any" in all rules. You cannot mix specific interfaces and "any" across the rules, or the system might not be able to match return traffic to the right node in the cluster. Using unique PAT pools per rule is the most reliable option.
- No round-robin—Round-robin for a PAT pool is not supported with clustering.
- No extended PAT—Extended PAT is not supported with clustering.
- Dynamic NAT xlates managed by the control node—The control node maintains and replicates the xlate table to data nodes. When a data node receives a connection that requires dynamic NAT, and the xlate is not in the table, it requests the xlate from the control node. The data node owns the connection.
- Stale xlates—The xlate idle time on the connection owner does not get updated. Thus, the idle time might exceed the idle timeout. An idle timer value higher than the configured timeout with a refcnt of 0 is an indication of a stale xlate.
- Per-session PAT feature—Although not exclusive to clustering, the per-session PAT feature improves the scalability of PAT and, for clustering, allows each data node to own PAT connections; by contrast, multi-session PAT connections have to be forwarded to and owned by the control node. By default, all TCP traffic and UDP DNS traffic use a per-session PAT xlate, whereas ICMP and all other UDP traffic uses multi-session. You can configure per-session NAT rules to change these defaults for TCP and UDP, but you cannot configure per-session PAT for ICMP. For example, with an increasing use of Quic protocol over UDP/443 as a best performance alternative compared to HTTPS TLS over TCP/443, you should enable per-session PAT for UDP/443. For traffic that benefits from multi-session PAT, such as H.323, SIP, or Skinny, you can disable per-session PAT for the associated TCP ports (the UDP ports for those H.323 and SIP are already multi-session by default). For more information about per-session PAT, see the firewall configuration guide.
- No static PAT for the following inspections—
  - FTP
  - PPTP

- RSH
- SQLNET
- TFTP
- XDMCP
- SIP

- If you have an extremely large number of NAT rules, over ten thousand, you should enable the transactional commit model using the **asp rule-engine transactional-commit nat** command in the device CLI. Otherwise, the node might not be able to join the cluster.

SCTP and Clustering

An SCTP association can be created on any node (due to load balancing); its multi-homing connections must reside on the same node.

SIP Inspection and Clustering

A control flow can be created on any node (due to load balancing); its child data flows must reside on the same node.

TLS Proxy configuration is not supported.

SNMP and Clustering

An SNMP agent polls each individual ASA by its Local IP address. You cannot poll consolidated data for the cluster.

You should always use the Local address, and not the Main cluster IP address for SNMP polling. If the SNMP agent polls the Main cluster IP address, if a new control node is elected, the poll to the new control node will fail.

When using SNMPv3 with clustering, if you add a new cluster node after the initial cluster formation, then SNMPv3 users are not replicated to the new node.You must re-add them on the control node to force the users to replicate to the new node, or directly on the data node.

STUN and Clustering

STUN inspection is supported in failover and cluster modes, as pinholes are replicated. However, the transaction ID is not replicated among nodes. In the case where a node fails after receiving a STUN Request and another node received the STUN Response, the STUN Response will be dropped.

Syslog and NetFlow and Clustering

- Syslog—Each node in the cluster generates its own syslog messages. You can configure logging so that each node uses either the same or a different device ID in the syslog message header field. For example, the hostname configuration is replicated and shared by all nodes in the cluster. If you configure logging to use the hostname as the device ID, syslog messages generated by all nodes look as if they come from a single node. If you configure logging to use the local-node name that is assigned in the cluster bootstrap configuration as the device ID, syslog messages look as if they come from different nodes.

- NetFlow—Each node in the cluster generates its own NetFlow stream. The NetFlow collector can only treat each ASA as a separate NetFlow exporter.

Cisco TrustSec and Clustering

Only the control node learns security group tag (SGT) information. The control node then populates the SGT to data nodes, and data nodes can make a match decision for SGT based on the security policy.

VPN and Clustering

Site-to-site VPN is a centralized feature; only the control node supports VPN connections.



Note Remote access VPN is not supported with clustering.

VPN functionality is limited to the control node and does not take advantage of the cluster high availability capabilities. If the control node fails, all existing VPN connections are lost, and VPN users will see a disruption in service. When a new control node is elected, you must reestablish the VPN connections.

For connections to an Individual interface when using PBR or ECMP, you must always connect to the Main cluster IP address, not a Local address.

VPN-related keys and certificates are replicated to all nodes.

Performance scaling factor

When you combine multiple units into a cluster, you can expect the total cluster performance to be approximately 80% of the maximum combined throughput.

Throughput calculation

For example, if your model can handle approximately 10 Gbps of traffic when running alone, then for a cluster of 8 units, the maximum combined throughput will be approximately 80% of 80 Gbps (8 units x 10 Gbps): 64 Gbps.

Control Node Election

The key components involved in the control node election process are:

- Cluster nodes: Communicate over the cluster control link and participate in the election process
- Election requests: Broadcast every 3 seconds by nodes when clustering is enabled
- Priority values: Set between 1 and 100 where 1 is the highest priority
- Control node: The elected node that coordinates cluster operations

These stages describe how cluster nodes elect a control node:

1. When clustering is enabled for a node (or when it first starts up with clustering already enabled), the node broadcasts an election request every 3 seconds.

2. Any other nodes with a higher priority respond to the election request. The priority is set between 1 and 100, where 1 is the highest priority.
3. If after 45 seconds, a node does not receive a response from another node with a higher priority, then it becomes the control node.



Note If multiple nodes tie for the highest priority, the cluster node name and then the serial number is used to determine the control node.

4. If a node later joins the cluster with a higher priority, it does not automatically become the control node. The existing control node always remains as the control node unless it stops responding, at which point a new control node is elected.
5. In a "split brain" scenario when there are temporarily multiple control nodes, the node with highest priority retains the role while the other nodes return to data node roles.



Note You can manually force a node to become the control node. For centralized features, if you force a control node change, then all connections are dropped, and you have to re-establish the connections on the new control node.

High Availability Within the ASAv Cluster

The ASAv Clustering provides high availability by monitoring node and interface health and by replicating connection states between nodes.

Node health monitoring

Each node periodically sends a broadcast heartbeat packet over the cluster control link. If the control node does not receive any heartbeat packets or other packets from a data node within the configurable timeout period, then the control node removes the data node from the cluster. If the data nodes do not receive packets from the control node, then a new control node is elected from the remaining nodes.

Split brain scenario

If nodes cannot reach each other over the cluster control link because of a network failure and not because a node has actually failed, then the cluster may go into a "split brain" scenario where isolated data nodes will elect their own control nodes. For example, if a router fails between two cluster locations, then the original control node at location 1 will remove the location 2 data nodes from the cluster. Meanwhile, the nodes at location 2 will elect their own control node and form their own cluster. Note that asymmetric traffic may fail in this scenario. After the cluster control link is restored, then the control node that has the higher priority will keep the control node's role.

Interface Monitoring

Each node monitors the link status of all named hardware interfaces in use, and reports status changes to the control node.

When you enable health monitoring, all physical interfaces are monitored by default; you can optionally disable monitoring per interface. Only named interfaces can be monitored.

A node is removed from the cluster if its monitored interfaces fail. The amount of time before the ASA removes a member from the cluster depends on whether the node is an established member or is joining the cluster. The ASA does not monitor interfaces for the first 90 seconds that a node joins the cluster. Interface status changes during this time will not cause the ASA to be removed from the cluster. The node is removed after 500 ms, regardless of the node state.

Status After Failure

When a node in the cluster fails, the connections hosted by that node are seamlessly transferred to other nodes; state information for traffic flows is shared over the control node's cluster control link.

If the control node fails, then another member of the cluster with the highest priority (lowest number) becomes the control node.

The ASA automatically tries to rejoin the cluster, depending on the failure event.

Note When the ASA becomes inactive and fails to automatically rejoin the cluster, all data interfaces are shut down; only the management-only interface can send and receive traffic. The management interface remains up using the IP address the node received from the cluster IP pool. However if you reload, and the node is still inactive in the cluster, the management interface is disabled. You must use the console port for any further configuration.

Rejoining the Cluster

After a cluster node is removed from the cluster, how it can rejoin the cluster depends on why it was removed:

- Failed cluster control link when initially joining—After you resolve the problem with the cluster control link, you must manually rejoin the cluster by re-enabling clustering at the CLI by entering **cluster group** *name*, and then **enable**.
- Failed cluster control link after joining the cluster—The ASA automatically tries to rejoin every 5 minutes, indefinitely. This behavior is configurable.
- Failed data interface—The ASA automatically tries to rejoin at 5 minutes, then at 10 minutes, and finally at 20 minutes. If the join is not successful after 20 minutes, then the ASA disables clustering. After you resolve the problem with the data interface, you have to manually enable clustering at the CLI by entering **cluster group** *name*, and then **enable**. This behavior is configurable.
- Failed node—If the node was removed from the cluster because of a node health check failure, then rejoining the cluster depends on the source of the failure. For example, a temporary power failure means the node will rejoin the cluster when it starts up again as long as the cluster control link is up and clustering is still enabled with the **enable** command. The ASA attempts to rejoin the cluster every 5 seconds.
- Internal error—Internal failures include: application sync timeout; inconsistent application statuses; and so on. A node will attempt to rejoin the cluster automatically at the following intervals: 5 minutes, 10 minutes, and then 20 minutes. This behavior is configurable.

Data Path Connection State Replication

Every connection has one owner and at least one backup owner in the cluster. The backup owner does not take over the connection in the event of a failure; instead, it stores TCP/UDP state information, so that the connection can be seamlessly transferred to a new owner in case of a failure. The backup owner is usually also the director.

Some traffic requires state information above the TCP or UDP layer. See the following table for clustering support or lack of support for this kind of traffic.

Table 20: Features Replicated Across the Cluster

Traffic	State Support	Notes
Up time	Yes	Keeps track of the system up time.
ARP Table	Yes	—
MAC address table	Yes	—
User Identity	Yes	Includes AAA rules (uauth).
IPv6 Neighbor database	Yes	—
Dynamic routing	Yes	—
SNMP Engine ID	**No**	—
Distributed VPN (Site-to-Site) for Firepower 4100/9300	Yes	Backup session becomes the active session, then a new backup session is created.

How the ASAv Cluster Manages Connections

Connections can be load-balanced to multiple nodes of the cluster. Connection roles determine how connections are handled in both normal operation and in a high availability situation.

Connection Roles

See the following roles defined for each connection:

- Owner—Usually, the node that initially receives the connection. The owner maintains the TCP state and processes packets. A connection has only one owner. If the original owner fails, then when new nodes receive packets from the connection, the director chooses a new owner from those nodes.
- Backup owner—The node that stores TCP/UDP state information received from the owner, so that the connection can be seamlessly transferred to a new owner in case of a failure. The backup owner does not take over the connection in the event of a failure. If the owner becomes unavailable, then the first node to receive packets from the connection (based on load balancing) contacts the backup owner for the relevant state information so it can become the new owner.

  As long as the director (see below) is not the same node as the owner, then the director is also the backup owner. If the owner chooses itself as the director, then a separate backup owner is chosen.

For clustering on the Firepower 9300, which can include up to 3 cluster nodes in one chassis, if the backup owner is on the same chassis as the owner, then an additional backup owner will be chosen from another chassis to protect flows from a chassis failure.

If you enable director localization for inter-site clustering, then there are two backup owner roles: the local backup and the global backup. The owner always chooses a local backup at the same site as itself (based on site ID). The global backup can be at any site, and might even be the same node as the local backup. The owner sends connection state information to both backups.

If you enable site redundancy, and the backup owner is at the same site as the owner, then an additional backup owner will be chosen from another site to protect flows from a site failure. Chassis backup and site backup are independent, so in some cases a flow will have both a chassis backup and a site backup.

- Director—The node that handles owner lookup requests from forwarders. When the owner receives a new connection, it chooses a director based on a hash of the source/destination IP address and ports (see below for ICMP hash details), and sends a message to the director to register the new connection. If packets arrive at any node other than the owner, the node queries the director about which node is the owner so it can forward the packets. A connection has only one director. If a director fails, the owner chooses a new director.

  As long as the director is not the same node as the owner, then the director is also the backup owner (see above). If the owner chooses itself as the director, then a separate backup owner is chosen.

  If you enable director localization for inter-site clustering, the cluster uses two director roles: the local director and the global director. For a new-flow packet, the receiving member selects the local director from members at the receiving member's site. The global director can be at any site and might be the same member as the local director. If the original owner fails and localization applies, the local director chooses a new connection owner at the local director's site.

  ICMP/ICMPv6 hash details:

  - For Echo packets, the source port is the ICMP identifier, and the destination port is 0.
  - For Reply packets, the source port is 0, and the destination port is the ICMP identifier.
  - For other packets, both source and destination ports are 0.

- Forwarder—A node that forwards packets to the owner. If a forwarder receives a packet for a connection it does not own, it queries the director for the owner, and then establishes a flow to the owner for any other packets it receives for this connection. The director can also be a forwarder. If you enable director localization, the forwarder queries the local director when localization applies. The forwarder queries the global director if the local director does not know the owner, if the connection is owned at another site, or if localization is bypassed. A cross-site director query is expected behavior in these cases. Note that if a forwarder receives the SYN-ACK packet, it can derive the owner directly from a SYN cookie in the packet, so it does not need to query the director. (If you disable TCP sequence randomization, the SYN cookie is not used; a query to the director is required.) For short-lived flows such as DNS and ICMP, instead of querying, the forwarder immediately sends the packet to the director, which then sends them to the owner. A connection can have multiple forwarders; the most efficient throughput is achieved by a good load-balancing method where there are no forwarders and all packets of a connection are received by the owner.

Note We do not recommend disabling TCP sequence randomization when using clustering. There is a small chance that some TCP sessions won't be established, because the SYN/ACK packet might be dropped.

- Fragment Owner—For fragmented packets, cluster nodes that receive a fragment determine a fragment owner using a hash of the fragment source IP address, destination IP address, and the packet ID. All fragments are then forwarded to the fragment owner over the cluster control link. Fragments may be load-balanced to different cluster nodes, because only the first fragment includes the 5-tuple used in the switch load balance hash. Other fragments do not contain the source and destination ports and may be load-balanced to other cluster nodes. The fragment owner temporarily reassembles the packet so it can determine the director based on a hash of the source/destination IP address and ports. If it is a new connection, the fragment owner will register to be the connection owner. If it is an existing connection, the fragment owner forwards all fragments to the provided connection owner over the cluster control link. The connection owner will then reassemble all fragments.

Port Address Translation Connections

When a connection uses Port Address Translation (PAT), then the PAT type (per-session or multi-session) influences which member of the cluster becomes the owner of a new connection:

- Per-session PAT—The owner is the node that receives the initial packet in the connection.

  By default, TCP and DNS UDP traffic use per-session PAT.

- Multi-session PAT—The owner is always the control node. If a multi-session PAT connection is initially received by a data node, then the data node forwards the connection to the control node.

  By default, UDP (except for DNS UDP) and ICMP traffic use multi-session PAT, so these connections are always owned by the control node.

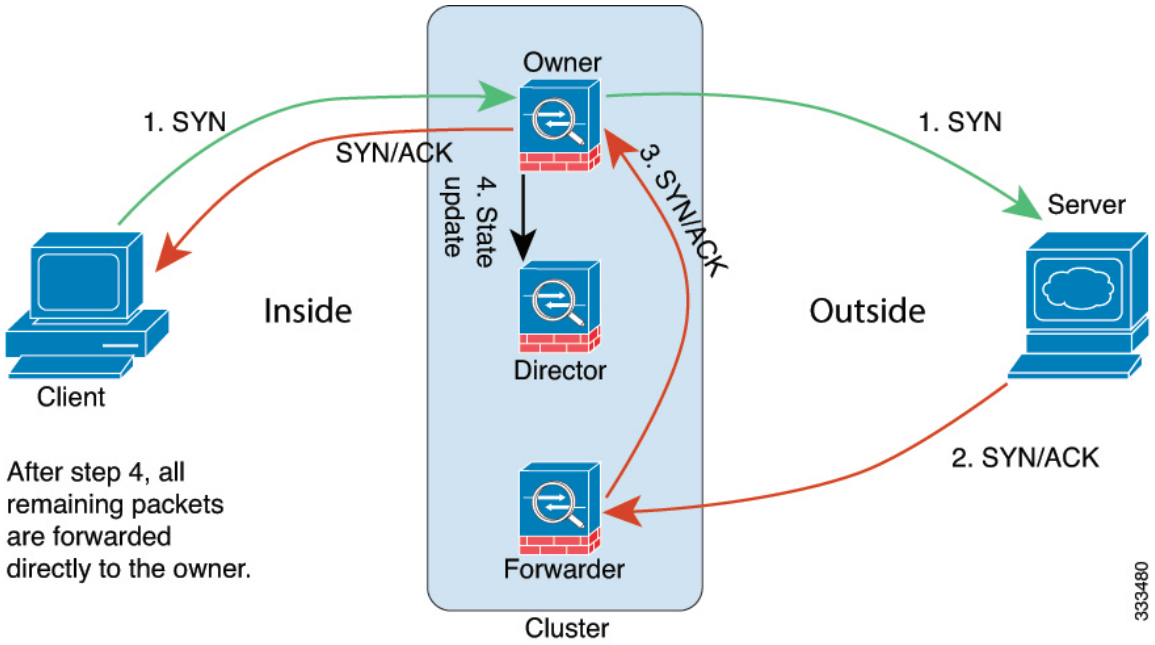
You can change the per-session PAT defaults for TCP and UDP so connections for these protocols are handled per-session or multi-session depending on the configuration. For ICMP, you cannot change from the default multi-session PAT. For more information about per-session PAT, see the firewall configuration guide.

New connection ownership

When a new connection is directed to a node of the cluster via load balancing, that node owns both directions of the connection. If any connection packets arrive at a different node, they are forwarded to the owner node over the cluster control link. For best performance, proper external load balancing is required for both directions of a flow to arrive at the same node, and for flows to be distributed evenly between nodes. If a reverse flow arrives at a different node, it is redirected back to the original node.

Sample Data Flow for TCP

The following example shows the establishment of a new connection.

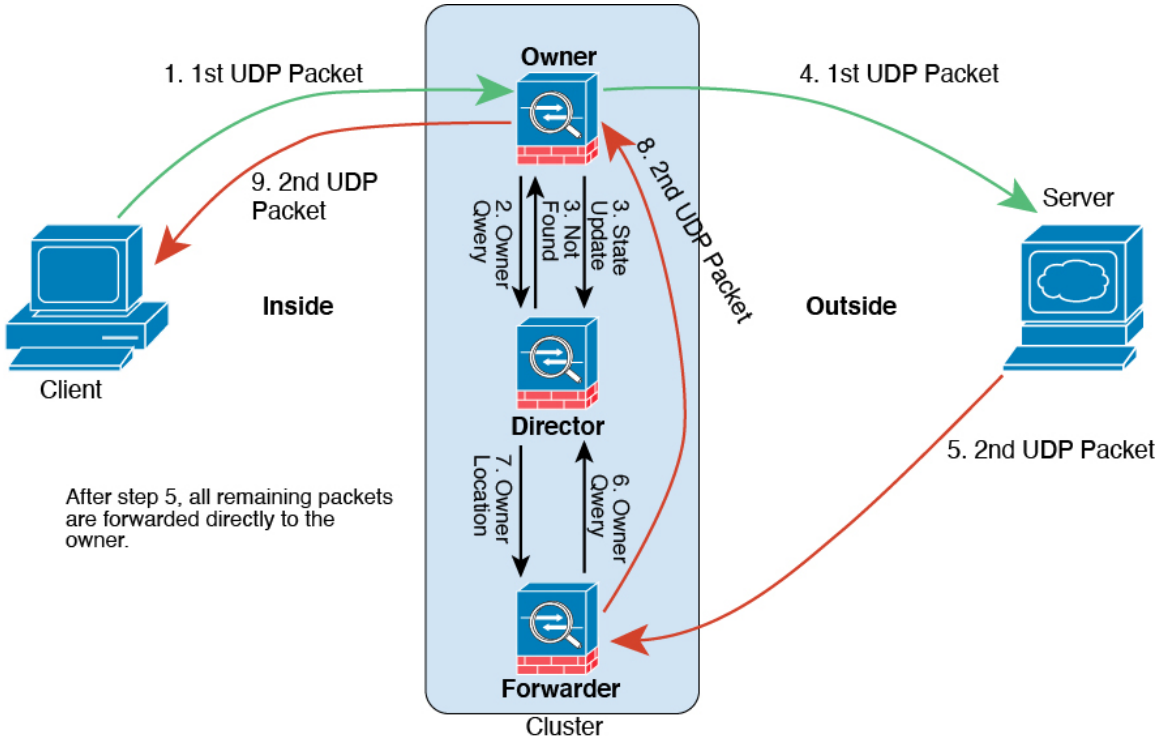


1. The SYN packet originates from the client and is delivered to one ASA (based on the load balancing method), which becomes the owner. The owner creates a flow, encodes owner information into a SYN cookie, and forwards the packet to the server.
2. The SYN-ACK packet originates from the server and is delivered to a different ASA (based on the load balancing method). This ASA is the forwarder.
3. Because the forwarder does not own the connection, it decodes owner information from the SYN cookie, creates a forwarding flow to the owner, and forwards the SYN-ACK to the owner.
4. The owner sends a state update to the director, and forwards the SYN-ACK to the client.
5. The director receives the state update from the owner, creates a flow to the owner, and records the TCP state information as well as the owner. The director acts as the backup owner for the connection.
6. Any subsequent packets delivered to the forwarder will be forwarded to the owner.
7. If packets are delivered to any additional nodes, it will query the director for the owner and establish a flow.
8. Any state change for the flow results in a state update from the owner to the director.

Sample Data Flow for ICMP and UDP

The following example shows the establishment of a new connection.

1. *Figure 51: ICMP and UDP Data Flow*



The first UDP packet originates from the client and is delivered to one ASA (based on the load balancing method).

2. The node that received the first packet queries the director node that is chosen based on a hash of the source/destination IP address and ports.
3. The director finds no existing flow, creates a director flow and forwards the packet back to the previous node. In other words, the director has elected an owner for this flow.
4. The owner creates the flow, sends a state update to the director, and forwards the packet to the server.
5. The second UDP packet originates from the server and is delivered to the forwarder.
6. The forwarder queries the director for ownership information. For short-lived flows such as DNS, instead of querying, the forwarder immediately sends the packet to the director, which then sends it to the owner.
7. The director replies to the forwarder with ownership information.
8. The forwarder creates a forwarding flow to record owner information and forwards the packet to the owner.
9. The owner forwards the packet to the client.

Rebalancing New TCP Connections Across the Cluster

If the load balancing capabilities of the upstream or downstream routers result in unbalanced flow distribution, you can configure new connection rebalancing so nodes with higher new connections per second will redirect new TCP flows to other nodes. No existing flows will be moved to other nodes.

Because this command only rebalances based on connections per second, the total number of established connections on each node is not considered, and the total number of connections may not be equal.

Once a connection is offloaded to a different node, it becomes an asymmetric connection.

Do not configure connection rebalancing for inter-site topologies; you do not want new connections rebalanced to cluster members at a different site.

History for ASAv Clustering

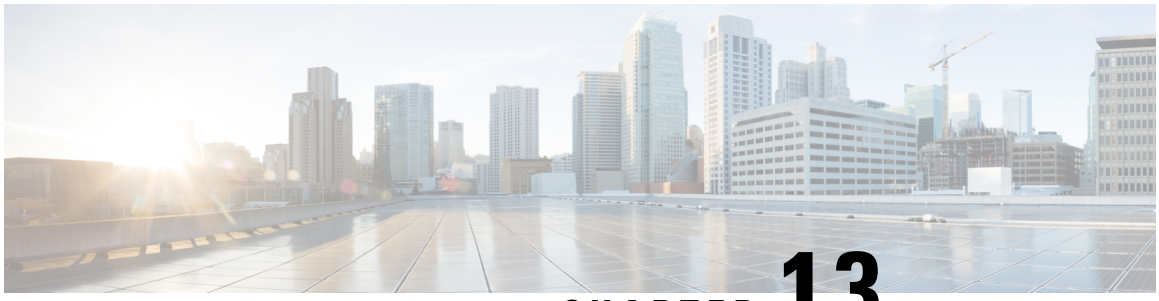
Feature Name	Version	Feature Information
ASAv30, ASAv50, and ASAv100 clustering for VMware and KVM	9.17(1)	The ASAv clustering lets you group up to 16 ASAvs together as a single logical device. A cluster provides all the convenience of a single device (management, integration into a network) while achieving the increased throughput and redundancy of multiple devices. The ASAv clustering supports Individual Interface mode in routed firewall mode; Spanned EtherChannels are not supported. The ASAv uses a VXLAN virtual interface (VNI) for the cluster control link. New/Modified commands: **cluster-interface vni**, **nve-only cluster**, **peer-group**, **show cluster info**, **show cluster info instance-type**, **show nve 1**



PART III

Interfaces

- Basic Interface Configuration, on page 593
- Basic Interface Configuration for Firepower 1010 Switch Ports, on page 607
- EtherChannel Interfaces, on page 625
- VLAN Subinterfaces, on page 637
- VXLAN Interfaces, on page 645
- Routed and Transparent Mode Interfaces, on page 669
- Advanced Interface Configuration, on page 707
- Traffic Zones, on page 717



CHAPTER 13

Basic Interface Configuration

This chapter includes basic interface configuration including Ethernet settings and Jumbo frame configuration.

Note For multiple context mode, complete all tasks in this section in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.

Note For the Firepower 2100 in Platform mode and Firepower 4100/9300 chassis, you configure basic interface settings in the FXOS operating system. See the configuration or getting started guide for your chassis for more information.

- About Basic Interface Configuration, on page 593
- Guidelines for Basic Interface Configuration, on page 596
- Default Settings for Basic Interface Configuration, on page 596
- Enable the Physical Interface and Configure Ethernet Parameters, on page 597
- Enable Jumbo Frame Support (ASAv, ISA 3000), on page 600
- Manage the Network Module for the Secure Firewall 3100, on page 601
- Monitoring Interfaces, on page 604
- Examples for Basic Interfaces, on page 604
- History for Basic Interface Configuration, on page 605

About Basic Interface Configuration

This section describes interface features and special interfaces.

Auto-MDI/MDIX Feature

For RJ-45 interfaces, the default auto-negotiation setting also includes the Auto-MDI/MDIX feature. Auto-MDI/MDIX eliminates the need for crossover cabling by performing an internal crossover when a straight cable is detected during the auto-negotiation phase. Either the speed or duplex must be set to auto-negotiate to enable Auto-MDI/MDIX for the interface. If you explicitly set both the speed and duplex to a fixed value, thus disabling auto-negotiation for both settings, then Auto-MDI/MDIX is also disabled. For

Gigabit Ethernet, when the speed and duplex are set to 1000 and full, then the interface always auto-negotiates; therefore Auto-MDI/MDIX is always enabled and you cannot disable it.

Management Interface

The management interface, depending on your model, is a separate interface just for management traffic.

Management Interface Overview

You can manage the ASA by connecting to:

- Any through-traffic interface
- A dedicated Management *Slot*/*Port* interface (if available for your model)

You may need to configure management access to the interface according to Management Access, on page 1161.

Management *Slot*/*Port* Interface

The following table shows the Management interfaces per model.

Table 21: Management Interfaces Per Model

Model	Management 0/0	Management 0/1	Management 1/0	Management 1/1	Configurable for Through Traffic	Subinterfaces Allowed
Firepower 1000	—	—	—	**Yes**	**Yes**	**Yes**
Firepower 2100	—	—	—	**Yes**	— **Note:** Technically, you can enable through traffic; however, the throughput of this interface is not adequate for data operations.	**Yes**
Secure Firewall 3100	—	—	—	**Yes**	**Yes**	**Yes**
Firepower 4100/9300	N/A The interface ID depends on the physical mgmt-type interface that you assigned to the ASA logical device	—	—	—	—	**Yes**
ISA 3000	—	—	—	**Yes**	—	—

Model	Management 0/0	Management 0/1	Management 1/0	Management 1/1	Configurable for Through Traffic	Subinterfaces Allowed
ASAv	**Yes**	—	—	—	**Yes**	—

Use Any Interface for Management-Only Traffic

You can use any interface as a dedicated management-only interface by configuring it for management traffic, including an EtherChannel interface (see the **management-only** command).

Management Interface for Transparent Mode

In transparent firewall mode, in addition to the maximum allowed through-traffic interfaces, you can also use the Management interface (either the physical interface, or a subinterface (if supported for your model)) as a separate management-only interface.

You cannot use any other interface types as Management interfaces.

For the Firepower 4100/9300 chassis, the management interface ID depends on the mgmt-type interface that you assigned to the ASA logical device.

Note The management interface is not part of a normal bridge group. Note that for operational purposes, it is part of a non-configurable bridge group.

Management-Only Traffic

The Management interface is for to- and from-the-box traffic only and cannot pass through traffic. In any case, since bridge groups cannot communicate with any other bridge groups, it wouldn't be possible for the Management interface to route to a bridge group or vice versa even if it did support through traffic.

Shared MAC Address Table and Switch Connections

In transparent firewall mode, the Management interface updates the ASA's MAC address table in the same manner as a data interface; therefore you should not connect both a Management and a data interface to the same switch unless you configure one of the switch ports as a routed port (by default Catalyst switches share a MAC address for all VLAN switch ports). Otherwise, if traffic arrives on the Management interface from the physically-connected switch, then the ASA updates the MAC address table to use the *Management* interface to access the switch, instead of the data interface. This action causes a temporary traffic interruption; the ASA will not re-update the MAC address table for packets from the switch to the data interface for at least 30 seconds for security reasons.

Multiple Context Mode

In multiple context mode, you cannot share any interfaces, including the Management interface, across contexts. To provide management per context on supported models, you can create subinterfaces of the Management interface and allocate a Management subinterface to each context. Some models do not allow subinterfaces on the Management interface, so per-context management for these models requires you to connect to a data interface. For the Firepower 4100/9300 chassis, the management interface and its subinterfaces are not recognized as specially-allowed management interfaces within the contexts; you must treat a management subinterface as a data interface in this case and add it to a BVI.

Guidelines for Basic Interface Configuration

Transparent Firewall Mode

For multiple context, transparent mode, each context must use different interfaces; you cannot share an interface across contexts.

Failover

You cannot share a failover or state interface with a data interface.

Additional Guidelines

- Some management-related services are not available until a non-management interface is enabled, and the the ASA achieves a “System Ready” state. The ASA generates the following syslog message when it is in a “System Ready” state:

```
%ASA-6-199002: Startup completed.  Beginning operation.
```

Default Settings for Basic Interface Configuration

This section lists default settings for interfaces if you do not have a factory default configuration.

Default State of Interfaces

The default state of an interface depends on the type and the context mode.

In multiple context mode, all allocated interfaces are enabled by default, no matter what the state of the interface is in the system execution space. However, for traffic to pass through the interface, the interface also has to be enabled in the system execution space. If you shut down an interface in the system execution space, then that interface is down in all contexts that share it.

In single mode or in the system execution space, interfaces have the following default states:

- Physical interfaces—Disabled.
- VLAN subinterfaces—Enabled. However, for traffic to pass through the subinterface, the physical interface must also be enabled.
- VXLAN VNI interfaces—Enabled.
- EtherChannel port-channel interfaces (ISA 3000)—Enabled. However, for traffic to pass through the EtherChannel, the channel group physical interfaces must also be enabled.
- EtherChannel port-channel interfaces (Other models)—Disabled.

Note For the Firepower 4100/9300, you can administratively enable and disable interfaces in both the chassis and on the ASA. For an interface to be operational, the interface must be enabled in both operating systems. Because the interface state is controlled independently, you may have a mismatch between the chassis and the ASA.

Default Speed and Duplex

- By default, the speed and duplex for copper (RJ-45) interfaces are set to auto-negotiate.

Default Connector Type

Some models include two connector types: copper RJ-45 and fiber SFP. RJ-45 is the default. You can configure the ASA to use the fiber SFP connectors.

Default MAC Addresses

By default, the physical interface uses the burned-in MAC address, and all subinterfaces of a physical interface use the same burned-in MAC address.

Enable the Physical Interface and Configure Ethernet Parameters

This section describes how to:

- Enable the physical interface
- Set a specific speed and duplex (if available)
- Set Forward Error Correction

This procedure only covers a small subset of Interface settings. Refrain from setting other parameters at this point. For example, you cannot name an interface that you want to use as part of an EtherChannel interface. The exact interface options vary depending on your model and interface type.

Note For the Firepower 4100/9300, you configure basic interface settings in FXOS.

Note For / 1010/ switch ports, see Basic Interface Configuration for Firepower 1010 Switch Ports, on page 607.

Before you begin

For multiple context mode, complete this procedure in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.

Procedure

Step 1 Specify the interface you want to configure:

interface *physical_interface*

Example:

```
ciscoasa(config)# interface gigabitethernet 0/0
```

The *physical_interface* ID includes the type, slot, and port number as type[slot/]port.

The physical interface types include the following:

- **ethernet**

  gigabitethernet
- **tengigabitethernet**
- **management**

Enter the type followed by *slot*/*port*, for example, **gigabitethernet0/1**. A space is optional between the type and the slot/port.

Step 2 (Optional) Set the speed (varies depending on the model).

speed {**auto** | *speed* | **nonegotiate** | **sfp-detect**}

Example:

```
ciscoasa(config-if)# speed 100
```

For Firepower 1100 and 2100 fiber interfaces, **speed nonegotiate** sets the speed to 1000 Mbps and disables link negotiation for flow-control parameters and remote fault information. For other models, see the **negotiate-auto** command.

Choose **sfp-detect** to detect the speed of the installed SFP module and use the appropriate speed. Duplex is always full, autonegotiation is enabled, and FEC is set to auto. For dual-speed transceivers, the lower speed is used. This option is useful if you later change the network module to a different model, and want the speed to update automatically. Some switches and transceivers don't support autonegotiation, especially for higher speed interfaces. In this case, set the interface on the ASA to a manual speed and also set **no negotiate-auto** so the link can come up.

Step 3 Set autonegotiation for the link status and flow control.

negotiate-auto

Autonegotiation is set separately from the speed. Some switches don't support autonegotiation, especially for higher speed interfaces. In this case, set the interface on the ASA to a manual speed and also set **no negotiate-auto** so the link can come up.

Example:

```
ciscoasa(config-if)# negotiate-auto
```

Step 4 (Optional) Set the duplex:

duplex {**auto** | **full** | **half**}

SFP interfaces only support full duplex.

Example:

```
ciscoasa(config-if)# duplex full
```

Step 5 (Optional) Set Forward Error Correction (FEC) for 25 Gbps and higher interfaces.

fec {**auto** | **cl108-rs** | **cl74-fc** | **disable**}

For an EtherChannel member interface, you must configure FEC before you add it to the EtherChannel. If you remove the interface from the EtherChannel, after rebooting, you may need to reconfigure the FEC for the interface.

Some switches don't support auto mode for FEC, especially for larger interfaces. Be sure to manually configure the setting, depending on the switch support. We don't recommend disabling FEC; if you do so, the EtherChannel may not come up properly.

The setting chosen when you use **auto** depends on the transceiver type and whether the interface is fixed (built-in) or on a network module.

Table 22: Default FEC for Auto Setting

Transceiver Type	Fixed Port Default FEC	Network Module Default FEC
25G-SR	cl74-fc	cl108-rs
25G-LR	cl74-fc	cl108-rs
10/25G-CSR	cl74-fc	cl74-fc
25G-AOC*x*M	cl74-fc	cl74-fc
25G-CU2.5/3M	Auto-Negotiate	Auto-Negotiate
25G-CU4/5M	Auto-Negotiate	Auto-Negotiate

Step 6 Enable the interface:

no shutdown

Example:

```
ciscoasa(config-if)# no shutdown
```

To disable the interface, enter the **shutdown** command. If you enter the **shutdown** command, you also shut down all subinterfaces. If you shut down an interface in the system execution space, then that interface is shut down in all contexts that share it.

Enable Jumbo Frame Support (ASAv, ISA 3000)

A jumbo frame is an Ethernet packet larger than the standard maximum of 1518 bytes (including Layer 2 header and VLAN header), up to 9216 bytes. You can enable support for jumbo frames for all interfaces by increasing the amount of memory to process Ethernet frames. Assigning more memory for jumbo frames might limit the maximum use of other features, such as ACLs. Note that the ASA MTU sets the payload size not including the Layer 2 (14 bytes) and VLAN header (4 bytes), so the maximum MTU is 9198, depending on your model.

This procedure only applies to the ISA 3000 and the ASAv. Other models support jumbo frames by default.

Jumbo frames are not supported on the ASAv5 and ASAv10 with less than 8GB RAM.

Before you begin

- In multiple context mode, set this option in the system execution space.
- Changes in this setting require you to reload the ASA.
- Be sure to set the MTU for each interface that needs to transmit jumbo frames to a higher value than the default 1500; for example, set the value to 9198 using the **mtu** command. In multiple context mode, set the MTU within each context.
- Be sure to adjust the TCP MSS, either to disable it for non-IPsec traffic (use the **sysopt connection tcpmss 0** command), or to increase it in accord with the MTU.

Procedure

Enable jumbo frame support:

jumbo-frame reservation

Examples

The following example enables jumbo frame reservation, saves the configuration, and reloads the ASA:

```
ciscoasa(config)# jumbo-frame reservation
WARNING: this command will take effect after the running-config is saved
and the system has been rebooted. Command accepted.

ciscoasa(config)# write memory
Building configuration...
Cryptochecksum: 718e3706 4edb11ea 69af58d0 0a6b7cb5

70291 bytes copied in 3.710 secs (23430 bytes/sec)
[OK]
ciscoasa(config)# reload
Proceed with reload? [confirm] Y
```

Manage the Network Module for the Secure Firewall 3100

If you install a network module before you first power on the firewall, no action is required; the network module is enabled and ready for use.

If you need to make changes to your network module installation after initial bootup, then see the following procedures.

Add a Network Module

To add a network module to a firewall after initial bootup, perform the following steps. Adding a new module requires a reload. For clustering or failover, zero downtime is not supported, so make sure to perform this procedure during a maintenance window.

Procedure

Step 1 Install the network module according to the hardware installation guide. You can install the network module while the firewall is powered on.

For clustering or failover, install the network module on all nodes.

Step 2 Reload the firewall; see Reload the ASA, on page 47.

For clustering or failover, reload all nodes. Because nodes with different network modules cannot join the cluster/failover pair, you need to reload all nodes with the new module before they can reform the cluster/failover pair.

Step 3 Enable the network module.

no netmod 2 disable

For clustering or failover, perform this step on the control node/active unit; the module state is replicated to the other nodes.

Example:

```
ciscoasa(config)# no netmod 2 disable
```

Hot Swap the Network Module

You can hot swap a network module for a new module of the same type without having to reload. However, you must shut down the current module to remove it safely. This procedure describes how to shut down the old module, install a new module, and enable it.

For clustering or failover, you cannot disable a network module if the cluster control link/failover link is on the module.

Procedure

Step 1 For clustering or failover, perform the following steps.

- **Clustering**—Ensure the unit you want to perform the hot swap on is a data node (see Change the Control Node, on page 381); then disable clustering on the node. See Become an Inactive Node, on page 378 or Deactivate a Node, on page 379.

  If the cluster control link is on the network module, you must leave the cluster. See Leave the Cluster, on page 380. Disabling the network module with an active cluster control link is not allowed.

- **Failover**—Ensure the unit you want to perform the hot swap on is the standby node. See Force Failover, on page 302.

  If the failover link is on the network module, you must disable failover. See Disable Failover, on page 303. Disabling the network module with an active failover link is not allowed.

Step 2 Disable the network module.

netmod 2 disable

Example:

```
ciscoasa(config)# netmod 2 disable
```

Step 3 Replace the network module according to the hardware installation guide. You can replace the network module while the firewall is powered on.

Step 4 Enable the network module.

no netmod 2 disable

Example:

```
ciscoasa(config)# no netmod 2 disable
```

Step 5 For clustering or failover, perform the following steps.

- **Clustering**—Add the node back to the cluster. See Rejoin the Cluster, on page 379.
- **Failover**—If you disabled failover, then reform failover.

Replace the Network Module with a Different Type

If you replace a network module with a different type, then a reload is required. If the new module has fewer interfaces than the old module, you will have to manually remove any configuration related to interfaces that will no longer be present. For clustering or failover, zero downtime is not supported, so make sure to perform this procedure during a maintenance window.

Procedure

Step 1 Disable the network module.

netmod 2 disable

For clustering or failover, perform this step on the control node/active unit; the module state is replicated to the other nodes. Do not save the configuration; when you reload, the module will be enabled using the saved configuration.

Example:

```
ciscoasa(config)# netmod 2 disable
```

Step 2 Replace the network module according to the hardware installation guide. You can replace the network module while the firewall is powered on.

For clustering or failover, install the network module on all nodes.

Step 3 Reload the firewall; see Reload the ASA, on page 47.

For clustering or failover, reload all nodes. Because nodes with different network modules cannot join the cluster/failover pair, you need to reload all nodes with the new module before they can reform the cluster/failover pair.

Step 4 If you saved the configuration before reloading, you will have to reenable the module.

Remove the Network Module

If you want to permanently remove the network module, follow these steps. Removing a network module requires a reload. For clustering or failover, zero downtime is not supported, so make sure to perform this procedure during a maintenance window.

Before you begin

For clustering or failover, make sure the cluster/failover link is not on the network module; you cannot remove the module in this case.

Procedure

Step 1 Disable the network module and save the configuration.

netmod 2 disable

write memory

For clustering or failover, perform this step on the control node/active unit; the module state is replicated to the other nodes.

Example:

```
ciscoasa(config)# netmod 2 disable
ciscoasa(config)# write memory
```

Step 2 Remove the network module according to the hardware installation guide. You can remove the network module while the firewall is powered on.

For clustering or failover, remove the network module on all nodes.

Step 3 Reload the firewall; see Reload the ASA, on page 47.

For clustering or failover, reload all nodes. Because nodes with different network modules cannot join the cluster/failover pair, you need to reload all nodes without the module before they can reform the cluster/failover pair.

Monitoring Interfaces

See the following commands.

Note For Firepower and Secure Firewall models, some statistics are not shown using the ASA commands. You must view more detailed interface statistics using FXOS commands.

- /eth-uplink/fabric# **show interface**
- /eth-uplink/fabric# **show port-channel**
- /eth-uplink/fabric/interface# **show stats**

For the Firepower 2100 in Platform mode, see also the following FXOS connect local-mgmt commands:

- (local-mgmt)# **show portmanager counters**
- (local-mgmt)# **show lacp**
- (local-mgmt)# **show portchannel**

See the FXOS troubleshooting guide for more information.

- **show interface**

  Displays interface statistics.

- **show interface ip brief**

  Displays interface IP addresses and status.

Examples for Basic Interfaces

See the following configuration examples.

Physical Interface Parameters Example

The following example configures parameters for the physical interface in single mode:

```
interface gigabitethernet 0/1
speed 1000
duplex full
no shutdown
```

Multiple Context Mode Example

The following example configures interface parameters in multiple context mode for the system configuration, and allocates the gigabitethernet 0/1.1 subinterface to contextA:

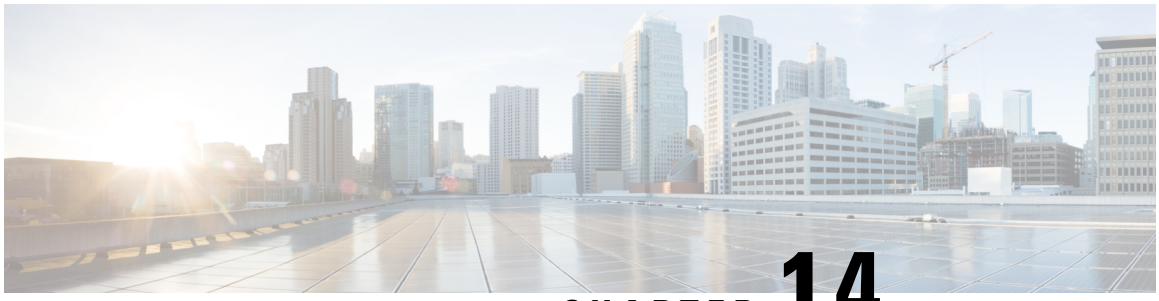
```
interface gigabitethernet 0/1
speed 1000
duplex full
no shutdown
interface gigabitethernet 0/1.1
vlan 101
context contextA
allocate-interface gigabitethernet 0/1.1
```

History for Basic Interface Configuration

Table 23: History for Interfaces

Feature Name	Releases	Feature Information
Support for hot swapping the network module for the Secure Firewall 3100	9.17(1)	You can add or remove the network module on the Secure Firewall 3100 while the firewall is powered up. To replace a module with another module of the same type, you do not need to reboot. After initial bootup, adding a module, permanently removing a module, or replacing a module with a new type requires a reboot. New/Modified commands: **netmod**
Support for Forward Error Correction for the Secure Firewall 3100	9.17(1)	Secure Firewall 3100 25 Gbps interfaces support Forward Error Correction (FEC). FEC is enabled by default and set to Auto. New/Modified commands: **fec**
Support for setting the speed based on the SFP for the Secure Firewall 3100	9.17(1)	The Secure Firewall 3100 supports speed detection for interfaces based on the SFP installed. Detect SFP is enabled by default. This option is useful if you later change the network module to a different model, and want the speed to update automatically. New/Modified commands: **speed sfp-detect**

Feature Name	Releases	Feature Information
Secure Firewall 3100 auto-negotiation can be enabled or disabled for 1Gigabit and higher interfaces.	9.17(1)	Secure Firewall 3100 auto-negotiation can be enabled or disabled separately from speed for 1Gigabit and higher interfaces. New/Modified commands: **negotiate-auto**
Speed auto-negotiation can be disabled on fiber interfaces on the Firepower 1100 and 2100	9.14(1)	You can now configure a Firepower 1100 or 2100 fiber interface to disable auto-negotiation. For 10GB interfaces, you can configure the speed down to 1GB without auto-negotiation; you cannot disable auto-negotiation for an interface with the speed set to 10GB. New/Modified commands: **speed nonegotiate**
Through traffic support on the Management 0/0 interface for the ASAv	9.6(2)	You can now allow through traffic on the Management 0/0 interface on the ASAv. Previously, only the ASAv on Microsoft Azure supported through traffic; now all ASAvs support through traffic. You can optionally configure this interface to be management-only, but it is not configured by default. We modified the following command: **management-only**
Support for Pause Frames for Flow Control on Gigabit Ethernet Interfaces	8.2(5)/8.4(2)	You can now enable pause (XOFF) frames for flow control for Gigabit Ethernet interfaces on all ASA models. We modified the following command: **flowcontrol**.
Support for Pause Frames for Flow Control on the ASA 5580 Ten Gigabit Ethernet Interfaces	8.2(2)	You can now enable pause (XOFF) frames for flow control. This feature is also supported on the ASA 5585-X. We introduced the following command: **flowcontrol**.
Jumbo packet support for the ASA 5580	8.1(1)	The ASA 5580 supports jumbo frames. A jumbo frame is an Ethernet packet larger than the standard maximum of 1518 bytes (including Layer 2 header and FCS), up to 9216 bytes. You can enable support for jumbo frames for all interfaces by increasing the amount of memory to process Ethernet frames. Assigning more memory for jumbo frames might limit the maximum use of other features, such as ACLs. This feature is also supported on the ASA 5585-X. We introduced the following command: **jumbo-frame reservation**.
Gigabit Ethernet Support for the ASA 5510 Security Plus License	7.2(3)	The ASA 5510 now supports GE (Gigabit Ethernet) for port 0 and 1 with the Security Plus license. If you upgrade the license from Base to Security Plus, the capacity of the external Ethernet0/0 and Ethernet0/1 ports increases from the original FE (Fast Ethernet) (100 Mbps) to GE (1000 Mbps). The interface names will remain Ethernet 0/0 and Ethernet 0/1. Use the **speed** command to change the speed on the interface and use the **show interface** command to see what speed is currently configured for each interface.
Increased interfaces for the Base license on the ASA 5510	7.2(2)	For the Base license on the ASA 5510, the maximum number of interfaces was increased from 3 plus a management interface to unlimited interfaces.



CHAPTER 14

Basic Interface Configuration for Firepower 1010 Switch Ports

You can configure each Firepower 1010 interface to run as a regular firewall interface or as a Layer 2 hardware switch port. This chapter includes tasks for starting your switch port configuration, including enabling or disabling the switch mode and creating VLAN interfaces and assigning them to switch ports. This chapter also describes how to customize Power over Ethernet (PoE) on supported interfaces.

- About switch ports, on page 607
- Prerequisites for switch ports, on page 608
- Guidelines for switch ports, on page 608
- Configure switch ports and Power Over Ethernet, on page 610
- Monitoring switch ports, on page 618
- Examples for Switch Ports, on page 619
- History for switch ports, on page 623

About switch ports

This section describes the switch ports for models with Layer 2 switches.

Understanding switch ports and interfaces

Ports and interfaces

For each physical interface, you can set its operation as a firewall interface or as a switch port. See the following information about physical interface and port types as well as logical VLAN interfaces to which you assign switch ports:

- Physical firewall interface—In routed mode, these interfaces forward traffic between networks at Layer 3, using the configured security policy to apply firewall and VPN services. In transparent mode, these interfaces are bridge group members that forward traffic between the interfaces on the same network at Layer 2, using the configured security policy to apply firewall services. In routed mode, you can also use Integrated Routing and Bridging with some interfaces as bridge group members and others as Layer 3 interfaces. By default, the Ethernet 1/1 interface is configured as a firewall interface.
- Physical switch port—Switch ports forward traffic at Layer 2, using the switching function in hardware. Switch ports on the same VLAN can communicate with each other using hardware switching, and traffic

is not subject to the ASA security policy. Access ports accept only untagged traffic, and you can assign them to a single VLAN. Trunk ports accept untagged and tagged traffic, and can belong to more than one VLAN. By default, Ethernet 1/2 and higher are configured as access switch ports on VLAN 1. You cannot configure the Management interface as a switch port.

- Logical VLAN interface—These interfaces operate the same as physical firewall interfaces, with the exception being that you cannot create subinterfaces, or EtherChannel interfaces. When a switch port needs to communicate with another network, then the ASA device applies the security policy to the VLAN interface and routes to another logical VLAN interface or firewall interface. You can even use Integrated Routing and Bridging with VLAN interfaces as bridge group members. Traffic between switch ports on the same VLAN are not subject to the ASA security policy, but traffic between VLANs in a bridge group are subject to the security policy, so you may choose to layer bridge groups and switch ports to enforce the security policy between certain segments.

Power Over Ethernet

PoE is available on Ethernet 1/7 and Ethernet 1/8 using IEEE 802.3af (PoE) and 802.3at (PoE+) up to 30 watts per port.

PoE+ uses Link Layer Discovery Protocol (LLDP) to negotiate the power level. Power is only supplied when needed.

If you shut down the interface, then you disable power to the device.

Auto-MDI/MDIX feature

For all switch ports, the default auto-negotiation setting also includes the Auto-MDI/MDIX feature. Auto-MDI/MDIX eliminates the need for crossover cabling by performing an internal crossover when a straight cable is detected during the auto-negotiation phase. Either the speed or duplex must be set to auto-negotiate to enable Auto-MDI/MDIX for the interface. If you explicitly set both the speed and duplex to a fixed value, thus disabling auto-negotiation for both settings, then Auto-MDI/MDIX is also disabled. When the speed and duplex are set to 1000 and full, then the interface always auto-negotiates; therefore Auto-MDI/MDIX is always enabled and you cannot disable it.

Prerequisites for switch ports

Model support

- Firepower 1010

Guidelines for switch ports

Context mode

- The Firepower 1010 does not support multiple context mode.
- The Secure Firewall 1210/1220 does not support multiple context mode.

Failover and clustering

- No cluster support.
- Active/Standby failover support only.
- You should not use the switch port functionality when using Failover. Because the switch ports operate in hardware, they continue to pass traffic on both the active *and* the standby units. Failover is designed to prevent traffic from passing through the standby unit, but this feature does not extend to switch ports. In a normal Failover network setup, active switch ports on both units will lead to network loops. We suggest that you use external switches for any switching capability. Note that VLAN interfaces can be monitored by failover, while switch ports cannot. Theoretically, you can put a single switch port on a VLAN and successfully use Failover, but a simpler setup is to use physical firewall interfaces instead.
- You can only use a firewall interface as the failover link.

Logical VLAN interfaces (SVIs)

- If you also use VLAN subinterfaces on a firewall interface, you cannot use the same VLAN ID as for a logical VLAN interface.
- MAC Addresses:
  - Routed firewall mode—All VLAN interfaces share a MAC address. Ensure that any connected switches can support this scenario. If the connected switches require unique MAC addresses, you can manually assign MAC addresses. See Manually Configure the MAC Address, on page 711.
  - Transparent firewall mode—Each VLAN interface has a unique MAC address. You can override the generated MAC addresses if desired by manually assigning MAC addresses. See Manually Configure the MAC Address, on page 711.

Bridge groups

You cannot mix logical VLAN interfaces and physical firewall interfaces in the same bridge group.

VLAN interface and switch port unsupported features

VLAN interfaces and switch ports do not support:

- Dynamic routing
- Multicast routing
- Policy based routing
- Equal-Cost Multi-Path routing (ECMP)
- VXLAN
- EtherChannels—Switch ports cannot be part of an EtherChannel. PoE is also not supported on a port in an EtherChannel.
- Failover and state link
- Traffic zones
- Security group tagging (SGT)

Other Guidelines and Limitations

- You can configure a maximum of 60 named interfaces.
- You cannot configure the Management interface as a switch port.

Default Settings

- Ethernet 1/1 is a firewall interface.
- Ethernet 1/2 through Ethernet 1/8 are switch ports assigned to VLAN 1.
- Default Speed and Duplex—By default, the speed and duplex are set to auto-negotiate.

Configure switch ports and Power Over Ethernet

To configure switch ports and PoE, complete the following tasks.

Enable or disable switch port mode

You can set each interface independently to be either a firewall interface or a switch port. By default, Ethernet 1/1 is a firewall interface, and the remaining Ethernet interfaces are configured as switch ports.

Procedure

Step 1 Enter interface configuration mode.

interface ethernet1/*port*

- *port*—Sets the port, 1 through 8.

You cannot set the Management 1/1 interface to switch port mode.

Example:

```
ciscoasa(config)# interface ethernet1/4
ciscoasa(config-if)#
```

Step 2 Enable switch port mode.

switchport

If this interface is already in switchport mode, you are prompted for switch port parameters instead of changing the mode.

```
ciscoasa(config-if)# switchport
ciscoasa(config-if)# switchport ?
interface mode commands/options:
  access         Set access mode characteristics of the interface
  mode           Set trunking mode of the interface
  monitor        Monitor another interface
  protected      Configure an interface to be a protected port
```

```
  trunk          Set trunking characteristics of the interface
<cr>
ciscoasa(config-if)#
```

Step 3 Disable switch port mode.

no switchport

```
ciscoasa(config-if)# no switchport
ciscoasa(config-if)# switchport ?

interface mode commands/options:
<cr>
```

Example

The following example sets Ethernet 1/3 and 1/4 to firewall mode:

```
ciscoasa(config)# interface ethernet1/3
ciscoasa(config-if)# no switchport
ciscoasa(config-if)# interface ethernet1/3
ciscoasa(config-if)# no switchport
ciscoasa(config-if)#
```

Configure a VLAN interface

This section describes how to configure VLAN interfaces (SVIs) for use with associated switch ports. You can create up to 60 VLAN interfaces associated with switch ports.

Procedure

Step 1 Add a VLAN interface.

interface vlan *id*

- *id*—Sets the VLAN ID for this interface, between 1 and 4070, excluding IDs in the range 3968 to 4047, which are reserved for internal use.

Example:

```
ciscoasa(config)# interface vlan 100
ciscoasa(config-if)#
```

Step 2 (Optional) Disable forwarding to another VLAN.

no forward interface *vlan_id*

- *vlan_id*—Specifies the VLAN ID to which this VLAN interface cannot initiate traffic.

For example, you have one VLAN assigned to the outside for internet access, one VLAN assigned to an inside business network, and a third VLAN assigned to your home network. The home network does not need to access the business network, so you can use the **no forward interface** command on the home VLAN; the business network can access the home network, but the home network cannot access the business network.

Example:

```
ciscoasa(config-if)# no forward interface 200
ciscoasa(config-if)#
```

Configure switch ports as access ports

To assign a switch port to a single VLAN, configure it as an access port. Access ports accept only untagged traffic. By default, Ethernet 1/2 through Ethernet 1/8 switch ports are enabled and assigned to VLAN 1.



Note The device does not support Spanning Tree Protocol for loop detection in the network. Therefore you must ensure that any connection with the ASA does not end up in a network loop.

Procedure

Step 1 Enter interface configuration mode.

interface ethernet1/*port*

- *port*—Sets the port, 1 through 8.

Example:

```
ciscoasa(config)# interface ethernet1/4
ciscoasa(config-if)#
```

Step 2 Assign this switch port to a VLAN.

switchport access vlan *number*

- *number*—Sets the VLAN ID, between 1 and 4070. The default is VLAN 1.

Example:

```
ciscoasa(config-if)# switchport access vlan 100
ciscoasa(config-if)#
```

Step 3 (Optional) Set this switch port as protected, so you can prevent the switch port from communicating with other protected switch ports on the same VLAN.

switchport protected

You might want to prevent switch ports from communicating with each other if: the devices on those switch ports are primarily accessed from other VLANs; you do not need to allow intra-VLAN access; and you want to isolate the devices from each other in case of infection or other security breach. For example, if you have a DMZ that hosts three web servers, you can isolate the web servers from each other if you apply the **switchport protected** command to each switch port. The inside and outside networks can both communicate with all three web servers, and vice versa, but the web servers cannot communicate with each other.

Example:

```
ciscoasa(config-if)# switchport protected
ciscoasa(config-if)#
```

Step 4 (Optional) Set the speed.

speed {**auto** | **10** | **100** | **1000**}

The default is **auto**.

Example:

```
ciscoasa(config-if)# speed 100
ciscoasa(config-if)#
```

Step 5 (Optional) Set the duplex.

duplex {**auto** | **full** | **half**}

The default is **auto**.

Example:

```
ciscoasa(config-if)# duplex half
ciscoasa(config-if)#
```

Step 6 Enable the switch port.

no shutdown

To disable the switch port, enter the **shutdown** command.

Example:

```
ciscoasa(config-if)# no shutdown
ciscoasa(config-if)#
```

Example

The following example assigns Ethernet 1/3, Ethernet 1/4, and Ethernet 1/5 to VLAN 101, and sets Ethernet 1/3 and Ethernet 1/4 as protected:

```
ciscoasa(config)# interface ethernet1/3
ciscoasa(config-if)# switchport access vlan 101
ciscoasa(config-if)# switchport protected
```

```
ciscoasa(config-if)# no shutdown
ciscoasa(config-if)# interface ethernet1/4
ciscoasa(config-if)# switchport access vlan 101
ciscoasa(config-if)# switchport protected
ciscoasa(config-if)# no shutdown
ciscoasa(config-if)# interface ethernet1/5
ciscoasa(config-if)# switchport access vlan 101
ciscoasa(config-if)# no shutdown
```

Configure switch ports as trunk ports

This procedure describes how to create a trunk port that can carry multiple VLANs using 802.1Q tagging. Trunk ports accept untagged and tagged traffic. Traffic on allowed VLANs pass through the trunk port unchanged.

When the trunk receives untagged traffic, it tags it to the native VLAN ID so that the ASA can forward the traffic to the correct switch ports, or can route it to another firewall interface. When the ASA sends native VLAN ID traffic out of the trunk port, it removes the VLAN tag. Be sure to set the same native VLAN on the trunk port on the other switch so that the untagged traffic will be tagged to the same VLAN.

Procedure

Step 1 Enter interface configuration mode.

interface ethernet1/*port*

- *port*—Sets the port, 1 through 8.

Example:

```
ciscoasa(config)# interface ethernet1/4
ciscoasa(config-if)#
```

Step 2 Make this switch port a trunk port.

switchport mode trunk

To restore this port to access mode, enter the **switchport mode access** command.

Example:

```
ciscoasa(config-if)# switchport mode trunk
ciscoasa(config-if)#
```

Step 3 Assign VLANs to this trunk.

switchport trunk allowed vlan *vlan_range*

- *vlan_range*—Sets the VLAN IDs between 1 and 4070. You can identify up to 20 IDs in one of the following ways:
  - A single number (n)

- A range (n-x)
- Numbers and ranges separated by commas, for example:

  5,7-10,13,45-100

You can enter spaces instead of commas, but the command is saved to the configuration with commas.

If you include the native VLAN in this command, it is ignored; the trunk port always removes the VLAN tagging when sending native VLAN traffic out of the port. Moreover, it will not receive traffic that still has native VLAN tagging.

Example:

```
ciscoasa(config-if)# switchport trunk allowed vlan 100,200,300
ciscoasa(config-if)#
```

Step 4 Set the native VLAN.

switchport trunk native vlan *vlan_id*

- *vlan_range*—Sets the VLAN ID between 1 and 4070. The default value is VLAN 1.

Each port can only have one native VLAN, but every port can have either the same or a different native VLAN.

Example:

```
ciscoasa(config-if)# switchport trunk native vlan 2
ciscoasa(config-if)#
```

Step 5 (Optional) Set this switch port as protected, so you can prevent the switch port from communicating with other protected switch ports on the same VLAN.

switchport protected

You might want to prevent switch ports from communicating with each other if: the devices on those switch ports are primarily accessed from other VLANs; you do not need to allow intra-VLAN access; and you want to isolate the devices from each other in case of infection or other security breach. For example, if you have a DMZ that hosts three web servers, you can isolate the web servers from each other if you apply the **switchport protected** command to each switch port. The inside and outside networks can both communicate with all three web servers, and vice versa, but the web servers cannot communicate with each other.

Example:

```
ciscoasa(config-if)# switchport protected
ciscoasa(config-if)#
```

Step 6 (Optional) Set the speed.

speed {**auto** | **10** | **100** | **1000**}

The default is **auto**.

Example:

```
ciscoasa(config-if)# speed 100
```

```
ciscoasa(config-if)#
```

Step 7 (Optional) Set the duplex.

duplex {**auto** | **full** | **half**}

The default is **auto**.

Example:

```
ciscoasa(config-if)# duplex half
ciscoasa(config-if)#
```

Step 8 Enable the switch port.

no shutdown

To disable the switch port, enter the **shutdown** command.

Example:

```
ciscoasa(config-if)# no shutdown
ciscoasa(config-if)#
```

Example

The following example sets Ethernet 1/6 as a trunk port with VLANs 20 through 30, and sets the native VLAN as 4:

```
ciscoasa(config)# interface ethernet1/6
ciscoasa(config-if)# switchport mode trunk
ciscoasa(config-if)# switchport trunk allowed vlan 20-30
ciscoasa(config-if)# switchport trunk native vlan 4
ciscoasa(config-if)# no shutdown
```

Configure Power Over Ethernet

Power over Ethernet (PoE) ports provide power for devices such as IP phones or wireless access points. PoE is enabled by default. This procedure describes how to disable and enable PoE and how to set optional parameters.

Procedure

Step 1 Enter interface configuration mode.

interface ethernet1/{**7** | **8**}

Example:

```
ciscoasa(config)# interface ethernet1/7
ciscoasa(config-if)#
```

Step 2 Enable or disable PoE.

power inline {**auto** | **never** | **consumption wattage** *milliwatts*}

- **auto**—Delivers power automatically to the powered device using a wattage appropriate to the class of the powered device. The firewall uses LLDP to further negotiate the correct wattage. When you connect a device of a certain class, it will be provisioned up to the maximum for that class in case it ever needs to use more power. For example, if you add a class 4 device that requests 12.95W, it will be allocated 30W even if it doesn't currently use all that power. Some devices can renegotiate power needs. If you know your device needs less power than is allocated, you can instead set the **consumption wattage** manually to free up the power for other devices.
- **never**—Disables PoE.
- **consumption wattage** *milliwatts*—Manually specifies the wattage in milliwatts, from 4000 to 30000. Use this command if you want to set the watts manually and disable LLDP negotiation. For manual allocation, the class will show as **n/a** in the show output because the class is not used to decide the power consumption.

View the current PoE status using the **show power inline** command.

Example:

```
ciscoasa(config-if)# power inline auto
ciscoasa(config-if)# show power inline
   Interface     Power   Class   Current (mA)   Voltage (V)
   ---------     -----   -----   ------------   -----------
   Ethernet1/1   n/a     n/a     n/a            n/a
   Ethernet1/2   n/a     n/a     n/a            n/a
   Ethernet1/3   n/a     n/a     n/a            n/a
   Ethernet1/4   n/a     n/a     n/a            n/a
   Ethernet1/5   n/a     n/a     n/a            n/a
   Ethernet1/6   n/a     n/a     n/a            n/a
   Ethernet1/7   On      4       121.00         53.00
   Ethernet1/8   On      4       88.00          53.00
```

Example

The following example manually sets the wattage for Ethernet 1/7 and sets the power to auto for Ethernet 1/8:

```
ciscoasa(config)# interface ethernet1/7
ciscoasa(config-if)# power inline consumption wattage 10000
ciscoasa(config-if)# interface ethernet1/8
ciscoasa(config-if)# power inline auto
ciscoasa(config-if)#
```

Monitoring switch ports

- **show interface**

  Displays interface statistics.

- **show interface ip brief**

  Displays interface IP addresses and status.

- **show switch vlan**

  Displays the VLAN–to–switch port association.

```
ciscoasa# show switch vlan
VLAN Name                 Status    Ports
---- -------------------- --------- --------------------
1    -                     down      Ethernet1/3,
                                     Ethernet1/4,
                                     Ethernet1/5,
                                     Ethernet1/6
                                     Ethernet1/7,
                                     Ethernet1/8
10   inside                up        Ethernet1/1
20   outside               up        Ethernet1/2
```

- **show switch mac-address-table**

  Shows the static and dynamic MAC address entries.

```
ciscoasa# show switch mac-address-table
Legend: Age - entry expiration time in seconds
  Mac Address  | VLAN |       Type       | Age | Port
-------------------------------------------------------
0c75.bd11.c504 | 0010 |     dynamic      | 330 | In0/0
885a.92f6.c6e3 | 0010 |     dynamic      | 330 | Et1/1
0c75.bd11.c504 | 0020 |     dynamic      | 330 | In0/0
885a.92f6.c45b | 0020 |     dynamic      | 330 | Et1/2
```

- **show arp**

  Shows dynamic, static, and proxy ARP entries. Dynamic ARP entries include the age of the ARP entry in seconds. Static ARP entries include a dash (-) instead of the age, and proxy ARP entries state “alias.” The following is sample output from the **show arp** command. The first entry is a dynamic entry aged 2 seconds. The second entry is a static entry, and the third entry is from proxy ARP.

```
ciscoasa# show arp
outside 10.86.194.61 0011.2094.1d2b 2
outside 10.86.194.1 001a.300c.8000 -
outside 10.86.195.2 00d0.02a8.440a alias
```

- **show power inline**

  Shows PoE+ status.

```
ciscoasa# show power inline
   Interface     Power   Class   Current (mA)   Voltage (V)
```

```
---------     -----   -----   ------------   -----------
Ethernet1/1   n/a     n/a     n/a            n/a
Ethernet1/2   n/a     n/a     n/a            n/a
Ethernet1/3   n/a     n/a     n/a            n/a
Ethernet1/4   n/a     n/a     n/a            n/a
Ethernet1/5   n/a     n/a     n/a            n/a
Ethernet1/6   n/a     n/a     n/a            n/a
Ethernet1/7   On      4       121.00         53.00
Ethernet1/8   On      4       88.00          53.00
```

Examples for Switch Ports

The following topics provide examples for configuring switch ports in routed and transparent mode.

Routed Mode Example

The following example creates two VLAN interfaces, and assigns two switchports to the inside interface, and one to the outside interface.

```
interface Vlan11
nameif inside
security-level 100
ip address 10.11.11.1 255.255.255.0
no shutdown
!
interface Vlan20
nameif outside
security-level 0
ip address 10.20.20.1 255.255.255.0
no shutdown
!
interface Ethernet1/1
switchport
switchport access vlan 11
no shutdown
!
interface Ethernet1/2
switchport
switchport access vlan 20
no shutdown
!
interface Ethernet1/3
switchport
switchport access vlan 11
no shutdown
```

Transparent Mode Example

The following example creates two VLAN interfaces in bridge group 1, and assigns two switchports to the inside interface, and one to the outside interface.

```
firewall transparent
!
interface BVI1
```

```
ip address 10.20.20.1 255.255.255.0
!
interface Vlan11
bridge-group 1
nameif inside
security-level 100
no shutdown
!
interface Vlan20
bridge-group 1
nameif outside
security-level 0
no shutdown
!
interface Ethernet1/1
switchport
switchport access vlan 11
no shutdown
!
interface Ethernet1/2
switchport
switchport access vlan 20
no shutdown
!
interface Ethernet1/3
switchport
switchport access vlan 11
no shutdown
```

Mixed Firewall Interface/Switch Port Example

The following example creates one VLAN interface for the inside interface, and two firewall interfaces for the outside and dmz.

```
interface Vlan11
nameif inside
security-level 100
ip address 10.11.11.1 255.255.255.0
no shutdown
!
interface Ethernet1/1
switchport
switchport access vlan 11
no shutdown
!
interface Ethernet1/2
switchport
switchport access vlan 11
no shutdown
!
interface Ethernet1/3
switchport
switchport access vlan 11
no shutdown
!
interface Ethernet1/4
nameif outside
security-level 0
ip address 10.12.11.1 255.255.255.0
no shutdown
!
```

```
interface Ethernet1/5
nameif dmz
security-level 50
ip address 10.13.11.1 255.255.255.0
no shutdown
```

Integrated Routing and Bridging Example

The following example creates two bridge groups, with two VLAN interfaces (inside_1 and inside_2) in bridge group 1, and one (outside) in bridge group 2. A fourth VLAN interface is not part of a bridge group, and is a regular routed interface. Traffic between switch ports on the same VLAN are not subject to the ASA's security policy. But traffic between VLANs in a bridge group are subject to the security policy, so you may choose to layer bridge groups and switch ports to enforce the security policy between certain segments.

```
interface BVI1
nameif inside_bvi
security-level 100
ip address 10.30.1.10 255.255.255.0
!
interface BVI2
nameif outside_bvi
security-level 0
ip address 10.40.1.10 255.255.255.0
!
interface Vlan10
bridge-group 1
nameif inside_1
security-level 100
no shutdown
!
interface Vlan20
bridge-group 2
nameif outside
security-level 0
no shutdown
!
interface Vlan30
bridge-group 1
nameif inside_2
security-level 100
no shutdown
!
interface Vlan 100
nameif dmz
security-level 0
ip address 10.1.1.1 255.255.255.0
no shutdown
!
interface Ethernet1/1
switchport
switchport access vlan 10
no shutdown
!
interface Ethernet1/2
switchport
switchport access vlan 20
no shutdown
!
interface Ethernet1/3
switchport
```

```
switchport access vlan 30
no shutdown
!
interface Ethernet1/4
switchport
switchport access vlan 20
security-level 100
no shutdown
!
interface Ethernet1/5
switchport
switchport access vlan 100
no shutdown
!
interface Ethernet1/6
switchport
switchport access vlan 10
no shutdown
!
interface Ethernet1/7
switchport
switchport access vlan 30
no shutdown
!
interface Ethernet1/8
switchport
switchport access vlan 100
no shutdown
```

Failover Example

The following example configures Ethernet 1/3 as the failover interface.

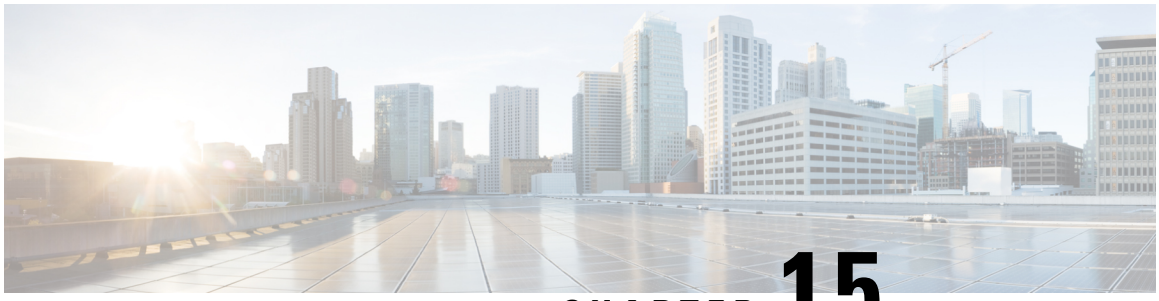
```
interface Vlan11
nameif inside
security-level 100
ip address 10.11.11.1 255.255.255.0 standby 10.11.11.2
no shutdown
!
interface Vlan20
nameif outside
security-level 0
ip address 10.20.20.1 255.255.255.0 standby 10.20.20.2
no shutdown
!
interface Ethernet1/1
switchport
switchport access vlan 11
no shutdown
!
interface Ethernet1/2
switchport
switchport access vlan 20
no shutdown
!
interface Ethernet1/3
description LAN/STATE Failover Interface
no shutdown
!
failover
failover lan unit primary
```

```
failover lan interface folink Ethernet1/3
failover replication http
failover link folink Ethernet1/3
failover interface ip folink 10.90.90.1 255.255.255.0 standby 10.90.90.2
```

History for switch ports

Table 24: History for Switch Ports

Feature Name	Version	Feature Information
Firepower 1010 hardware switch support	9.13(1)	The Firepower 1010 supports setting each Ethernet interface to be a switch port or a firewall interface. New/Modified commands: **forward interface, interface vlan, show switch mac-address-table, show switch vlan, switchport, switchport access vlan, switchport mode, switchport trunk allowed vlan**
Firepower 1010 PoE+ support on Ethernet 1/7 and Ethernet 1/8	9.13(1)	The Firepower 1010 supports Power over Ethernet+ (PoE+) on Ethernet 1/7 and Ethernet 1/8. New/Modified commands: **power inline, show power inline**



CHAPTER 15

EtherChannel Interfaces

This chapter tells how to configure EtherChannels interfaces.

Note For multiple context mode, complete all tasks in this section in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.

For ASA cluster interfaces, which have special requirements, see ASA Cluster for the Secure Firewall 3100, on page 333.

Note For Firepower 2100 in Platform mode and Firepower 4100/9300 chassis, EtherChannel interfaces are configured in the FXOS operating system. See the configuration or getting started guide for your chassis for more information.

- About EtherChannels, on page 625
- Guidelines for EtherChannels, on page 628
- Default Settings for EtherChannels Interfaces, on page 630
- Configure an EtherChannel, on page 631
- Monitoring EtherChannels, on page 634
- Examples for EtherChannels, on page 635
- History for EtherChannels, on page 636

About EtherChannels

This section describes EtherChannels.

About EtherChannels

An 802.3ad EtherChannel is a logical interface (called a port-channel interface) consisting of a bundle of individual Ethernet links (a channel group) so that you increase the bandwidth for a single network. A port channel interface is used in the same way as a physical interface when you configure interface-related features.

You can configure up to 48 EtherChannels, depending on how many interfaces your model supports.

Channel Group Interfaces

Each channel group can have up to 8 active interfaces, except for the ISA 3000, which supports 16 active interfaces. For switches that support only 8 active interfaces, you can assign up to 16 interfaces to a channel group: while only 8 interfaces can be active, the remaining interfaces can act as standby links in case of interface failure.

All interfaces in the channel group must be the same type and speed. The first interface added to the channel group determines the correct type and speed.

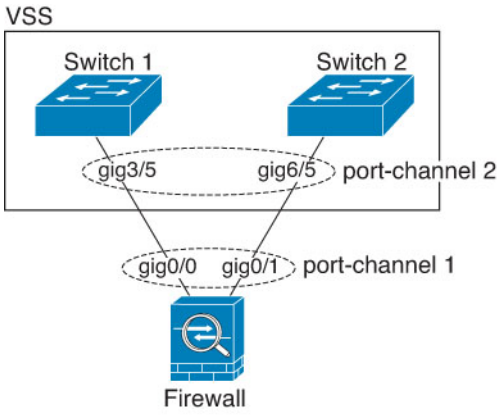
The EtherChannel aggregates the traffic across all the available active interfaces in the channel. The interface is selected using a proprietary hash algorithm, based on source or destination MAC addresses, IP addresses, TCP and UDP port numbers and VLAN numbers.

Connecting to an EtherChannel on Another Device

The device to which you connect the ASA EtherChannel must also support 802.3ad EtherChannels; for example, you can connect to the Catalyst 6500 switch or the Cisco Nexus 7000.

When the switch is part of a Virtual Switching System (VSS) or Virtual Port Channel (vPC), then you can connect ASA interfaces within the same EtherChannel to separate switches in the VSS/vPC. The switch interfaces are members of the same EtherChannel port-channel interface, because the separate switches act like a single switch.

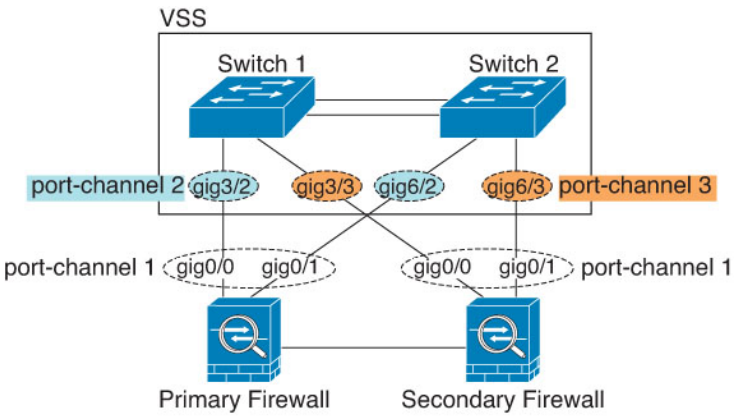
Figure 52: Connecting to a VSS/vPC



Note If the ASA device is in transparent firewall mode, and you place the ASA device between two sets of VSS/vPC switches, then be sure to disable Unidirectional Link Detection (UDLD) on any switch ports connected to the ASA device with an EtherChannel. If you enable UDLD, then a switch port may receive UDLD packets sourced from both switches in the other VSS/vPC pair. The receiving switch will place the receiving interface in a down state with the reason "UDLD Neighbor mismatch".

If you use the ASA device in an Active/Standby failover deployment, then you need to create separate EtherChannels on the switches in the VSS/vPC, one for each ASA device. On each ASA device, a single EtherChannel connects to both switches. Even if you could group all switch interfaces into a single EtherChannel connecting to both ASA devices (in this case, the EtherChannel will not be established because of the separate ASA system IDs), a single EtherChannel would not be desirable because you do not want traffic sent to the standby ASA device.

Figure 53: Active/Standby Failover and VSS/vPC



Link Aggregation Control Protocol

The Link Aggregation Control Protocol (LACP) aggregates interfaces by exchanging the Link Aggregation Control Protocol Data Units (LACPDUs) between two network devices.

You can configure each physical interface in an EtherChannel to be:

- Active—Sends and receives LACP updates. An active EtherChannel can establish connectivity with either an active or a passive EtherChannel. You should use the active mode unless you need to minimize the amount of LACP traffic.
- Passive—Receives LACP updates. A passive EtherChannel can only establish connectivity with an active EtherChannel. Not supported on hardware models.
- On—The EtherChannel is always on, and LACP is not used. An “on” EtherChannel can only establish a connection with another “on” EtherChannel.

LACP coordinates the automatic addition and deletion of links to the EtherChannel without user intervention. It also handles misconfigurations and checks that both ends of member interfaces are connected to the correct channel group. “On” mode cannot use standby interfaces in the channel group when an interface goes down, and the connectivity and configurations are not checked.

Load Balancing

The ASA device distributes packets to the interfaces in the EtherChannel by hashing the source and destination IP address of the packet (this criteria is configurable). The resulting hash is divided by the number of active links in a modulo operation where the resulting remainder determines which interface owns the flow. All packets with a *hash_value* **mod** *active_links* result of 0 go to the first interface in the EtherChannel, packets with a result of 1 go to the second interface, packets with a result of 2 go to the third interface, and so on. For example, if you have 15 active links, then the modulo operation provides values from 0 to 14. For 6 active links, the values are 0 to 5, and so on.

For a spanned EtherChannel in clustering, load balancing occurs on a per ASA basis. For example, if you have 32 active interfaces in the spanned EtherChannel across 8 ASAs, with 4 interfaces per ASA in the EtherChannel, then load balancing only occurs across the 4 interfaces on the ASA.

If an active interface goes down and is not replaced by a standby interface, then traffic is rebalanced between the remaining links. The failure is masked from both Spanning Tree at Layer 2 and the routing table at Layer 3, so the switchover is transparent to other network devices.

Related Topics

Customize the EtherChannel (ISA 3000), on page 633

EtherChannel MAC Address

All interfaces that are part of the channel group share the same MAC address. This feature makes the EtherChannel transparent to network applications and users, because they only see the one logical connection; they have no knowledge of the individual links.

Firepower and Secure Firewall Hardware

The port-channel interface uses the MAC address of the internal interface Internal-Data 0/1. Alternatively you can manually configure a MAC address for the port-channel interface. In multiple context mode, you can automatically assign unique MAC addresses to *shared* interfaces, including an EtherChannel port interface. All EtherChannel interfaces on a chassis use the same MAC address, so be aware that if you use SNMP polling, for example, multiple interfaces will have the same MAC address.



Note Member interfaces only use the Internal-Data 0/1 MAC address after a reboot. Prior to rebooting, the member interface uses its own MAC address. If you add a new member interface after a reboot, you will have to perform another reboot to update its MAC address.

Guidelines for EtherChannels

Bridge Group

In routed mode, ASA-defined EtherChannels are not supported as bridge group members. EtherChannels on the Firepower 4100/9300 can be bridge group members.

Failover

- When you use an EtherChannel interface as a Failover link, it must be pre-configured on both units in the Failover pair; you cannot configure it on the primary unit and expect it to replicate to the secondary unit because *the Failover link itself is required for replication*.
- If you use an EtherChannel interface for the state link, no special configuration is required; the configuration can replicate from the primary unit as normal. For the Firepower 4100/9300 chassis, all interfaces, including EtherChannels, need to be pre-configured on both units.
- You can monitor EtherChannel interfaces for Failover using the **monitor-interface** command. When an active member interface fails over to a standby interface, this activity does not cause the EtherChannel interface to appear to be failed when being monitored for device-level Failover. Only when all physical interfaces fail does the EtherChannel interface appear to be failed (for an EtherChannel interface, the number of member interfaces allowed to fail is configurable).

- If you use an EtherChannel interface for a Failover or state link, then to prevent out-of-order packets, only one interface in the EtherChannel is used. If that interface fails, then the next interface in the EtherChannel is used. You cannot alter the EtherChannel configuration while it is in use as a Failover link. To alter the configuration, you need to temporarily disable Failover, which prevents Failover from occurring for the duration.

Model Support

- You cannot add EtherChannels in ASA for the Firepower 2100 in platform mode, Firepower 4100/9300, or the ASAv. The Firepower 4100/9300 supports EtherChannels, but you must perform all hardware configuration of EtherChannels in FXOS on the chassis.
- You cannot use Firepower 1010 switch ports or VLAN interfaces in EtherChannels.

Clustering

- When you use the EtherChannel interface as the Cluster Control Link, it must be pre-configured on all units in the cluster; you cannot configure it on the primary unit and expect it to replicate to member units because *the Cluster Control Link itself is required for replication*.
- To configure a spanned EtherChannel or an individual cluster interface, see the clustering chapter.

General EtherChannel Guidelines

- You can configure up to 48 EtherChannels, depending on how many interfaces are available on your model.
- Each channel group can have up to 8 active interfaces, except for the ISA 3000, which supports 16 active interfaces. For switches that support only 8 active interfaces, you can assign up to 16 interfaces to a channel group: while only 8 interfaces can be active, the remaining interfaces can act as standby links in case of interface failure.
- When you add the first member interface, it sets the required hardware properties of all member interfaces.
  - The media type of member interfaces can be either RJ-45 or SFP; SFPs of different types (copper and fiber) can be mixed. You cannot mix RJ-45 and SFP interfaces.
  - All interfaces must be set to the same speed and duplex.
  - The first interface sets the speed *capacity*, which cannot be changed later.
    - For SFP Detect interfaces—You can include interfaces with different speed capacities as long as they have a common speed. When you set the speed to SFP Detect (the default), the speed will be dynamically set to the highest common speed. If you later change the member interfaces so that the common speed is now higher, the EtherChannel speed will also be higher automatically.

      You can set a specific speed, but only speeds that are available on the first member interface. For example, if your first interface is 1/10GB, then the available speeds for the EtherChannel will be 1GB, 10GB, and SFP Detect. If you later remove the 1/10GB interfaces and replace them with 1/10/25GB interfaces, you cannot manually set the speed to 25GB. In this case, you can use SFP Detect to use the 25GB speed.
    - For non-SFP Detect interfaces—All additional interfaces must have the same speed capacity. For example, if your first interface speed capacity is 10MB/100MB/1GB, you must add other

10MB/100MB/1GB interfaces. You can set the EtherChannel (and its member interfaces) to any of those speeds. You cannot later add 1/10GB interfaces to the EtherChannel, even if you remove the lower capacity interfaces. You also cannot mix interface capacities (for example 1GB and 10GB interfaces) by setting the speed to be lower on the larger-capacity interface.

- The device to which you connect the ASA EtherChannel must also support 802.3ad EtherChannels.
- The ASA device does not support LACPDUs that are VLAN-tagged. If you enable native VLAN tagging on the neighboring switch using the Cisco IOS **vlan dot1Q tag native** command, then the ASA device will drop the tagged LACPDUs. Be sure to disable native VLAN tagging on the neighboring switch. In multiple context mode, these messages are not included in a packet capture, so that you cannot diagnose the issue easily.
- The LACP rate depends on the model. When you set the rate (normal or fast), the device requests that rate from the connecting switch. In return, the device will send at the rate requested by the connecting switch. We recommend that you set the same rate on both sides.
  - Firepower 4100/9300—The LACP rate is set to fast by default in FXOS, but you can configure it as normal (also known as slow).
  - Secure Firewall 3100—The LACP rate is set to normal (slow) by default, but you can configure it as fast on the device.
  - All other models—The LACP rate set to normal (also known as slow), and it is not configurable, which means the device will always request a slow rate from the connecting switch. We recommend setting the rate on the switch to slow, so both sides send LACP messages at the same rate.
- In Cisco IOS software versions earlier than 15.1(1)S2, ASA did not support connecting an EtherChannel to a switch stack. With default switch settings, if the ASA EtherChannel is connected cross stack, and if the primary switch is powered down, then the EtherChannel connected to the remaining switch will not come up. To improve compatibility, set the **stack-mac persistent timer** command to a large enough value to account for reload time; for example, 8 minutes or 0 for indefinite. Or, you can upgrade to more a more stable switch software version, such as 15.1(1)S2.
- All the ASA configuration refers to the logical EtherChannel interface instead of the member physical interfaces.

Default Settings for EtherChannels Interfaces

This section lists default settings for interfaces if you do not have a factory default configuration.

Default State of Interfaces

The default state of an interface depends on the type and the context mode.

In multiple context mode, all allocated interfaces are enabled by default, no matter what the state of the interface is in the system execution space. However, for traffic to pass through the interface, the interface also has to be enabled in the system execution space. If you shut down an interface in the system execution space, then that interface is down in all contexts that share it.

In single mode or in the system execution space, interfaces have the following default states:

- Physical interfaces—Disabled.

- EtherChannel port-channel interfaces—Enabled. However, for traffic to pass through the EtherChannel, the channel group physical interfaces must also be enabled.

Configure an EtherChannel

This section describes how to create an EtherChannel port-channel interface, assign interfaces to the EtherChannel, and customize the EtherChannel.

Add Interfaces to the EtherChannel

This section describes how to create an EtherChannel port-channel interface and assign interfaces to the EtherChannel. By default, port-channel interfaces are enabled.

Before you begin

- When you add the first member interface, it sets the required hardware properties of all member interfaces. See Guidelines for EtherChannels, on page 628 for more details about member interface requirements.
- To configure a spanned EtherChannel for clustering, see the clustering chapter instead of this procedure.
- You cannot add a physical interface to the channel group if you configured a name for it. You must first remove the name using the **no nameif** command.
- For multiple context mode, complete this procedure in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.

Caution If you are using a physical interface already in your configuration, removing the name will clear any configuration that refers to the interface.

Procedure

Step 1 Specify the interface you want to add to the channel group:

interface *physical_interface*

Example:

```
ciscoasa(config)# interface gigabitethernet 0/0
```

The *physical_interface* ID includes the type, slot, and port number as type[slot/]port. This first interface in the channel group determines the type and speed for all other interfaces in the group.

In transparent mode, if you create a channel group with multiple Management interfaces, then you can use this EtherChannel as the management-only interface.

Step 2 Assign this physical interface to an EtherChannel:

channel-group *channel_id* **mode** {**active** | **passive** | **on**}

Example:

```
ciscoasa(config-if)# channel-group 1 mode active
```

The *channel_id* is an integer between 1 and 48 (1 and 8 for the Firepower 1010). If the port-channel interface for this channel ID does not yet exist in the configuration, one will be added:

interface port-channel *channel_id*

We recommend using **active** mode. **passive** mode is only available for the ISA 3000 only.

Step 3 (Optional) (Secure Firewall 3100 only) Set the LACP data unit receive rate for a physical interface in the channel group.

lacp rate {**normal** | **fast**}

The default is **normal** (slow, every 30 seconds). The **fast** option receives LACP data units every second. You should match the setting on the connected switch.

Step 4 (Optional) (ISA 3000 models only) Set the priority for a physical interface in the channel group.

lacp port-priority *number*

Example:

```
ciscoasa(config-if)# lacp port-priority 12345
```

The priority *number* is an integer between 1 and 65535. The default is 32768. The higher the number, the lower the priority. The ASA uses this setting to decide which interfaces are active and which are standby if you assign more interfaces than can be used. If the port priority setting is the same for all interfaces, then the priority is determined by the interface ID (slot/port). The lowest interface ID is the highest priority. For example, GigabitEthernet 0/0 is a higher priority than GigabitEthernet 0/1.

If you want to prioritize an interface to be active even though it has a higher interface ID, then set this command to have a lower value. For example, to make GigabitEthernet 1/3 active before GigabitEthernet 0/7, then make the **lacp port-priority** value be 12345 on the 1/3 interface vs. the default 32768 on the 0/7 interface.

If the device at the other end of the EtherChannel has conflicting port priorities, the system priority is used to determine which port priorities to use. See the **lacp system-priority** command.

Step 5 (Optional) Set the Ethernet properties for the port-channel interface to override the properties set on the individual interfaces.

interface port-channel *channel_id*

See Enable the Physical Interface and Configure Ethernet Parameters, on page 597 for Ethernet commands. This method provides a shortcut to set these parameters because these parameters must match for all interfaces in the channel group.

Step 6 Repeat Steps 1 through 3 for each interface you want to add to the channel group.

Each interface in the channel group must be the same type and speed. Half duplex is not supported. If you add an interface that does not match, it will be placed in a suspended state.

Step 7 (Optional) (Secure Firewall 3100 only) Set the LACP data unit receive rate for a physical interface in the channel group.

a) Click the physical interface in the **Interfaces** table, and click **Edit**.

The **Edit Interface** dialog box appears.

b) Click the **Advanced** tab.
c) In the **EtherChannel** area, from the **Rate** drop down list, choose **Normal** or **Fast**.

The default is **Normal** (slow, every 30 seconds). The **Fast** option received LACP updates every second. You should match the setting on the connected switch.

Related Topics

Link Aggregation Control Protocol, on page 627
Customize the EtherChannel (ISA 3000), on page 633

Customize the EtherChannel (ISA 3000)

This section describes how to set the maximum number of interfaces in the EtherChannel, the minimum number of operating interfaces for the EtherChannel to be active, the load balancing algorithm, and other optional parameters. These parameters only apply to the ISA 3000.

Procedure

Step 1 Specify the port-channel interface:

interface port-channel *channel_id*

Example:

```
ciscoasa(config)# interface port-channel 1
```

This interface was created automatically when you added an interface to the channel group. If you have not yet added an interface, then this command creates the port-channel interface.

You need to add at least one member interface to the port-channel interface before you can configure logical parameters for it such as a name.

Step 2 Specify the maximum number of active interfaces allowed in the channel group:

lacp max-bundle *number*

Example:

```
ciscoasa(config-if)# lacp max-bundle 6
```

The *number* is between 1 and 16. The default is 16. If your switch does not support 16 active interfaces, be sure to set this command to 8 or fewer.

Step 3 Specify the minimum number of active interfaces required for the port-channel interface to become active:

port-channel min-bundle *number*

Example:

```
ciscoasa(config-if)# port-channel min-bundle 2
```

The *number* is between 1 and 16. The default is 1. If the active interfaces in the channel group falls below this value, then the port-channel interface goes down, and could trigger a device-level failover.

Step 4 Configure the load-balancing algorithm:

port-channel load-balance {**dst-ip** | **dst-ip-port** | **dst-mac** | **dst-port** | **src-dst-ip** | **src-dst-ip-port** | **src-dst-mac** | **src-dst-port** | **src-ip** | **src-ip-port** | **src-mac** | **src-port** | **vlan-dst-ip** | **vlan-dst-ip-port** | **vlan-only** | **vlan-src-dst-ip** | **vlan-src-dst-ip-port** | **vlan-src-ip** | **vlan-src-ip-port**}

Example:

```
ciscoasa(config-if)# port-channel load-balance src-dst-mac
```

By default, the ASA balances the packet load on interfaces according to the source and destination IP address (**src-dst-ip**) of the packet. If you want to change the properties on which the packet is categorized, use this command. For example, if your traffic is biased heavily towards the same source and destination IP addresses, then the traffic assignment to interfaces in the EtherChannel will be unbalanced. Changing to a different algorithm can result in more evenly distributed traffic.

Step 5 Set the LACP system priority:

lacp system-priority *number*

Example:

```
ciscoasa(config)# lacp system-priority 12345
```

The *number* is between 1 and 65535. The default is 32768. The higher the number, the lower the priority. This command is global for the ASA.

If the device at the other end of the EtherChannel has conflicting port priorities, the system priority is used to determine which port priorities to use. For interface priorities within an EtherChannel, see the **lacp port-priority** command.

Related Topics

Load Balancing, on page 627
Add Interfaces to the EtherChannel, on page 631

Monitoring EtherChannels

See the following commands:

Note For Firepower and Secure Firewall models, some statistics are not shown using the ASA commands. You must view more detailed interface statistics using FXOS commands.

- /eth-uplink/fabric# **show interface**
- /eth-uplink/fabric# **show port-channel**
- /eth-uplink/fabric/interface# **show stats**

For the Firepower 2100 in Platform mode, see also the following FXOS connect local-mgmt commands:

- (local-mgmt)# **show portmanager counters**
- (local-mgmt)# **show lacp**
- (local-mgmt)# **show portchannel**

See the FXOS troubleshooting guide for more information.

- **show interface**

  Displays interface statistics.

- **show interface ip brief**

  Displays interface IP addresses and status.

- (ISA 3000 only) **show lacp** {[*channel_group_number*] {**counters** | **internal** | **neighbor**} | **sys-id**}

  For EtherChannel, displays LACP information such as traffic statistics, system identifier and neighbor details.

- (ISA 3000 only) **show port-channel** [*channel_group_number*] [**brief** | **detail** | **port** | **protocol** | **summary**]

  For EtherChannel, displays EtherChannel information in a detailed and one-line summary form. This command also displays the port and port-channel information.

- (ISA 3000 only) **show port-channel** *channel_group_number* **load-balance** [**hash-result** {**ip** | **ipv6** | **l4port** | **mac** | **mixed** | **vlan-only**} *parameters*]

  For EtherChannel, displays port-channel load-balance information along with the hash result and member interface selected for a given set of parameters.

Examples for EtherChannels

The following example configures three interfaces as part of an EtherChannel. It also sets the system priority to be a higher priority, and GigabitEthernet 0/2 to be a higher priority than the other interfaces in case more than eight interfaces are assigned to the EtherChannel.

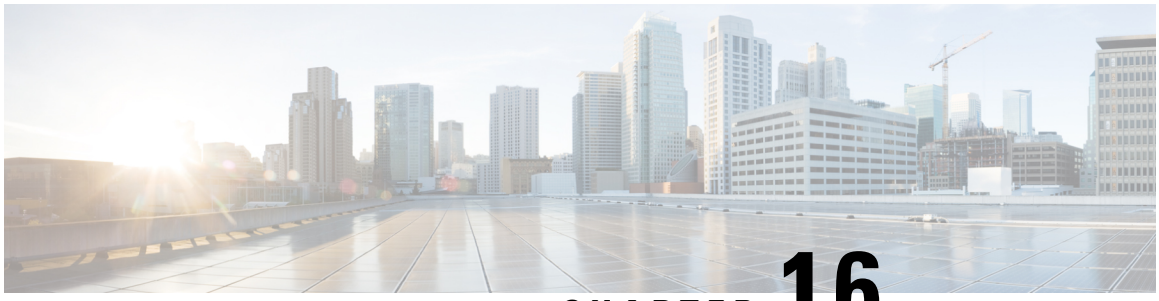
```
lacp system-priority 1234
interface GigabitEthernet0/0
  channel-group 1 mode active
interface GigabitEthernet0/1
  channel-group 1 mode active
```

```
interface GigabitEthernet0/2
  lacp port-priority 1234
  channel-group 1 mode passive
interface Port-channel1
  lacp max-bundle 4
  port-channel min-bundle 2
  port-channel load-balance dst-ip
```

History for EtherChannels

Table 25: History for EtherChannels

Feature Name	Releases	Feature Information
EtherChannel support	8.4(1)	You can configure up to 48 802.3ad EtherChannels of eight active interfaces each. We introduced the following commands: **channel-group**, **lacp port-priority**, **interface port-channel**, **lacp max-bundle**, **port-channel min-bundle**, **port-channel load-balance**, **lacp system-priority**, **clear lacp counters**, **show lacp**, **show port-channel**. **Note** EtherChannel is not supported on the ASA 5505.
Support for 16 active links in an EtherChannel	9.2(1)	You can now configure up to 16 active links in an EtherChannel. Previously, you could have 8 active links and 8 standby links. Be sure that your switch can support 16 active links (for example the Cisco Nexus 7000 with F2-Series 10 Gigabit Ethernet Module). **Note** If you upgrade from an earlier ASA version, the maximum active interfaces is set to 8 for compatibility purposes (the **lacp max-bundle** command). We modified the following commands: **lacp max-bundle** and **port-channel min-bundle**.



CHAPTER 16

VLAN Subinterfaces

This chapter tells how to configure VLAN subinterfaces.

Note For multiple context mode, complete all tasks in this section in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.

- About VLAN Subinterfaces, on page 637
- Licensing for VLAN Subinterfaces, on page 637
- Guidelines and Limitations for VLAN Subinterfaces, on page 638
- Default Settings for VLAN Subinterfaces, on page 639
- Configure VLAN Subinterfaces and 802.1Q Trunking, on page 639
- Monitoring VLAN Subinterfaces, on page 641
- Examples for VLAN Subinterfaces, on page 641
- History for VLAN subinterfaces, on page 642

About VLAN Subinterfaces

VLAN subinterfaces let you divide a physical or EtherChannel interface into multiple logical interfaces that are tagged with different VLAN IDs. An interface with one or more VLAN subinterfaces is automatically configured as an 802.1Q trunk. Because VLANs allow you to keep traffic separate on a given physical interface, you can increase the number of interfaces available to your network without adding additional physical interfaces or ASAs. This feature is particularly useful in multiple context mode so that you can assign unique interfaces to each context.

You can configure a primary VLAN, as well as one or more secondary VLANs. When the ASA receives traffic on the secondary VLANs, it maps it to the primary VLAN.

Licensing for VLAN Subinterfaces

Model	License Requirement
Firepower 1010	Standard License: 60

Model	License Requirement
Firepower 1120	Standard License: 512
Firepower 1140, 1150	Standard License: 1024
Firepower 2100	Standard License: 1024
Secure Firewall 3100	Standard License: 1024
Firepower 4100	Standard License: 1024
Firepower 9300	Standard License: 1024
ASAv	Throughput capability: 100 Mbps: 25 1 Gbps: 50 2 Gbps: 200 10 Gbps: 1024
ISA 3000	Standard License: 5 Security Plus License: 100



Note For an interface to count against the VLAN limit, you must assign a VLAN to it. For example:

```
interface gigabitethernet 0/0.100
  vlan 100
```

Guidelines and Limitations for VLAN Subinterfaces

Model Support

- 1010—VLAN subinterfaces are not supported on switch ports or VLAN interfaces.
- For ASA models, you cannot configure subinterfaces on the Management interface. See Management Slot/Port Interface, on page 594 for subinterface support.

Additional Guidelines

- Preventing untagged packets on the physical interface—If you use subinterfaces, you typically do not also want the physical interface to pass traffic, because the physical interface passes untagged packets. This property is also true for the active physical interface for EtherChannel links. Because the physical or EtherChannel interface must be enabled for the subinterface to pass traffic, ensure that the physical or EtherChannel interface does not pass traffic by leaving out the **nameif** command. If you want to let

the physical or EtherChannel interface pass untagged packets, you can configure the **nameif** command as usual.

- All subinterfaces on the same parent interface must be either bridge group members or routed interfaces; you cannot mix and match.
- The ASA does not support the Dynamic Trunking Protocol (DTP), so you must configure the connected switch port to trunk unconditionally.
- You might want to assign unique MAC addresses to subinterfaces defined on the ASA, because they use the same burned-in MAC address of the parent interface. For example, your service provider might perform access control based on the MAC address. Also, because IPv6 link-local addresses are generated based on the MAC address, assigning unique MAC addresses to subinterfaces allows for unique IPv6 link-local addresses, which can avoid traffic disruption in certain instances on the ASA. You can automatically generate unique MAC addresses; see Automatically Assign MAC Addresses, on page 712.



Note If you manually assign a MAC address, be sure to assign MAC addresses to all subinterfaces on the same physical interface to avoid unexpected behavior and outages.

Default Settings for VLAN Subinterfaces

This section lists default settings for interfaces if you do not have a factory default configuration.

Default State of Interfaces

The default state of an interface depends on the type and the context mode.

In multiple context mode, all allocated interfaces are enabled by default, no matter what the state of the interface is in the system execution space. However, for traffic to pass through the interface, the interface also has to be enabled in the system execution space. If you shut down an interface in the system execution space, then that interface is down in all contexts that share it.

In single mode or in the system execution space, interfaces have the following default states:

- Physical interfaces—Disabled.
- VLAN subinterfaces—Enabled. However, for traffic to pass through the subinterface, the physical interface must also be enabled.

Configure VLAN Subinterfaces and 802.1Q Trunking

Add a VLAN subinterface to a physical or EtherChannel interface.

Before you begin

For multiple context mode, complete this procedure in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.

Procedure

Step 1 Specify the new subinterface:

interface {*physical_interface* | **port-channel** *number*}.*subinterface*

Example:

```
ciscoasa(config)# interface gigabitethernet 0/1.100
```

The **port-channel** *number* argument is the EtherChannel interface ID, such as **port-channel 1**.

The *subinterface* ID is an integer between 1 and 4294967293.

Step 2 Specify the VLAN for the subinterface:

vlan *vlan_id* [**secondary** *vlan_range*]

Example:

```
ciscoasa(config-subif)# vlan 101 secondary 52 64,66-74
```

The *vlan_id* is an integer between 1 and 4094. Some VLAN IDs might be reserved on connected switches, so check the switch documentation for more information.

The secondary VLANs can be separated by spaces, commas, and dashes (for a contiguous range). When the ASA receives traffic on the secondary VLANs, it maps the traffic to the primary VLAN.

You cannot assign the same VLAN to multiple subinterfaces. You cannot assign a VLAN to the physical interface. Each subinterface must have a VLAN ID before it can pass traffic. To change a VLAN ID, you do not need to remove the old VLAN ID with the **no** option; you can enter the **vlan** command with a different VLAN ID, and the ASA changes the old ID. To remove some secondary VLANs from the list, you can use the **no** command and only list the VLANs to remove. You can only selectively remove listed VLANs; you cannot remove a single VLAN in a range, for example.

Examples

The following example maps a set of secondary VLANs to VLAN 200:

```
interface gigabitethernet 0/6.200
   vlan 200 secondary 500 503 600-700
```

The following example removes secondary VLAN 503 from the list:

```
no vlan 200 secondary 503
show running-config interface gigabitethernet0/6.200
!
interface GigabitEthernet0/6.200
 vlan 200 secondary 500 600-700
 no nameif
 no security-level
```

```
no ip address
```

Related Topics

Licensing for VLAN Subinterfaces, on page 637

Monitoring VLAN Subinterfaces

See the following commands:

- **show interface**

  Displays interface statistics.

- **show interface ip brief**

  Displays interface IP addresses and status.

- **show vlan mapping**

  Shows the interface, secondary VLANs, and the primary VLANs to which they are mapped.

Examples for VLAN Subinterfaces

The following example configures parameters for a subinterface in single mode:

```
interface gigabitethernet 0/1
  no nameif
  no security-level
  no ip address
  no shutdown
interface gigabitethernet 0/1.1
  vlan 101
  nameif inside
  security-level 100
  ip address 192.168.6.6 255.255.255.0
  no shutdown
```

The following example shows how VLAN mapping works with the Catalyst 6500. Consult the Catalyst 6500 configuration guide on how to connect nodes to PVLANS.

ASA Configuration

```
interface GigabitEthernet1/1
  description Connected to Switch GigabitEthernet1/5
  no nameif
  no security-level
  no ip address
  no shutdown
!
interface GigabitEthernet1/1.70
  vlan 70 secondary 71 72
  nameif vlan_map1
  security-level 50
  ip address 10.11.1.2 255.255.255.0
```

```
  no shutdown
!
interface GigabitEthernet1/2
  nameif outside
  security-level 0
  ip address 172.16.171.31 255.255.255.0
  no shutdown

Catalyst 6500 Configuration

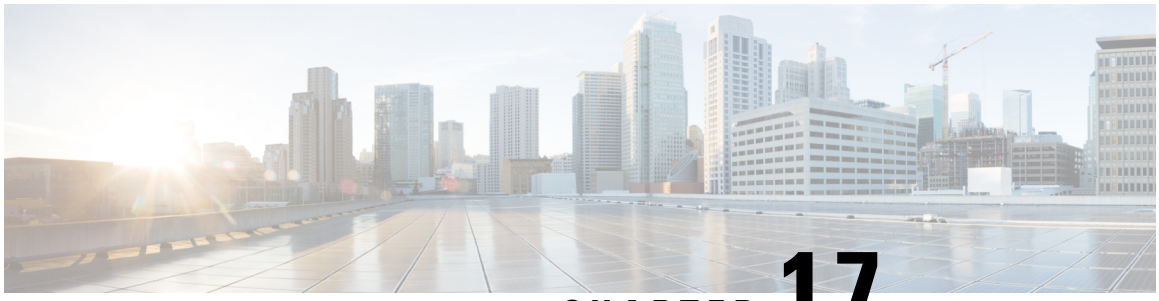
vlan 70
  private-vlan primary
  private-vlan association 71-72
!
vlan 71
  private-vlan community
!
vlan 72
  private-vlan isolated
!
interface GigabitEthernet1/5
  description Connected to ASA GigabitEthernet1/1
  switchport
  switchport trunk encapsulation dot1q
  switchport trunk allowed vlan 70-72
  switchport mode trunk
!
```

History for VLAN subinterfaces

Table 26: History for VLAN Subinterfaces

Feature Name	Version	Feature Information
Increased VLANs	7.0(5)	Increased the following limits: • ASA5510 Base license VLANs from 0 to 10. • ASA5510 Security Plus license VLANs from 10 to 25. • ASA5520 VLANs from 25 to 100. • ASA5540 VLANs from 100 to 200.
Increased VLANs	7.2(2)	VLAN limits were increased for the ASA 5510 (from 10 to 50 for the Base license, and from 25 to 100 for the Security Plus license), the ASA 5520 (from 100 to 150), the ASA 5550 (from 200 to 250).
Increased VLANs for the ASA 5580	8.1(2)	The number of VLANs supported on the ASA 5580 are increased from 100 to 250.
Support to map a Secondary VLANs to a Primary VLAN	9.5(2)	You can now configure one or more secondary VLANs for a subinterface. When the ASA receives traffic on the secondary VLANs, it maps it to the primary VLAN. We introduced or modified the following commands: **vlan secondary, show vlan mapping**

Feature Name	Version	Feature Information
Increased VLANs for the ISA 3000	9.13(1)	The maximum VLANs for the ISA 3000 with the Security Plus license increased from 25 to 100.



CHAPTER 17

VXLAN Interfaces

This chapter tells how to configure Virtual eXtensible LAN (VXLAN) interfaces. VXLANs act as Layer 2 virtual networks over Layer 3 physical networks to stretch Layer 2 networks.

- About VXLAN Interfaces, on page 645
- Requirements and Prerequisites for VXLAN Interfaces, on page 651
- Guidelines for VXLAN Interfaces, on page 652
- Default Settings for VXLAN Interfaces, on page 652
- Configure VXLAN Interfaces, on page 653
- Configure Geneve Interfaces, on page 657
- Allow Gateway Load Balancer Health Checks, on page 660
- Monitoring VXLAN Interfaces, on page 661
- Examples for VXLAN Interfaces, on page 664
- History for VXLAN Interfaces, on page 667

About VXLAN Interfaces

VXLAN provides the same Ethernet Layer 2 network services as VLAN does, but with greater extensibility and flexibility. Compared to VLAN, VXLAN offers the following benefits:

- Flexible placement of multitenant segments throughout the data center.
- Higher scalability to address more Layer 2 segments: up to 16 million VXLAN segments.

This section describes how VXLAN works. For detailed information about VXLAN, see RFC 7348. For detailed information about Geneve, see RFC 8926.

Encapsulation

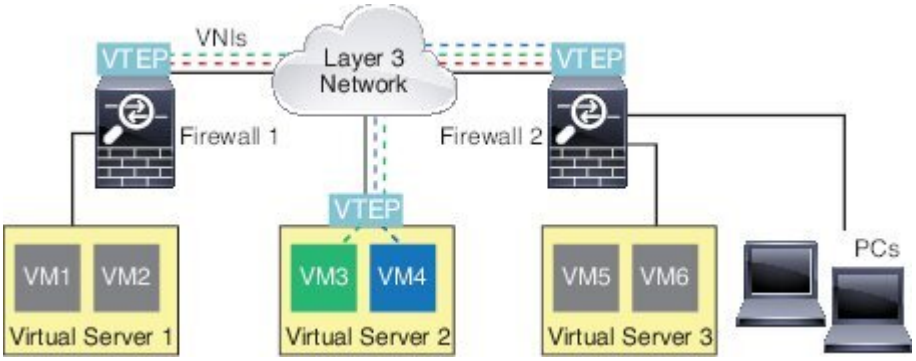
The ASA supports two types of VXLAN encapsulation:

- VXLAN (all models)—VXLAN uses MAC Address-in-User Datagram Protocol (MAC-in-UDP) encapsulation. The original Layer 2 frame has a VXLAN header added and is then placed in a UDP-IP packet.
- Geneve (ASAv only)—Geneve has a flexible inner header that is not limited to the MAC address. Geneve encapsulation is required for transparent routing of packets between an Amazon Web Services (AWS) Gateway Load Balancer and appliances, and for sending extra information.

VXLAN Tunnel Endpoint

VXLAN tunnel endpoint (VTEP) devices perform VXLAN encapsulation and decapsulation. Each VTEP has two interface types: one or more virtual interfaces called VXLAN Network Identifier (VNI) interfaces to which you apply your security policy, and a regular interface called the VTEP source interface that tunnels the VNI interfaces between VTEPs. The VTEP source interface is attached to the transport IP network for VTEP-to-VTEP communication.

The following figure shows two ASAs and Virtual Server 2 acting as VTEPs across a Layer 3 network, extending the VNI 1, 2, and 3 networks between sites. The ASAs act as bridges or gateways between VXLAN and non-VXLAN networks.



The underlying IP network between VTEPs is independent of the VXLAN overlay. Encapsulated packets are routed based on the outer IP address header, which has the initiating VTEP as the source IP address and the terminating VTEP as the destination IP address. For VXLAN encapsulation: The destination IP address can be a multicast group when the remote VTEP is not known. With Geneve, the ASA only supports static peers. The destination port for VXLAN is UDP port 4789 by default (user configurable). The destination port for Geneve is 6081.

VTEP Source Interface

The VTEP source interface is a regular ASA interface (physical, EtherChannel, or even VLAN) with which you plan to associate all VNI interfaces. You can configure one VTEP source interface per ASA/security context. Because you can only configure one VTEP source interface, you cannot configure both VXLAN and Geneve interfaces on the same device.

The VTEP source interface can be devoted wholly to VXLAN traffic, although it is not restricted to that use. If desired, you can use the interface for regular traffic and apply a security policy to the interface for that traffic. For VXLAN traffic, however, all security policy must be applied to the VNI interfaces. The VTEP interface serves as a physical port only.

In transparent firewall mode, the VTEP source interface is not part of a BVI, and you do configure an IP address for it, similar to the way the management interface is treated.

VNI Interfaces

VNI interfaces are similar to VLAN interfaces: they are virtual interfaces that keep network traffic separated on a given physical interface by using tagging. You apply your security policy directly to each VNI interface.

You can only add one VTEP interface, and all VNI interfaces are associated with the same VTEP interface.

VXLAN Packet Processing

VXLAN

Traffic entering and exiting the VTEP source interface is subject to VXLAN processing, specifically encapsulation or decapsulation.

Encapsulation processing includes the following tasks:

- The VTEP source interface encapsulates the inner MAC frame with the VXLAN header.
- The UDP checksum field is set to zero.
- The Outer frame source IP is set to the VTEP interface IP.
- The Outer frame destination IP is decided by a remote VTEP IP lookup.

Decapsulation; the ASA only decapsulates a VXLAN packet if:

- It is a UDP packet with the destination port set to 4789 (this value is user configurable).
- The ingress interface is the VTEP source interface.
- The ingress interface IP address is the same as the destination IP address.
- The VXLAN packet format is compliant with the standard.

Geneve

Traffic entering and exiting the VTEP source interface is subject to Geneve processing, specifically encapsulation or decapsulation.

Encapsulation processing includes the following tasks:

- The VTEP source interface encapsulates the inner MAC frame with the Geneve header.
- The UDP checksum field is set to zero.
- The Outer frame source IP is set to the VTEP interface IP.
- The Outer frame destination IP is set the peer IP address that you configured.

Decapsulation; the ASA only decapsulates a Geneve packet if:

- It is a UDP packet with the destination port set to 6081 (this value is user configurable).
- The ingress interface is the VTEP source interface.
- The ingress interface IP address is the same as the destination IP address.
- The Geneve packet format is compliant with the standard.

Peer VTEP

When the ASA sends a packet to a device behind a peer VTEP, the ASA needs two important pieces of information:

- The destination MAC address of the remote device

- The destination IP address of the peer VTEP

The ASA maintains a mapping of destination MAC addresses to remote VTEP IP addresses for the VNI interfaces.

VXLAN Peer

There are two ways in which the ASA can find this information:

- A single peer VTEP IP address can be statically configured on the ASA.

  You cannot manually define multiple peers.

  The ASA then sends a VXLAN-encapsulated ARP broadcast to the VTEP to learn the end node MAC address.

- A multicast group can be configured on each VNI interface (or on the VTEP as a whole).



Note This option is not supported with Geneve.

The ASA sends a VXLAN-encapsulated ARP broadcast packet within an IP multicast packet through the VTEP source interface. The response to this ARP request enables the ASA to learn both the remote VTEP IP address along with the destination MAC address of the remote end node.

Geneve Peer

The ASAv only supports statically defined peers. You can define the ASAv peer IP address on the AWS Gateway Load Balancer. Because the ASAv never initiates traffic to the Gateway Load Balancer, you do not also have to specify the Gateway Load Balancer IP address on the ASAv; it learns the peer IP address when it receives Geneve traffic. Multicast groups are not supported with Geneve.

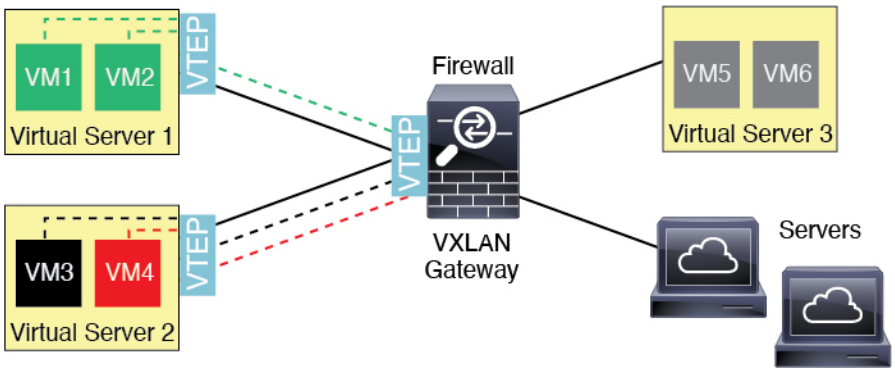
VXLAN Use Cases

This section describes the use cases for implementing VXLAN on the ASA.

VXLAN Bridge or Gateway Overview

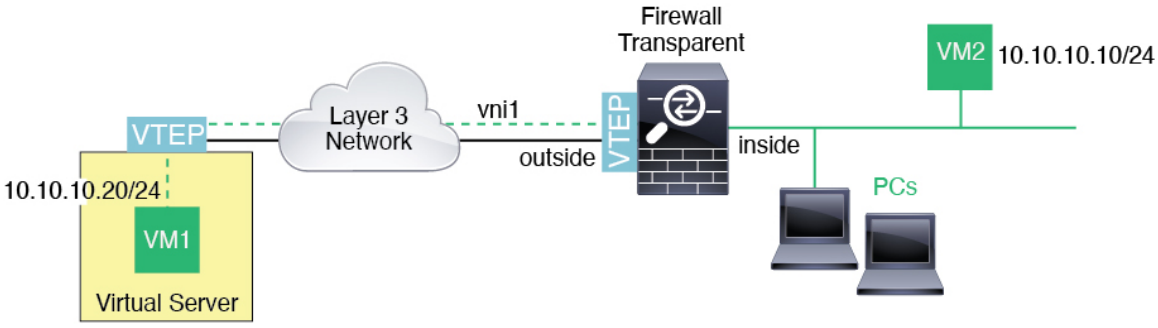
Each ASA VTEP acts as a bridge or gateway between end nodes such as VMs, servers, and PCs and the VXLAN overlay network. For incoming frames received with VXLAN encapsulation over the VTEP source interface, the ASA strips out the VXLAN header and forwards it to a physical interface connected to a non-VXLAN network based on the destination MAC address of the inner Ethernet frame.

The ASA always processes VXLAN packets; it does not just forward VXLAN packets untouched between two other VTEPs.



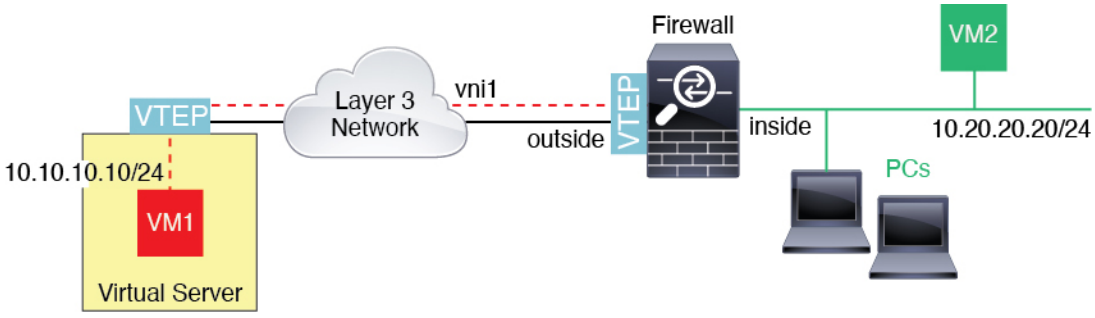
VXLAN Bridge

When you use a bridge group (transparent firewall mode, or optionally routed mode), the ASA can serve as a VXLAN bridge between a (remote) VXLAN segment and a local segment where both are in the same network. In this case, one member of the bridge group is a regular interface while the other member is a VNI interface.



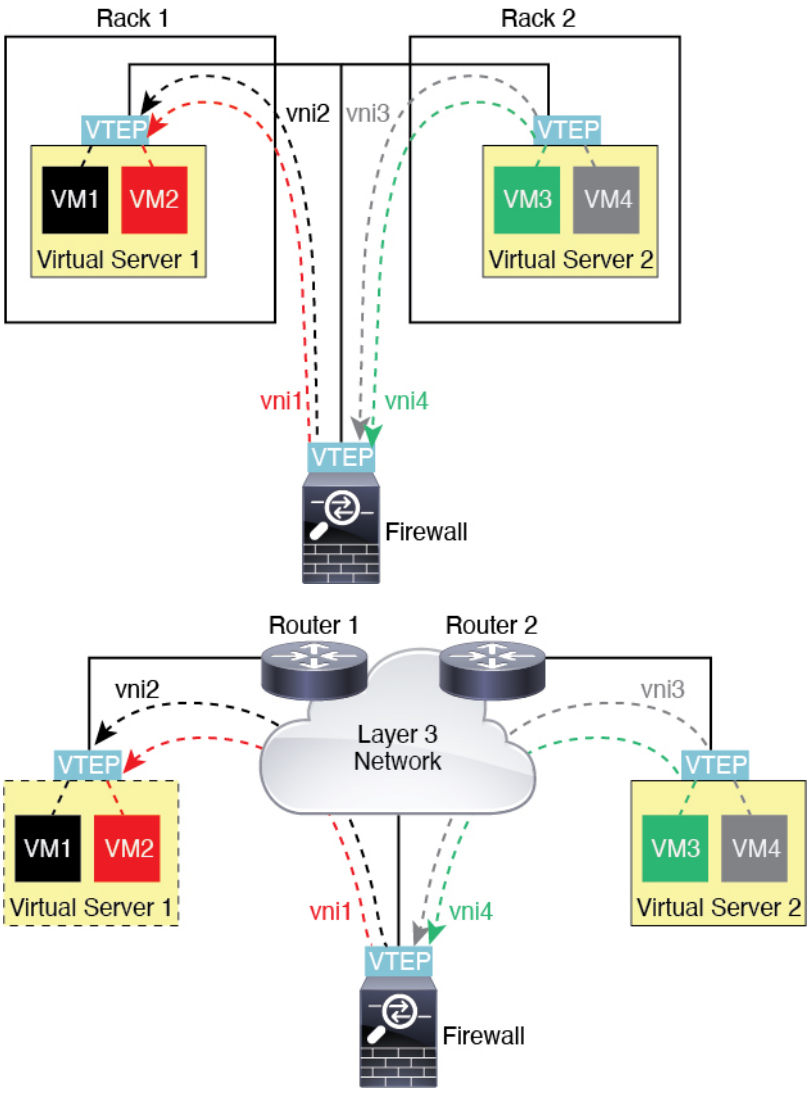
VXLAN Gateway (Routed Mode)

The ASA can serve as a router between VXLAN and non-VXLAN domains, connecting devices on different networks.



Router Between VXLAN Domains

With a VXLAN-stretched Layer 2 domain, a VM can point to an ASA as its gateway while the ASA is not on the same rack, or even when the ASA is far away over the Layer 3 network.



See the following notes about this scenario:

1. For packets from VM3 to VM1, the destination MAC address is the ASA MAC address, because the ASA is the default gateway.
2. The VTEP source interface on Virtual Server 2 receives packets from VM3, then encapsulates the packets with VNI 3's VXLAN tag and sends them to the ASA.
3. When the ASA receives the packets, it decapsulates the packets to get the inner frames.
4. The ASA uses the inner frames for route lookup, then finds that the destination is on VNI 2. If it does not already have a mapping for VM1, the ASA sends an encapsulated ARP broadcast on the multicast group IP on VNI 2.

Note The ASA must use dynamic VTEP peer discovery because it has multiple VTEP peers in this scenario.

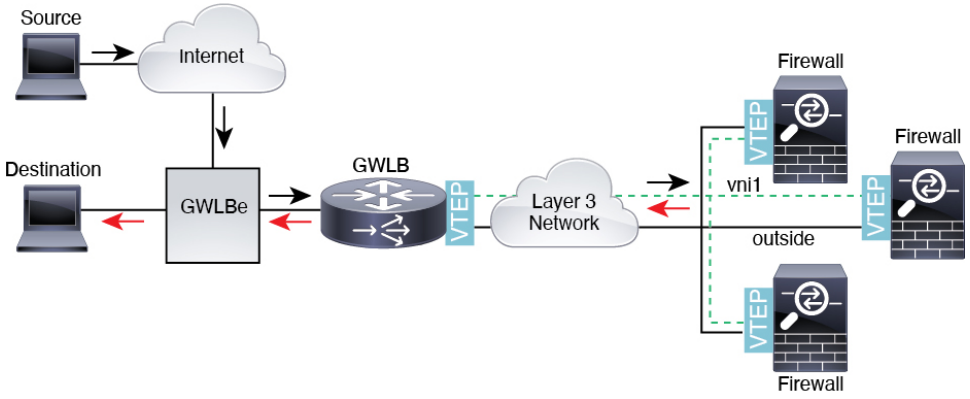
5. The ASA encapsulates the packets again with the VXLAN tag for VNI 2 and sends the packets to Virtual Server 1. Before encapsulation, the ASA changes the inner frame destination MAC address to be the MAC of VM1 (multicast-encapsulated ARP might be needed for the ASA to learn the VM1 MAC address).
6. When Virtual Server 1 receives the VXLAN packets, it decapsulates the packets and delivers the inner frames to VM1.

AWS Gateway Load Balancer and Geneve Single-Arm Proxy

Note This use case is the only currently supported use case for Geneve interfaces.

The AWS Gateway Load Balancer combines a transparent network gateway and a load balancer that distributes traffic and scales virtual appliances on demand. The ASAv supports the Gateway Load Balancer centralized control plane with a distributed data plane (Gateway Load Balancer endpoint). The following figure shows traffic forwarded to the Gateway Load Balancer from the Gateway Load Balancer endpoint. The Gateway Load Balancer balances traffic among multiple ASAvs, which inspect the traffic before either dropping it or sending it back to the Gateway Load Balancer (U-turn traffic). The Gateway Load Balancer then sends the traffic back to the Gateway Load Balancer endpoint and to the destination.

Figure 54: Geneve Single-Arm Proxy



Requirements and Prerequisites for VXLAN Interfaces

Model Requirements

- Firepower 1010 switch ports and VLAN interfaces are not supported as VTEP interfaces.
- Geneve encapsulation is supported for the following models: ASAv30, ASAv50, ASAv100 on Amazon Web Services (AWS)

Guidelines for VXLAN Interfaces

Firewall Mode

- Geneve interfaces are only supported in routed firewall mode.

IPv6

- The VNI interface supports both IPv4 and IPv6 traffic.
- The VTEP source interface IP address only supports IPv4.

Clustering and Multiple Context Mode

- ASA clustering does not support VXLAN in Individual Interface mode. Only Spanned EtherChannel mode supports VXLAN.
- Geneve interfaces are only supported in standalone, single context mode. They are not supported with clustering or multiple context mode.

Routing

- Only static routing or Policy Based Routing is supported on the VNI interface; dynamic routing protocols are not supported.

VPN

You cannot configure the VTEP source interface for VPN or use it as a VTI.

MTU

- VXLAN encapsulation—If the source interface MTU is less than 1554 bytes, then the ASA automatically raises the MTU. In this case, the entire Ethernet datagram is being encapsulated, so the new packet is larger and requires a larger MTU. If the MTU used by other devices is larger, then you should set the source interface MTU to be the network MTU + 54 bytes. This MTU requires you to enable jumbo frame reservation on some models; see Enable Jumbo Frame Support (ASAv, ISA 3000), on page 600.
- Geneve encapsulation—If the source interface MTU is less than 1806 bytes, then the ASA automatically raises the MTU to 1806 bytes. In this case, the entire Ethernet datagram is being encapsulated, so the new packet is larger and requires a larger MTU. If the MTU used by other devices is larger, then you should set the source interface MTU to be the network MTU + 306 bytes. This MTU requires you to enable jumbo frame reservation on some models; see Enable Jumbo Frame Support (ASAv, ISA 3000), on page 600.

Default Settings for VXLAN Interfaces

VNI interfaces are enabled by default.

Configure VXLAN Interfaces

To configure VXLAN, perform the following steps.

Note You can configure either VXLAN or Geneve (ASAv only). For Geneve interfaces, see Configure Geneve Interfaces, on page 657.

Procedure

Step 1 Configure the VTEP Source Interface, on page 653.

Step 2 Configure the VNI Interface, on page 655

Step 3 (Optional) Change the VXLAN UDP Port, on page 656.

Configure the VTEP Source Interface

You can configure one VTEP source interface per ASA or per security context. The VTEP is defined as a Network Virtualization Endpoint (NVE).

Before you begin

For multiple context mode, complete the tasks in this section in the context execution space. Enter the **changeto context** *name* command to change to the context you want to configure.

Procedure

Step 1 (Transparent mode) Specify that the source interface is NVE-only:

interface *id*

nve-only

Example:

```
ciscoasa(config)# interface gigabitethernet 1/1
ciscoasa(config-if)# nve-only
```

This setting lets you configure an IP address for the interface. This command is optional for routed mode where this setting restricts traffic to VXLAN and common management traffic only on this interface.

Step 2 Configure the source interface name and IPv4 address.

Example:

(Routed Mode)

```
ciscoasa(config)# interface gigabitethernet 1/1
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0
```

Example:

(Transparent Mode)

```
ciscoasa(config)# interface gigabitethernet 1/1
ciscoasa(config-if)# nve-only
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0
```

Step 3 Specify the NVE instance:

nve 1

You can only specify one NVE instance, with the ID 1.

Example:

```
ciscoasa(config)# nve 1
ciscoasa(cfg-nve)#
```

Step 4 Specify VXLAN encapsulation.

encapsulation vxlan

Example:

```
ciscoasa(cfg-nve)# encapsulation vxlan
```

Step 5 Specify the source interface name that you configured in Step 2:

source-interface *interface-name*

Example:

```
ciscoasa(cfg-nve)# source-interface outside
```

Note
If the VTEP interface MTU is less than 1554 bytes, then the ASA automatically raises the MTU to 1554 bytes.

Step 6 (Multiple context mode; Optional for single mode) Manually specify the peer VTEP IP address:

peer ip *ip_address*

Example:

```
ciscoasa(cfg-nve)# peer ip 10.1.1.2
```

If you specify the peer IP address, you cannot use multicast group discovery. Multicast is not supported in multiple context mode, so manual configuration is the only option. You can only specify one peer for the VTEP.

Step 7 (Optional; single mode only) Specify a default multicast group for all associated VNI interfaces:

default-mcast-group *mcast_ip*

Example:

```
ciscoasa(cfg-nve)# default-mcast-group 236.0.0.100
```

If you do not configure the multicast group per VNI interface, then this group is used. If you configure a group at the VNI interface level, then that group overrides this setting.

Configure the VNI Interface

Add a VNI interface, associate it with the VTEP source interface, and configure basic interface parameters.

Procedure

Step 1 Create the VNI interface:

interface vni *vni_num*

Example:

```
ciscoasa(config)# interface vni 1
```

Set the ID between 1 and 10000. This ID is only an internal interface identifier.

Step 2 Specify the VXLAN segment ID:

segment-id *id*

Example:

```
ciscoasa(config-if)# segment-id 1000
```

Set the ID between 1 and 16777215. The segment ID is used for VXLAN tagging.

Step 3 (Required for transparent mode) Specify the bridge group to which you want to associate this interface:

bridge-group *number*

Example:

```
ciscoasa(config-if)# bridge-group 1
```

See Configure Bridge Group Interfaces, on page 677 to configure the BVI interface and associate regular interfaces to this bridge group.

Step 4 Associate this interface with the VTEP source interface:

vtep-nve 1

Step 5 Name the interface:

nameif *vni_interface_name*

Example:

```
ciscoasa(config-if)# nameif vxlan1000
```

The *name* is a text string up to 48 characters, and is not case-sensitive. You can change the name by reentering this command with a new value. Do not enter the **no** form, because that command causes all commands that refer to that name to be deleted.

Step 6 (Routed mode) Assign an IPv4 and/or IPv6 address:

ip address {*ip_address* [*mask*] [**standby** *ip_address*] | **dhcp** [**setroute**] | **pppoe** [**setroute**]}

ipv6 address {**autoconfig** | *ipv6-address*/*prefix-length* [**standby** *ipv6-address*]}

Example:

```
ciscoasa(config-if)# ip address 192.168.1.1 255.255.255.0 standby 192.168.1.2
ciscoasa(config-if)# ipv6 address 2001:0DB8::BA98:0:3210/48
```

Step 7 Set the security level:

security-level *level*

Example:

```
ciscoasa(config-if)# security-level 50
```

Where *number* is an integer between 0 (lowest) and 100 (highest).

Step 8 (Single mode) Set the multicast group address:

mcast-group *multicast_ip*

Example:

```
ciscoasa(config-if)# mcast-group 236.0.0.100
```

If you do not set the multicast group for the VNI interface, the default group from the VTEP source interface configuration is used, if available. If you manually set a VTEP peer IP for the VTEP source interface, you cannot specify a multicast group for the VNI interface. Multicast is not supported in multiple context mode.

(Optional) Change the VXLAN UDP Port

By default, the VTEP source interface accepts VXLAN traffic to UDP port 4789. If your network uses a non-standard port, you can change it.

Before you begin

For multiple context mode, complete this task in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.

Procedure

Set the VXLAN UDP port:

vxlan *port number*

Example:

```
ciscoasa(config)# vxlan port 5678
```

Configure Geneve Interfaces

To configure Geneve interfaces for the ASAv, perform the following steps.

Note You can configure either VXLAN or Geneve. For VXLAN interfaces, see Configure VXLAN Interfaces, on page 653.

Procedure

Step 1 Configure the VTEP Source Interface for Geneve, on page 657.

Step 2 Configure the VNI Interface for Geneve, on page 658

Step 3 Allow Gateway Load Balancer Health Checks, on page 660.

Configure the VTEP Source Interface for Geneve

You can configure one VTEP source interface per ASAv. The VTEP is defined as a Network Virtualization Endpoint (NVE).

Procedure

Step 1 (Optional) Specify that the source interface is NVE-only.

interface *id*

nve-only

Example:

```
ciscoasa(config)# interface gigabitethernet 1/1
ciscoasa(config-if)# nve-only
```

This setting restricts traffic to VXLAN and common management traffic only on this interface.

Step 2 Configure the source interface name and IPv4 address.

Example:

```
ciscoasa(config)# interface gigabitethernet 1/1
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0
```

Step 3 Specify the NVE instance:

nve 1

You can only specify one NVE instance, with the ID 1.

Example:

```
ciscoasa(config)# nve 1
ciscoasa(cfg-nve)#
```

Step 4 Specify Geneve encapsulation.

encapsulation geneve

Do not change the Geneve port; AWS requires a port of 6081.

Example:

```
ciscoasa(cfg-nve)# encapsulation geneve
```

Step 5 Specify the source interface name that you configured in Step 2:

source-interface *interface-name*

Example:

```
ciscoasa(cfg-nve)# source-interface outside
```

Note
If the source interface MTU is less than 1806 bytes, then the ASA automatically raises the MTU to 1806 bytes.

Configure the VNI Interface for Geneve

Add a VNI interface, associate it with the VTEP source interface, and configure basic interface parameters.

Procedure

Step 1 Create the VNI interface:

interface vni *vni_num*

Example:

```
ciscoasa(config)# interface vni 1
```

Set the ID between 1 and 10000. This ID is only an internal interface identifier.

Step 2 Associate this interface with the VTEP source interface:

vtep-nve 1

Step 3 Name the interface:

nameif *vni_interface_name*

Example:

```
ciscoasa(config-if)# nameif geneve1000
```

The *name* is a text string up to 48 characters, and is not case-sensitive. You can change the name by reentering this command with a new value. Do not enter the **no** form, because that command causes all commands that refer to that name to be deleted.

Step 4 Assign an IPv4 and/or IPv6 address:

ip address {*ip_address* [*mask*] [**standby** *ip_address*]}

ipv6 address {**autoconfig** | *ipv6-address*/*prefix-length* [**standby** *ipv6-address*]}

Geneve only supports a static IP address.

Example:

```
ciscoasa(config-if)# ip address 192.168.1.1 255.255.255.0 standby 192.168.1.2
ciscoasa(config-if)# ipv6 address 2001:0DB8::BA98:0:3210/48
```

Step 5 Set the security level:

security-level *level*

Where *level* is an integer between 0 (lowest) and 100 (highest).

Example:

```
ciscoasa(config-if)# security-level 50
```

Step 6 Enable single-arm proxy .

proxy single-arm

Example:

```
ciscoasa(config-if)# proxy single-arm
```

Step 7 Allow traffic to enter and exit the same interface.

same-security-traffic permit intra-interface

Example:

```
ciscoasa(config)# same-security-traffic permit intra-interface
```

Allow Gateway Load Balancer Health Checks

The AWS Gateway Load Balancer requires appliances to answer a health check properly. The AWS Gateway Load Balancer will only send traffic to appliances that are considered healthy.

You must configure the ASAv to respond to an SSH, Telnet, HTTP, or HTTPS health check.

SSH Connection

For SSH, allow SSH from the Gateway Load Balancer. The Gateway Load Balancer will attempt to establish a connection to the ASAv, and the ASAv prompt to log in is taken as proof of health.

Note An SSH login attempt will time out after 1 minute. You will need to configure a longer health check interval on the Gateway Load Balancer to accommodate this timeout.

Example

```
! Allow SSH connections from GWLB network: 10.0.1.0/24
ssh 10.0.1.0 255.255.255.0 outside
```

Telnet Connection

For Telnet, allow Telnet from the Gateway Load Balancer. The Gateway Load Balancer will attempt to establish a connection to the ASAv, and the ASAv prompt to log in is taken as proof of health.

Note You cannot Telnet to the lowest security level interface, so this method may not be practical.

Example

```
! Allow Telnet connections from GWLB network: 10.0.1.0/24
telnet 10.0.1.0 255.255.255.0 outside
```

HTTP(S) Cut-Through Proxy

You can configure the ASA to prompt the Gateway Load Balancer for an HTTP(S) login.

Example

```
! Identify health probe HTTPS traffic from GWLB nw 10.0.1.0/24 to ASAv interface 10.2.2.2
access-list gwlb extended permit tcp 10.0.1.0 255.255.255.0 host 10.2.2.2 eq https
! Enable HTTPS authentication
aaa authentication https console LOCAL
! Require authentication for the health probe traffic
aaa authentication match gwlb outside LOCAL
! Use an HTTPs login page on the ASA
aaa authentication listener https outside port https
```

HTTP(S) Redirection Using Static Interface NAT with Port Translation

You can configure the ASAv to redirect health checks to a metadata HTTP(S) server. For HTTP(S) health checks, the HTTP(S) server must reply to the Gateway Load Balancer with a status code in the range 200 to 399. Because the ASAv has limits on the the number of simultaneous management connections, you may choose to offload the health check to an external server.

Static interface NAT with port translation lets you redirect a connection to a port (such as port 80) to a different IP address. For example, translate an HTTP packet from the Gateway Load Balancer with a destination of the ASAv outside interface so that it appears to be from the ASAv outside interface with a destination of the HTTP server. The ASAv then forwards the packet to the mapped destination address. The HTTP server responds to the ASAv outside interface, and then the ASAv forwards the response back to the Gateway Load Balancer. You need an access rule that allows traffic from the Gateway Load Balancer to the HTTP server.

Example

```
! Permit HTTPS traffic from GWLB nw 10.0.1.0/24 to HTTP server 10.2.2.3
access-list gwlb-health extended permit tcp 10.0.1.0 255.255.255.0 host 10.2.2.3 eq https
access-group gwlb-health in interface outside

! Create network objects
object network gwlb-subnet
 subnet 10.0.1.0 255.255.255.0
object-group network gwlb
    network-object object gwlb-subnet
object-group network http-server
    network-object host 10.2.2.3
object service http443
    service tcp destination eq https

! For HTTPS, translate src GWLB IP to outside IP; translate dest of outside IP to HTTPS
Server IP
nat (outside,outside) source static gwlb interface destination static interface http-server
 service http443 http443
```

Monitoring VXLAN Interfaces

See the following commands to monitor VTEP and VNI interfaces.

- **show nve** [*id*] [**summary**]

This command shows the parameters, status and statistics of a NVE interface, status of its carrier interface (source-interface), IP address of the carrier interface, VNIs that use this NVE as the VXLAN VTEP, and peer VTEP IP addresses associated with this NVE interface. With the **summary** option, this command only shows the status of the NVE interface, number of VNIs behind the NVE interface, and number of VTEPs discovered.

See the following output for the **show nve 1** command:

```
ciscoasa# show nve 1
ciscoasa(config-if)# show nve
nve 1, source-interface "inside" is up
IP address 15.1.2.1, subnet mask 255.255.255.0
Encapsulation: vxlan
Encapsulated traffic statistics:
6701004 packets input, 3196266002 bytes
6700897 packets output, 3437418084 bytes
1 packets dropped
Number of configured static peer VTEPs: 0
Number of discovered peer VTEPs: 1
Discovered peer VTEPs:
IP address 15.1.2.3
Number of VNIs attached to nve 1: 2
VNIs attached:
vni 2: segment-id 5002, mcast-group 239.1.2.3
vni 1: segment-id 5001, mcast-group 239.1.2.3
```

See the following output for the **show nve 1 summary** command:

```
ciscoasa# show nve 1 summary
nve 1, source-interface "inside" is up
Encapsulation: vxlan
Number of configured static peer VTEPs: 0
Number of discovered peer VTEPs: 1
Default multicast group: 239.1.2.3
Number of VNIs attached to nve 1: 2
```

- **show interface vni** *id* [**summary**]

  This command shows the parameters, status and statistics of a VNI interface, status of its bridged interface (if configured), and NVE interface it is associated with. The **summary** option shows only the VNI interface parameters.

  See the following output for the **show interface vni 1** command:

```
ciscoasa# show interface vni 1
Interface vni1 "vni-inside", is up, line protocol is up
VTEP-NVE 1
Segment-id 5001
Tag-switching: disabled
MTU: 1500
MAC: aaaa.bbbb.1234
IP address 192.168.0.1, subnet mask 255.255.255.0
Multicast group 239.1.3.3
Traffic Statistics for "vni-inside":
235 packets input, 23606 bytes
524 packets output, 32364 bytes
14 packets dropped
1 minute input rate 0 pkts/sec, 0 bytes/sec
1 minute output rate 0 pkts/sec, 2 bytes/sec
```

```
1 minute drop rate, 0 pkts/sec
5 minute input rate 0 pkts/sec, 0 bytes/sec
5 minute output rate 0 pkts/sec, 0 bytes/sec
5 minute drop rate, 0 pkts/sec
```

See the following output for the **show interface vni 1 summary** command:

```
ciscoasa# show interface vni 1 summary
Interface vni1 "vni-inside", is up, line protocol is up
VTEP-NVE 1
Segment-id 5001
Tag-switching: disabled
MTU: 1500
MAC: aaaa.bbbb.1234
IP address 192.168.0.1, subnet mask 255.255.255.0
Multicast group not configured
```

- **show vni vlan-mapping**

  This command shows the mapping between VNI segment IDs and VLAN interfaces or physical interfaces. This command is only valid in transparent firewall mode because in routed mode, the mapping between VXLANs and VLANs can include too many values to show.

  See the following output for the **show vni vlan-mapping** command:

```
ciscoasa# show vni vlan-mapping
vni1: segment-id: 6000, interface: 'g0110', vlan 10, interface: 'g0111', vlan 11
vni2: segment_id: 5000, interface: 'g01100', vlan 1, interface: 'g111', vlan 3, interface:
 'g112', vlan 4
```

- **show arp vtep-mapping**

  This command displays MAC addresses cached on the VNI interface for IP addresses located in the remote segment domain and the remote VTEP IP addresses.

  See the following output for the **show arp vtep-mapping** command:

```
ciscoasa# show arp vtep-mapping
vni-outside 192.168.1.4 0012.0100.0003 577 15.1.2.3
vni-inside 192.168.0.4 0014.0100.0003 577 15.1.2.3
```

- **show mac-address-table vtep-mapping**

  This command displays the Layer 2 forwarding table (MAC address table) on the VNI interface with the remote VTEP IP addresses.

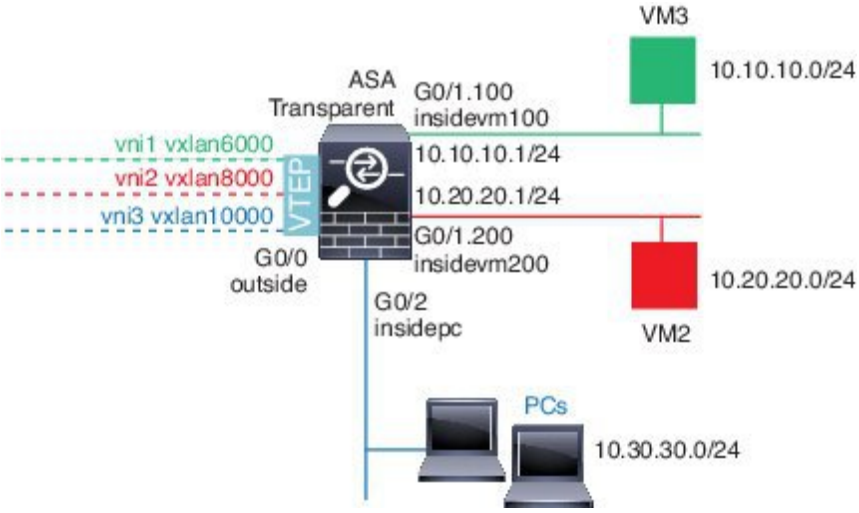
  See the following output for the **show mac-address-table vtep-mapping** command:

```
ciscoasa# show mac-address-table vtep-mapping
interface                  mac  address       type     Age(min)   bridge-group    VTEP
-----------------------------------------------------------------------------------------
vni-outside               00ff.9200.0000      dynamic      5         1        10.9.1.3
vni-inside                 0041.9f00.0000      dynamic      5         1      10.9.1.3
```

Examples for VXLAN Interfaces

See the following configuration examples for VXLAN.

Transparent VXLAN Gateway Example



See the following description of this example:

- The outside interface on GigabitEthernet 0/0 is used as the VTEP source interface, and it is connected to the Layer 3 network.
- The insidevm100 VLAN subinterface on GigabitEthernet 0/1.100 is connected to the 10.10.10.0/24 network, on which VM3 resides. When VM3 communicates with VM1 (not shown; both have 10.10.10.0/24 IP addresses), the ASA uses VXLAN tag 6000.
- The insidevm200 VLAN subinterface on GigabitEthernet 0/1.200 is connected to the 10.20.20.0/24 network, on which VM2 resides. When VM2 communicates with VM4 (not shown; both have 10.20.20.0/24 IP addresses), the ASA uses VXLAN tag 8000.
- The insidepc interface on GigabitEthernet 0/2 is connected to the 10.30.30.0/24 network on which a few PCs reside. When those PCs communicate with VMs/PCs (not shown) behind a remote VTEP that belongs to same network (all have 10.30.30.0/24 IP addresses), the ASA uses VXLAN tag 10000.

ASA Configuration

```
firewall transparent
vxlan port 8427
!
interface gigabitethernet0/0
  nve-only
  nameif outside
  ip address 192.168.1.30 255.255.255.0
  no shutdown
!
nve 1
```

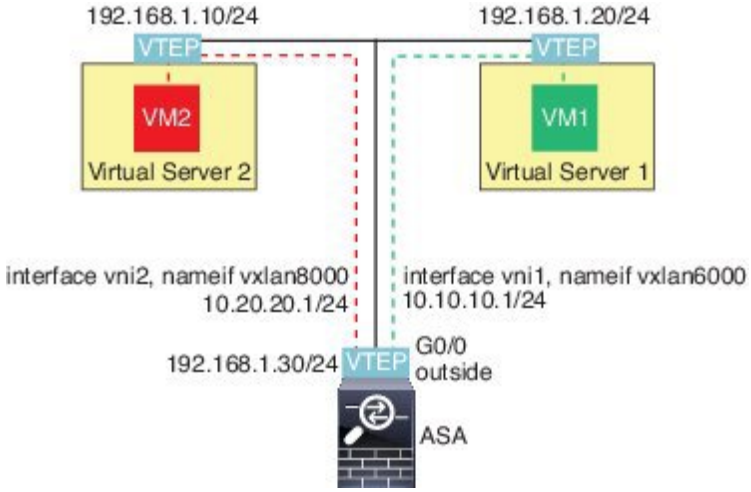
```
  encapsulation vxlan
  source-interface outside
!
interface vni1
  segment-id 6000
  nameif vxlan6000
  security-level 0
  bridge-group 1
  vtep-nve 1
  mcast-group 235.0.0.100
!
interface vni2
  segment-id 8000
  nameif vxlan8000
  security-level 0
  bridge-group 2
  vtep-nve 1
  mcast-group 236.0.0.100
!
interface vni3
  segment-id 10000
  nameif vxlan10000
  security-level 0
  bridge-group 3
  vtep-nve 1
  mcast-group 236.0.0.100
!
interface gigabitethernet0/1.100
  nameif insidevm100
  security-level 100
  bridge-group 1
!
interface gigabitethernet0/1.200
  nameif insidevm200
  security-level 100
  bridge-group 2
!
interface gigabitethernet0/2
  nameif insidepc
  security-level 100
  bridge-group 3
!
interface bvi 1
  ip address 10.10.10.1 255.255.255.0
!
interface bvi 2
  ip address 10.20.20.1 255.255.255.0
!
interface bvi 3
  ip address 10.30.30.1 255.255.255.0
```

Notes

- For VNI interfaces vni1 and vni2, the inner VLAN tag is removed during encapsulation.
- VNI interfaces vni2 and vni3 share the same multicast IP address for encapsulated ARP over multicast. This sharing is allowed.
- The ASA bridges the VXLAN traffic to non-VXLAN-supported interfaces based on the above BVIs and bridge group configurations. For each of the stretched Layer 2 network segments (10.10.10.0/24, 10.20.20.0/24 and 10.30.30.0/24), the ASA serves as a bridge.

- It is allowed to have more than one VNI or more than one regular interface (VLAN or just physical interface) in a bridge group. The forwarding or association between VXLAN segment ID to the VLAN ID (or a physical interface) is decided by the destination MAC address and which interface connects to the destination.
- The VTEP source-interface is a Layer 3 interface in transparent firewall mode indicated by **nve-only** in the interface configuration. The VTEP source interface is not a BVI interface or a management interface, but it has an IP address and uses the routing table.

VXLAN Routing Example



See the following description of this example:

- VM1 (10.10.10.10) is hosted on Virtual Server 1, and VM2 (10.20.20.20) is hosted on Virtual Server 2.
- The default gateway for VM1 is the ASA, which is not in the same pod as Virtual Server 1, but VM1 is not aware of it. VM1 only knows that its default gateway IP address is 10.10.10.1. Similarly, VM2 only knows that its default gateway IP address is 10.20.20.1.
- The VTEP-supported hypervisors on Virtual Server 1 and 2 are able to communicate with the ASA over the same subnet or through a Layer 3 network (not shown; in which case, the ASA and uplinks of virtual servers have different network addresses).
- VM1's packet will be encapsulated by its hypervisor's VTEP and sent to its default gateway over VXLAN tunneling.
- When VM1 sends a packet to VM2, the packet will be sent through default gateway 10.10.10.1 from its perspective. Virtual Server1 knows 10.10.10.1 is not local, so the VTEP encapsulates the packet over VXLAN and sends it to ASA's VTEP.
- On the ASA, the packet is decapsulated. The VXLAN segment ID is learned during decapsulation. The ASA then re-injects the inner frame to the corresponding VNI interface (vni1) based on the VXLAN segment ID. The ASA then conducts a route lookup and sends the inner packet through another VNI interface, vni2. All egressing packets through vni2 are encapsulated with VXLAN segment 8000 and sent through the VTEP to outside.

- Eventually the encapsulated packet is received by the VTEP of Virtual Server 2, which decapsulates it and forwards it to VM2.

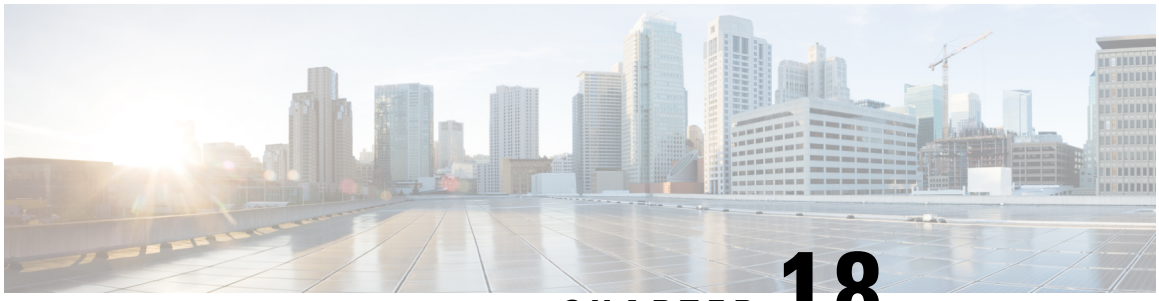
ASA Configuration

```
interface gigabitethernet0/0
  nameif outside
  ip address 192.168.1.30 255.255.255.0
  no shutdown
!
nve 1
  encapsulation vxlan
  source-interface outside
  default-mcast-group 235.0.0.100
!
interface vni1
  segment-id 6000
  nameif vxlan6000
  security-level 0
  vtep-nve 1
  ip address 10.20.20.1 255.255.255.0
!
interface vni2
  segment-id 8000
  nameif vxlan8000
  security-level 0
  vtep-nve 1
  ip address 10.10.10.1 255.255.255.0
!
```

History for VXLAN Interfaces

Table 27: History for VXLAN Interfaces

Feature Name	Releases	Feature Information
Geneve support for the ASAv on AWS for the AWS Gateway Load Balancer	9.17(1)	Geneve encapsulation support was added for the ASAv30, ASAv50, and ASAv100 to support single-arm proxy for the AWS Gateway Load Balancer. New/Modified commands: **debug geneve**, **debug nve**, **debug vxlan**, **encapsulation**, **packet-tracer geneve**, **proxy single-arm**, **show asp drop**, **show capture**, **show interface**, **show nve**,
VXLAN support	9.4(1)	VXLAN support was added, including VXLAN tunnel endpoint (VTEP) support. You can define one VTEP source interface per ASA or security context. We introduced the following commands: **debug vxlan**, **default-mcast-group**, **encapsulation vxlan**, **inspect vxlan**, **interface vni**, **mcast-group**, **nve**, **nve-only**, **peer ip**, **segment-id**, **show arp vtep-mapping**, **show interface vni**, **show mac-address-table vtep-mapping**, **show nve**, **show vni vlan-mapping**, **source-interface**, **vtep-nve**, **vxlan port**



CHAPTER 18

Routed and Transparent Mode Interfaces

This chapter includes tasks to complete the interface configuration for all models in routed or transparent firewall mode.

Note For multiple context mode, complete the tasks in this section in the context execution space. Enter the **changeto context** *name* command to change to the context you want to configure.

- About Routed and Transparent Mode Interfaces, on page 669
- Guidelines and Limitations for Routed and Transparent Mode Interfaces, on page 671
- Configure Routed Mode Interfaces, on page 673
- Configure Bridge Group Interfaces, on page 677
- Configure IPv6 Addressing, on page 683
- Monitoring Routed and Transparent Mode Interfaces, on page 695
- Examples for Routed and Transparent Mode Interfaces, on page 700
- History for Routed and Transparent Mode Interfaces, on page 703

About Routed and Transparent Mode Interfaces

The ASA supports two types of interfaces: routed and bridged.

Each Layer 3 routed interface requires an IP address on a unique subnet.

Bridged interfaces belong to a bridge group, and all interfaces are on the same network. The bridge group is represented by a Bridge Virtual Interface (BVI) that has an IP address on the bridge network. Routed mode supports both routed and bridged interfaces, and you can route between routed interfaces and BVIs. Transparent firewall mode only supports bridge group and BVI interfaces.

Security Levels

Each interface must have a security level from 0 (lowest) to 100 (highest), including bridge group member interfaces. For example, you should assign your most secure network, such as the inside host network, to level 100. While the outside network connected to the Internet can be level 0. Other networks, such as DMZs can be in between. You can assign interfaces to the same security level.

Whether you assign a security level to a BVI depends on the firewall mode. In transparent mode, the BVI interface does not have a security level because it does not participate in routing between interfaces. In routed mode, BVI interfaces have a security level if you choose to route between the BVIs and other interfaces. For routed mode, the security level on a bridge group member interface only applies for communication within the bridge group. Similarly, the BVI security level only applies for inter-BVI/Layer 3 interface communication.

The level controls the following behavior:

- Network access—By default, there is an implicit permit from a higher security interface to a lower security interface (outbound). Hosts on the higher security interface can access any host on a lower security interface. You can limit access by applying an ACL to the interface.

  If you enable communication for same-security interfaces, there is an implicit permit for interfaces to access other interfaces on the same security level or lower.

- Inspection engines—Some application inspection engines are dependent on the security level. For same-security interfaces, inspection engines apply to traffic in either direction.
  - NetBIOS inspection engine—Applied only for outbound connections.
  - SQL*Net inspection engine—If a control connection for the SQL*Net (formerly OraServ) port exists between a pair of hosts, then only an inbound data connection is permitted through the ASA.

Dual IP Stack (IPv4 and IPv6)

The ASA supports both IPv6 and IPv4 addresses on an interface. Make sure you configure a default route for both IPv4 and IPv6.

31-Bit Subnet Mask

For routed interfaces, you can configure an IP address on a 31-bit subnet for point-to-point connections. The 31-bit subnet includes only 2 addresses; normally, the first and last address in the subnet is reserved for the network and broadcast, so a 2-address subnet is not usable. However, if you have a point-to-point connection and do not need network or broadcast addresses, a 31-bit subnet is a useful way to preserve addresses in IPv4. For example, the failover link between 2 ASAs only requires 2 addresses; any packet that is transmitted by one end of the link is always received by the other, and broadcasting is unnecessary. You can also have a directly-connected management station running SNMP or Syslog.

31-Bit Subnet and Clustering

You can use a 31-bit subnet mask in Spanned clustering mode, excluding the management interface and the Cluster Control Link.

You cannot use a 31-bit subnet mask in Individual clustering mode on any interface.

31-Bit Subnet and Failover

For failover, when you use a 31-bit subnet for the ASA interface IP address, you cannot configure a standby IP address for the interface because there are not enough addresses. Normally, an interface for failover should have a standby IP address so the active unit can perform interface tests to ensure standby interface health. Without a standby IP address, the ASA cannot perform any network tests; only the link state can be tracked.

For the failover and optional separate state link, which are point-to-point connections, you can also use a 31-bit subnet.

31-Bit Subnet and Management

If you have a directly-connected management station, you can use a point-to-point connection for SSH or HTTP on the ASA, or for SNMP or Syslog on the management station.

31-Bit Subnet Unsupported Features

The following features do not support the 31-Bit subnet:

- BVI interfaces for bridge groups—The bridge group requires at least 3 host addresses: the BVI, and two hosts connected to two bridge group member interfaces. you must use a /29 subnet or smaller.
- Multicast Routing

Guidelines and Limitations for Routed and Transparent Mode Interfaces

Context Mode

- In multiple context mode, you can only configure context interfaces that you already assigned to the context in the system configuration according to Configure Multiple Contexts, on page 224.
- PPPoE is not supported in multiple context mode.
- For multiple context mode in transparent mode, each context must use different interfaces; you cannot share an interface across contexts.
- For multiple context mode in transparent mode, each context typically uses a different subnet. You can use overlapping subnets, but your network topology requires router and NAT configuration to make it possible from a routing standpoint.
- DHCPv6 and prefix delegation options are not supported with multiple context mode.
- In routed firewall mode, bridge group interfaces are not supported in multiple context mode.

Failover, Clustering

- Do not configure failover links with the procedures in this chapter. See the Failover chapter for more information.
- For cluster interfaces, see the clustering chapter for requirements.
- When you use Failover, you must set the IP address and standby address for data interfaces manually; DHCP and PPPoE are not supported.

IPv6

- IPv6 is supported on all interfaces.

- You can only configure IPv6 addresses manually in transparent mode.
- The ASA does not support IPv6 anycast addresses.
- DHCPv6 and prefix delegation options are not supported with multiple context mode, transparent mode, clustering, or Failover.

Model Guidelines

- For the ASAv50, bridge groups are not supported in either transparent or routed mode.
- For the Firepower 2100 series, bridge groups are not supported in routed mode.

Transparent Mode and Bridge Group Guidelines

- You can create up to 250 bridge groups, with 64 interfaces per bridge group.
- Each directly-connected network must be on the same subnet.
- The ASA does not support traffic on secondary networks; only traffic on the same network as the BVI IP address is supported.
- An IP address for the BVI is required for each bridge group for to-the-device and from-the-device management traffic, as well as for data traffic to pass through the ASA. For IPv4 traffic, specify an IPv4 address. For IPv6 traffic, specify an IPv6 address.
- You can only configure IPv6 addresses manually.
- The BVI IP address must be on the same subnet as the connected network. You cannot set the subnet to a host subnet (255.255.255.255).
- Management interfaces are not supported as bridge group members.
- For the ASAv50 on VMware with bridged ixgbevf interfaces, transparent mode is not supported, and bridge groups are not supported in routed mode.
- For the Firepower 2100 series, bridge groups are not supported in routed mode.
- For the 200/ 1010/ , you cannot mix logical VLAN interfaces and physical firewall interfaces in the same bridge group.
- In transparent mode, you must use at least 1 bridge group; data interfaces must belong to a bridge group.
- In transparent mode, do not specify the BVI IP address as the default gateway for connected devices; devices need to specify the router on the other side of the ASA as the default gateway.
- In transparent mode, the *default* route, which is required to provide a return path for management traffic, is only applied to management traffic from one bridge group network. This is because the default route specifies an interface in the bridge group as well as the router IP address on the bridge group network, and you can only define one default route. If you have management traffic from more than one bridge group network, you need to specify a regular static route that identifies the network from which you expect management traffic.
- In transparent mode, PPPoE is not supported for the Management interface.
- In routed mode, to route between bridge groups and other routed interfaces, you must name the BVI.

- In routed mode, ASA-defined EtherChannel and VNI interfaces are not supported as bridge group members. EtherChannels on the Firepower 4100/9300 can be bridge group members.
- Bidirectional Forwarding Detection (BFD) echo packets are not allowed through the ASA when using bridge group members. If there are two neighbors on either side of the ASA running BFD, then the ASA will drop BFD echo packets because they have the same source and destination IP address and appear to be part of a LAND attack.

Default Security Level

The default security level is 0. If you name an interface "inside," and you do not set the security level explicitly, then the ASA sets the security level to 100.



Note If you change the security level of an interface, and you do not want to wait for existing connections to time out before the new security information is used, you can clear the connections using the **clear conn** command.

Additional Guidelines and Requirements

- The ASA supports only one 802.1Q header in a packet and does not support multiple headers (known as Q-in-Q support).
- Interface problems, such as frequent up/down status changes, can prevent the floating connection timer from applying correctly to the connections going through the interface. If you have problems with an interface's status, consider clearing all connections after the status becomes stable to clear invalid connections.

Configure Routed Mode Interfaces

To configure routed mode interfaces, perform the following steps.

Configure General Routed Mode Interface Parameters

This procedure describes how to set the name, security level, IPv4 address, and other options.

Before you begin

In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the **changeto context** *name* command.

Procedure

Step 1 Enter interface configuration mode:

interface *id*

Example:

```
ciscoasa(config)# interface gigabithethernet 0/0
```

The interface ID can be:

- **port-channel**
- *physical*—For example, **ethernet**, **gigabitethernet**, **tengigabitethernet**, **management**. Refer to the hardware installation guide for your model for interface names.
- *physical***.***subinterface*—For example, **gigabitethernet0/0.100**.
- **vni**
- **vlan**
- *mapped_name*—For multiple context mode.

Note
For the Firepower 1010, you cannot configure switch ports as routed mode interfaces.

Step 2 Name the interface:

nameif *name*

Example:

```
ciscoasa(config-if)# nameif inside
```

The *name* is a text string up to 48 characters, and is not case-sensitive. You can change the name by reentering this command with a new value. Do not enter the **no** form, because that command causes all commands that refer to that name to be deleted.

Step 3 Set the IP address using one of the following methods.

For failover and clustering, you must set the IP address manually; DHCP and PPPoE are not supported.

- Set the IP address manually:

  ip address *ip_address* [*mask*] [**standby** *ip_address*]

  Example:

  ```
  ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0 standby 10.1.1.2
  ```

  The standby *ip_address* argument is used for failover. If you do not set the standby IP address, the active unit cannot monitor the standby interface using network tests; it can only track the link state.

  The *ip_address* and *mask* arguments set the interface IP address and subnet mask. For point-to-point connections, you can specify a 31-bit subnet mask (255.255.255.254). In this case, no IP addresses are reserved for the network or broadcast addresses. You cannot set the standby IP address in this case.

  Example:

  ```
  ciscoasa(config-if)# ip address 10.1.1.0 255.255.255.254
  ```

- Obtain an IP address from a DHCP server:

ip address dhcp [**setroute**]

Example:

```
ciscoasa(config-if)# ip address dhcp
```

The **setroute** keyword lets the ASA use the default route supplied by the DHCP server.

Reenter this command to reset the DHCP lease and request a new lease.

Note
If you do not enable the interface using the **no shutdown** command before you enter the **ip address dhcp** command, some DHCP requests might not be sent.

- Obtain an IP address from a PPPoE server:

  ip address pppoe [**setroute**]

  Example:

  ```
  ciscoasa(config-if)# ip address pppoe setroute
  ```

  You can alternatively enable PPPoE by manually entering the IP address:

  ip address *ip_address mask* **pppoe**

  Example:

  ```
  ciscoasa(config-if)# ip address 10.1.1.78 255.255.255.0 pppoe
  ```

  The **setroute** option sets the default routes when the PPPoE client has not yet established a connection. When using the **setroute** option, you cannot have a statically defined route in the configuration.

  Note
  If PPPoE is enabled on two interfaces (such as a primary and backup interface), and you do not configure dual ISP support, then the ASA can only send traffic through the first interface to acquire an IP address.

Step 4 Set the security level:

security-level *number*

Example:

```
ciscoasa(config-if)# security-level 50
```

The *number* is an integer between 0 (lowest) and 100 (highest)..

Step 5 (Optional) Set an interface to management-only mode so that it does not pass through traffic:

management-only

By default, Management interfaces are configured as management-only.

Examples

The following example configures parameters for VLAN 101:

```
ciscoasa(config)# interface vlan 101
ciscoasa(config-if)# nameif inside
ciscoasa(config-if)# security-level 100
ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0
```

The following example configures parameters in multiple context mode for the context configuration. The interface ID is a mapped name.

```
ciscoasa/contextA(config)# interface int1
ciscoasa/contextA(config-if)# nameif outside
ciscoasa/contextA(config-if)# security-level 100
ciscoasa/contextA(config-if)# ip address 10.1.2.1 255.255.255.0
```

Related Topics

Configure IPv6 Addressing, on page 683
Enable the Physical Interface and Configure Ethernet Parameters, on page 597
Configure PPPoE, on page 676

Configure PPPoE

If the interface is connected to a DSL, cable modem, or other connection to your ISP, and your ISP uses PPPoE to provide your IP address, configure the following parameters.

Procedure

Step 1 Define the Virtual Private Dialup Network (VPDN) group name of your choice to represent this connection:

vpdn group *group_name* **request dialout pppoe**

Example:

```
ciscoasa(config)# vpdn group pppoe-sbc request dialout pppoe
```

Step 2 If your ISP requires authentication, select an authentication protocol:

vpdn group *group_name* **ppp authentication** {**chap** | **mschap** | **pap**}

Example:

```
ciscoasa(config)# vpdn group pppoe-sbc ppp authentication chap
```

Enter the appropriate keyword for the type of authentication used by your ISP.

When using CHAP or MS-CHAP, the username may be referred to as the remote system name, while the password may be referred to as the CHAP secret.

Step 3 Associate the username assigned by your ISP to the VPDN group:

vpdn group *group_name* **localname** *username*

Example:

```
ciscoasa(config)# vpdn group pppoe-sbc localname johncrichton
```

Step 4 Create a username and password pair for the PPPoE connection:

vpdn username *username* **password** *password* [**store-local**]

Example:

```
ciscoasa(config)# vpdn username johncrichton password moya
```

The **store-local** option stores the username and password in a special location of NVRAM on the ASA. If an Auto Update Server sends a **clear config** command to the ASA and the connection is then interrupted, the ASA can read the username and password from NVRAM and re-authenticate to the Access Concentrator.

Configure Bridge Group Interfaces

A bridge group is a group of interfaces that the ASA bridges instead of routes. Bridge groups are supported in both transparent and routed firewall mode. For more information about bridge groups, see Bridge groups, on page 187.

To configure bridge groups and associated interfaces, perform these steps.

Configure the Bridge Virtual Interface (BVI)

Each bridge group requires a BVI for which you configure an IP address. The ASA uses this IP address as the source address for packets originating from the bridge group. The BVI IP address must be on the same subnet as the connected network. For IPv4 traffic, the BVI IP address is required to pass any traffic. For IPv6 traffic, you must, at a minimum, configure the link-local addresses to pass traffic, but a global management address is recommended for full functionality, including remote management and other management operations.

For routed mode, if you provide a name for the BVI, then the BVI participates in routing. Without a name, the bridge group remains isolated as in transparent firewall mode.

Some models include a bridge group and BVI in the default configuration. You can create additional bridge groups and BVIs and reassign member interfaces between the groups.

Note For a separate management interface in transparent mode (for supported models), a non-configurable bridge group (ID 301) is automatically added to your configuration. This bridge group is not included in the bridge group limit.

Procedure

Step 1 Create a BVI:

interface bvi *bridge_group_number*

Example:

```
ciscoasa(config)# interface bvi 2
```

The *bridge_group_number* is an integer between 1 and 250. You will later assign physical interfaces to this bridge group number.

Step 2 (Transparent Mode) Specify the IP address for the BVI:

ip address *ip_address* [*mask*] [**standby** *ip_address*]

Example:

```
ciscoasa(config-if)# ip address 10.1.3.1 255.255.255.0 standby 10.1.3.2
```

Do not assign a host address (/32 or 255.255.255.255) to the BVI. Also, do not use other subnets that contain fewer than 3 host addresses (one each for the upstream router, downstream router, and BVI) such as a /30 subnet (255.255.255.252). The ASA drops all ARP packets to or from the first and last addresses in a subnet. Therefore, if you use a /30 subnet and assign a reserved address from that subnet to the upstream router, then the ASA drops the ARP request from the downstream router to the upstream router.

The **standby** keyword and address is used for failover.

Step 3 (Routed Mode) Set the IP address using one of the following methods.

For failover and clustering, you must set the IP address manually; DHCP is not supported.

- Set the IP address manually:

  ip address *ip_address* [*mask*] [**standby** *ip_address*]

  Example:

  ```
  ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0 standby 10.1.1.2
  ```

  The standby *ip_address* argument is used for failover.

  The *ip_address* and *mask* arguments set the interface IP address and subnet mask.

- Obtain an IP address from a DHCP server:

  ip address dhcp [**setroute**]

  Example:

  ```
  ciscoasa(config-if)# ip address dhcp
  ```

  The **setroute** keyword lets the ASA use the default route supplied by the DHCP server.

Reenter this command to reset the DHCP lease and request a new lease.

If you do not enable the interface using the no shutdown command before you enter the **ip address dhcp** command, some DHCP requests might not be sent.

Step 4 (Routed Mode) Name the interface:

nameif *name*

Example:

```
ciscoasa(config-if)# nameif inside
```

You must name the BVI if you want to route traffic outside the bridge group members, for example, to the outside interface or to members of other bridge groups. The *name* is a text string up to 48 characters, and is not case-sensitive. You can change the name by reentering this command with a new value. Do not enter the **no** form, because that command causes all commands that refer to that name to be deleted.

Step 5 (Routed Mode) Set the security level:

security-level *number*

Example:

```
ciscoasa(config-if)# security-level 50
```

The *number* is an integer between 0 (lowest) and 100 (highest).

Example

The following example sets the BVI 2 address and standby address:

```
ciscoasa(config)# interface bvi 2
ciscoasa(config-if)# ip address 10.1.3.1 255.255.255.0 standby 10.1.3.2
ciscoasa(config-if)# nameif inside
ciscoasa(config-if)# security-level 100
```

Configure General Bridge Group Member Interface Parameters

This procedure describes how to set the name, security level, and bridge group for each bridge group member interface.

Before you begin

- The same bridge group can include different types of interfaces: physical interfaces, VLAN subinterfaces, VNI interfaces, and EtherChannels. The Management interface is not supported. In routed mode, EtherChannels and VNIs are not supported.
- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the **changeto context** *name* command.

- For transparent mode, do not use this procedure for Management interfaces; see Configure a Management Interface for Transparent Mode, on page 681 to configure the Management interface.

Procedure

Step 1 Enter interface configuration mode:

interface *id*

Example:

```
ciscoasa(config)# interface gigabithethernet 0/0
```

The interface ID can be:

- **port-channel**
- *physical*—For example, **ethernet**, **gigabitethernet**, **tengigabitethernet**. Management interfaces are not supported. Refer to the hardware installation guide for your model for interface names.
- *physical_or_port-channel.subinterface*—For example, **gigabitethernet0/0.100** or **port-channel1.100**.
- **vni**
- **vlan**
- *mapped_name*—For multiple context mode.

Note
For the Firepower 1010, you cannot configure switch ports as bridge group members.

You cannot mix logical VLAN interfaces and physical router interfaces in the same bridge group.

Note
In routed mode, the **port-channel** and **vni** interfaces are not supported as bridge group members.

Step 2 Assign the interface to a bridge group:

bridge-group *number*

Example:

```
ciscoasa(config-if)# bridge-group 1
```

The *number* is an integer between 1 and 250, and must match the BVI interface number. You can assign up to 64 interfaces to a bridge group. You cannot assign the same interface to more than one bridge group.

Step 3 Name the interface:

nameif *name*

Example:

```
ciscoasa(config-if)# nameif inside1
```

The *name* is a text string up to 48 characters, and is not case-sensitive. You can change the name by reentering this command with a new value. Do not enter the **no** form, because that command causes all commands that refer to that name to be deleted.

Step 4 Set the security level:

security-level *number*

Example:

```
ciscoasa(config-if)# security-level 50
```

The *number* is an integer between 0 (lowest) and 100 (highest)..

Related Topics

Configure the MTU and TCP MSS, on page 713

Configure a Management Interface for Transparent Mode

In transparent firewall mode, all interfaces must belong to a bridge group. The only exception is the Management interface (either the physical interface, a subinterface (if supported for your model), or an EtherChannel interface comprised of Management interfaces (if you have multiple Management interfaces)) which you can configure as a separate management interface; for the Firepower 4100/9300 chassis, the management interface ID depends on the mgmt-type interface that you assigned to the ASA logical device. You cannot use any other interface types as management interfaces. You can configure one management interface in single mode or per context. For more information see Management Interface for Transparent Mode, on page 595.

The Management interface is for to- and from-the-box traffic only and cannot pass through traffic.

Before you begin

- Do not assign this interface to a bridge group; a non-configurable bridge group (ID 301) is automatically added to your configuration. This bridge group is not included in the bridge group limit.
- For the Firepower 4100/9300 chassis, the management interface ID depends on the mgmt-type interface that you assigned to the ASA logical device.
- In multiple context mode, you cannot share any interfaces, including the Management interface, across contexts. You must connect to a data interface.
- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the **changeto context** *name* command.

Procedure

Step 1 Enter interface configuration mode:

interface {{**port-channel** *number* | **management** *slot*/*port* | *mgmt-type_interface_id* }[**.** *subinterface*] | *mapped_name*}

Example:

```
ciscoasa(config)# interface management 0/0.1
```

The **port-channel** *number* argument is the EtherChannel interface ID, such as **port-channel 1**. The EtherChannel interface must have only Management member interfaces.

In multiple context mode, enter the *mapped_name* if one was assigned using the **allocate-interface** command.

For the Firepower 4100/9300 chassis, specify the interface ID for the mgmt type interface (individual or EtherChannel) that you assigned to the ASA logical device.

Step 2 Name the interface:

nameif *name*

Example:

```
ciscoasa(config-if)# nameif management
```

The *name* is a text string up to 48 characters, and is not case-sensitive. You can change the name by reentering this command with a new value. Do not enter the **no** form, because that command causes all commands that refer to that name to be deleted.

Step 3 Set the IP address using one of the following methods.

- Set the IP address manually:

  For use with failover, you must set the IP address and standby address manually; DHCP is not supported.

  The *ip_address* and *mask* arguments set the interface IP address and subnet mask.

  The standby *ip_address* argument is used for failover.

  ip address *ip_address* [*mask*] [**standby** *ip_address*]

  Example:

  ```
  ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0 standby 10.1.1.2
  ```

- Obtain an IP address from a DHCP server:

  ip address dhcp [**setroute**]

  Example:

  ```
  ciscoasa(config-if)# ip address dhcp
  ```

  The **setroute** keyword lets the ASA use the default route supplied by the DHCP server.

  Reenter this command to reset the DHCP lease and request a new lease.

  If you do not enable the interface using the no shutdown command before you enter the **ip address dhcp** command, some DHCP requests might not be sent.

Step 4 Set the security level:

security-level *number*

Example:

```
ciscoasa(config-if)# security-level 100
```

The *number* is an integer between 0 (lowest) and 100 (highest).

Configure IPv6 Addressing

This section describes how to configure IPv6 addressing.

About IPv6

This section includes information about IPv6.

IPv6 Addressing

You can configure two types of unicast addresses for IPv6:

- Global—The global address is a public address that you can use on the public network. For a bridge group, this address needs to be configured for the BVI, and not per member interface. You can also configure a global IPv6 address for the management interface in transparent mode.
- Link-local—The link-local address is a private address that you can only use on the directly-connected network. Routers do not forward packets using link-local addresses; they are only for communication on a particular physical network segment. They can be used for address configuration or for the Neighbor Discovery functions such as address resolution. In a bridge group, only member interfaces have link-local addresses; the BVI does not have a link-local address.

At a minimum, you need to configure a link-local address for IPv6 to operate. If you configure a global address, a link-local address is automatically configured on the interface, so you do not also need to specifically configure a link-local address. For bridge group member interfaces, when you configure the global address on the BVI, the ASA automatically generates link-local addresses for member interfaces. If you do not configure a global address, then you need to configure the link-local address, either automatically or manually.

Note If you want to only configure the link-local addresses, see the **ipv6 enable** (to auto-configure) or **ipv6 address link-local** (to manually configure) command in the command reference.

Modified EUI-64 Interface IDs

RFC 3513: Internet Protocol Version 6 (IPv6) Addressing Architecture requires that the interface identifier portion of all unicast IPv6 addresses, except those that start with binary value 000, be 64 bits long and be constructed in Modified EUI-64 format. The ASA can enforce this requirement for hosts attached to the local link.

When this feature is enabled on an interface, the source addresses of IPv6 packets received on that interface are verified against the source MAC addresses to ensure that the interface identifiers use the Modified EUI-64

format. If the IPv6 packets do not use the Modified EUI-64 format for the interface identifier, the packets are dropped and the following system log message is generated:

```
325003: EUI-64 source address check failed.
```

The address format verification is only performed when a flow is created. Packets from an existing flow are not checked. Additionally, the address verification can only be performed for hosts on the local link.

Configure the IPv6 Prefix Delegation Client

The ASA can act as a DHPCv6 Prefix Delegation client so that the client interface, for example the outside interface connected to a cable modem, can receive one or more IPv6 prefixes that the ASA can then subnet and assign to its inside interfaces.

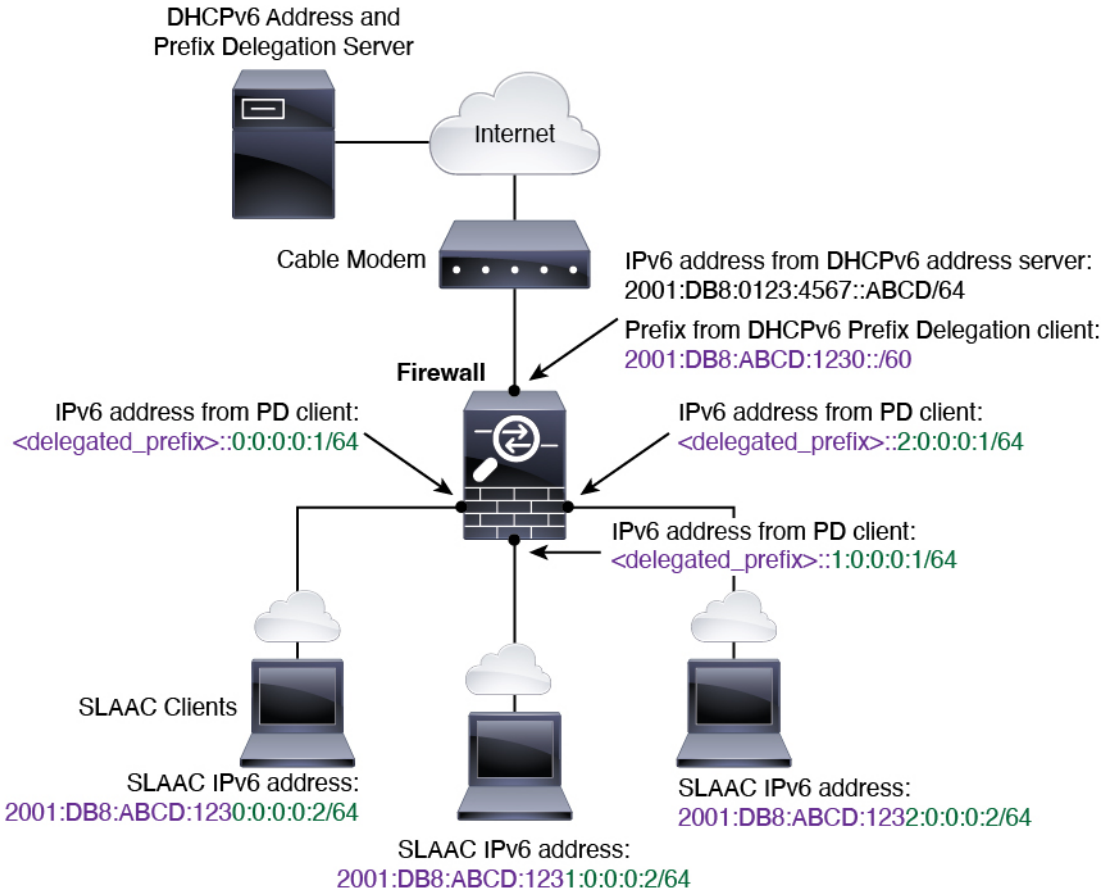
About IPv6 Prefix Delegation

The ASA can act as a DHPCv6 Prefix Delegation client so that the client interface, for example the outside interface connected to a cable modem, can receive one or more IPv6 prefixes that the ASA can then subnet and assign to its inside interfaces. Hosts connected to the inside interfaces can then use StateLess Address Auto Configuration (SLAAC) to obtain global IPv6 addresses. Note that the inside ASA interfaces do not in turn act as Prefix Delegation servers; the ASA can only provide global IP addresses to SLAAC clients. For example, if a router is connected to the ASA, it can act as a SLAAC client to obtain its IP address. But if you want to use a subnet of the delegated prefix for the networks behind the router, you must manually configure those addresses on the router's inside interfaces.

The ASA includes a light DHCPv6 server so the ASA can provide information such as the DNS server and domain name to SLAAC clients when they send Information Request (IR) packets to the ASA. The ASA only accepts IR packets, and does not assign addresses to the clients. You will configure the client to generate its own IPv6 address by enabling IPv6 autoconfiguration on the client. Enabling stateless autoconfiguration on a client configures IPv6 addresses based on prefixes received in Router Advertisement messages; in other words, based on the prefix that the ASA received using Prefix Delegation.

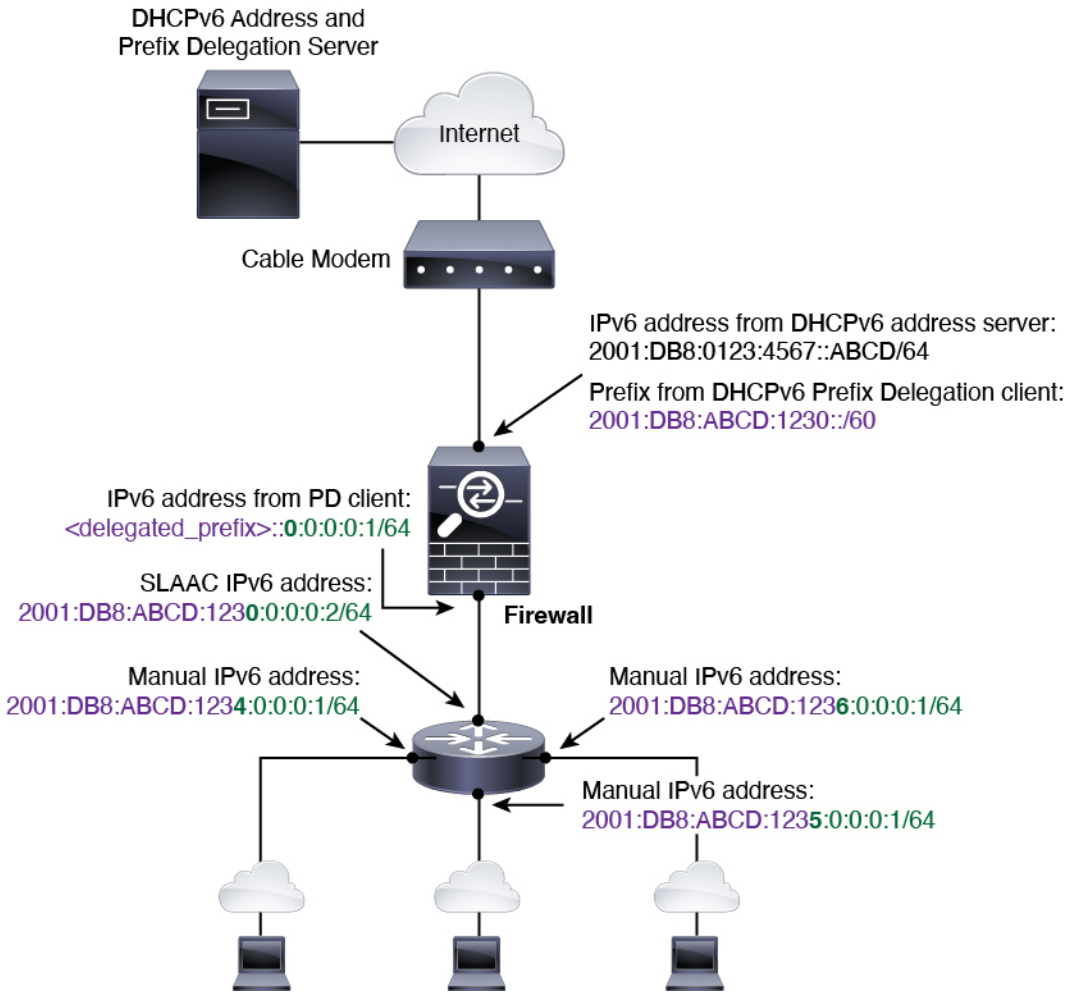
IPv6 Prefix Delegation /64 Subnet Example

The following example shows the ASA receiving an IP address on the outside interface using the DHCPv6 address client. It also gets a delegated prefix using the DHCPv6 Prefix Delegation client. The ASA subnets the delegated prefix into /64 networks and assigns global IPv6 addresses to its inside interfaces dynamically using the delegated prefix plus a manually configured subnet (::0, ::1, or ::2) and IPv6 address (0:0:0:1) per interface. SLAAC clients connected to those inside interfaces obtain IPv6 addresses on each /64 subnet.



IPv6 Prefix Delegation /62 Subnet Example

The following example shows the ASA subnetting the prefix into 4 /62 subnets: 2001:DB8:ABCD:123**0**::/62, 2001:DB8:ABCD:123**4**::/62, 2001:DB8:ABCD:123**8**::/62, and 2001:DB8:ABCD:123**C**::/62. The ASA uses one of 4 available /64 subnets on 2001:DB8:ABCD:1230::/62 for its inside network (::0). You can then manually use additional /62 subnets for downstream routers. The router shown uses 3 of 4 available /64 subnets on 2001:DB8:ABCD:1234::/62 for its inside interfaces (::4, ::5, and ::6). In this case, the inside router interfaces cannot dynamically obtain the delegated prefix, so you need to view the delegated prefix on the ASA, and then use that prefix for your router configuration. Usually, ISPs delegate the same prefix to a given client when the lease expires, but if the ASA receives a new prefix, you will have to modify the router configuration to use the new prefix.



Enable the IPv6 Prefix Delegation Client

Enable the DHCPv6 Prefix Delegation client on one or more interfaces. The ASA obtains one or more IPv6 prefixes that it can subnet and assign to inside networks. Typically, the interface on which you enable the prefix delegation client obtains its IP address using the DHCPv6 address client; only other ASA interfaces use addresses derived from the delegated prefix.

Before you begin

- This feature is only supported in routed firewall mode.
- This feature is not supported in multiple context mode.
- This feature is not supported in clustering.
- You cannot configure this feature on a management-only interface.
- When you use Prefix Delegation, you must set the ASA IPv6 neighbor discovery router advertisement interval to be much lower than the preferred lifetime of the prefix assigned by the DHCPv6 Server to prevent IPv6 traffic interruption. For example, if the DHCPv6 server sets the preferred Prefix Delegation lifetime to 300 seconds, you should set the ASA RA interval to be 150 seconds. To set the preferred

lifetime, use the **show ipv6 general-prefix** command. To set the ASA RA interval, see Configure IPv6 Neighbor Discovery, on page 690; the default is 200 seconds.

Procedure

Step 1 Enter interface configuration mode for the interface connected to the DHCPv6 server network:

interface *id*

Example:

```
ciscoasa(config)# interface gigabithethernet 0/0
ciscoasa(config-if)#
```

Step 2 Enable the DHCPv6 Prefix Delegation client, and name the prefix(es) obtained on this interface:

ipv6 dhcp client pd *name*

Example:

```
ciscoasa(config-if)# ipv6 dhcp client pd Outside-Prefix
```

The *name* can be up to 200 characters.

Step 3 Provide one or more hints about the delegated prefix you want to receive:

ipv6 dhcp client pd hint *ipv6_prefix/prefix_length*

Example:

```
ciscoasa(config-if)# ipv6 dhcp client pd hint 2001:DB8:ABCD:1230::/60
```

Typically you want to request a particular prefix length, such as ::/60, or if you have received a particular prefix before and want to ensure you get it again when the lease expires, you can enter the whole prefix as the hint. If you enter multiple hints (different prefixes or lengths), then it is up to the DHCP server which hint to honor, or whether to honor the hint at all.

Step 4 See Configure a Global IPv6 Address, on page 688 to assign a subnet of the prefix as the global IP address for an ASA interface.

Step 5 (Optional) See Configure the DHCPv6 Stateless Server, on page 767 to provide domain-name and server parameters to SLAAC clients.

Step 6 (Optional) See Configure IPv6 Network Settings, on page 926 to advertise the prefix(es) with BGP.

Example

The following example configures the DHCPv6 address client and prefix delegation client on GigabitEthernet 0/0, then assigns addresses with the prefix on GigabitEthernet 0/1 and 0/2:

```
interface gigabitethernet 0/0
```

```
  ipv6 address dhcp default
  ipv6 dhcp client pd Outside-Prefix
  ipv6 dhcp client pd hint ::/60
interface gigabitethernet 0/1
  ipv6 address Outside-Prefix ::1:0:0:0:1/64
interface gigabitethernet 0/2
  ipv6 address Outside-Prefix ::2:0:0:0:1/64
```

Configure a Global IPv6 Address

To configure a global IPv6 address for any routed mode interface and for the transparent or routed mode BVI, perform the following steps.

DHCPv6 and prefix delegation options are not supported with multiple context mode.

Note Configuring the global address automatically configures the link-local address, so you do not need to configure it separately. For bridge groups, configuring the global address on the BVI automatically configures link-local addresses on all member interfaces.

For subinterfaces, we recommend that you also set the MAC address manually, because they use the same burned-in MAC address of the parent interface. IPv6 link-local addresses are generated based on the MAC address, so assigning unique MAC addresses to subinterfaces allows for unique IPv6 link-local addresses, which can avoid traffic disruption in certain instances on the ASA. See Manually Configure the MAC Address, on page 711.

Before you begin

- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the **changeto context** *name* command.

Procedure

Step 1 Enter interface configuration mode:

interface *id*

Example:

```
ciscoasa(config)# interface gigabithethernet 0/0
```

In transparent mode or for a bridge group in routed mode, specify the BVI:

Example:

```
ciscoasa(config)# interface bvi 1
```

In transparent mode, in addition to the BVI, you can also specify a Management interface:

Example:

```
ciscoasa(config)# interface management 1/1
```

Step 2 (Routed interface) Set the IP address using one of the following methods.

For failover and clustering, you must set the IP address manually. For clustering, manually configuring the link-local address is not supported.

- Enable StateLess Address Auto Configuration (SLAAC) on the interface:

  ipv6 address autoconfig [**default trust** {**dhcp** | **ignore**}]

  Enabling SLAAC on the interface configures IPv6 addresses based on prefixes received in Router Advertisement messages. A link-local address, based on the Modified EUI-64 interface ID, is automatically generated for the interface whenSLAAC is enabled.

  Note
  Although RFC 4862 specifies that hosts configured for SLAAC do not send Router Advertisement messages, the ASA does send Router Advertisement messages in this case. See the **ipv6 nd suppress-ra** command to suppress messages.

  If you want to install a default route, specify **default trust dhcp** or **ignore**. **dhcp** specifies the ASA only uses a default route from Router Advertisements that come from a trusted source (in other words, from the same server that provided the IPv6 address). **ignore** specifies that Router Advertisements can be sourced from another network, which can be a riskier method.

- Obtain an address using DHCPv6:

  ipv6 address dhcp [**default**]

  Example:

  ```
  ciscoasa(config-if)# ipv6 address dhcp default
  ```

  The **default** keyword obtains a default route from Router Advertisements.

- Manually assign a global address to the interface:

  ipv6 address *ipv6_address*/*prefix-length* [**standby** *ipv6_address*]

  Example:

  ```
  ciscoasa(config-if)# ipv6 address 2001:0DB8:BA98::3210/64 standby 2001:0DB8:BA98::3211
  ```

  When you assign a global address, the link-local address is automatically created for the interface.

  standby specifies the interface address used by the secondary unit or failover group in a failover pair.

- Assign a global address to the interface by combining the specified prefix with an interface ID generated from the interface MAC address using the Modified EUI-64 format:

  ipv6 address *ipv6-prefix*/*prefix-length* **eui-64**

  Example:

  ```
  ciscoasa(config-if)# ipv6 address 2001:0DB8:BA98::/64 eui-64
  ```

When you assign a global address, the link-local address is automatically created for the interface.

You do not need to specify the standby address; the interface ID will be generated automatically.

- Use a delegated prefix:

  ipv6 address *prefix_name ipv6_address/prefix_length*

  Example:

  ```
  ciscoasa(config-if)# ipv6 address Outside-Prefix ::1:0:0:0:1/64
  ```

  This feature requires the ASA to have the DHCPv6 Prefix Delegation client enabled *on a different interface*. See Enable the IPv6 Prefix Delegation Client, on page 686. Typically, the delegated prefix will be /60 or smaller so you can subnet to multiple /64 networks. /64 is the supported subnet length if you want to support SLAAC for connected clients. You should specify an address that completes the /60 subnet, for example ::1:0:0:0:1. Enter :: before the address in case the prefix is smaller than /60. For example, if the delegated prefix is 2001:DB8:1234:5670::/60, then the global IP address assigned to this interface is 2001:DB8:1234:5671::1/64. The prefix that is advertised in router advertisements is 2001:DB8:1234:5671::/64. In this example, if the prefix is smaller than /60, the remaining bits of the prefix will be 0's as indicated by the leading ::. For example, if the prefix is 2001:DB8:1234::/48, then the IPv6 address will be 2001:DB8:1234::1:0:0:0:1/64.

Step 3 (BVI interface) Manually assign a global address to the BVI. For a management interface in Transparent mode, use this method as well.

ipv6 address *ipv6_address*/*prefix-length* [**standby** *ipv6_address*]

Example:

```
ciscoasa(config-if)# ipv6 address 2001:0DB8::BA98:0:3210/48
```

When you assign a global address, the link-local address is automatically created for the interface.

standby specifies the interface address used by the secondary unit or failover group in a failover pair.

Step 4 (Optional) Enforce the use of Modified EUI-64 format interface identifiers in IPv6 addresses on a local link:

ipv6 enforce-eui64 *if_name*

Example:

```
ciscoasa(config)# ipv6 enforce-eui64 inside
```

The *if_name* argument is the name of the interface, as specified by the **nameif** command, on which you are enabling the address format enforcement.

Configure IPv6 Neighbor Discovery

The IPv6 neighbor discovery process uses ICMPv6 messages and solicited-node multicast addresses to determine the link-layer address of a neighbor on the same network (local link), verify the readability of a neighbor, and keep track of neighboring routers.

Nodes (hosts) use neighbor discovery to determine the link-layer addresses for neighbors known to reside on attached links and to quickly purge cached values that become invalid. Hosts also use neighbor discovery to find neighboring routers that are willing to forward packets on their behalf. In addition, nodes use the protocol to actively keep track of which neighbors are reachable and which are not, and to detect changed link-layer addresses. When a router or the path to a router fails, a host actively searches for functioning alternates.

Procedure

Step 1 Specify the IPv6 interface you want to configure.

interface *name*

Example:

```
ciscoasa(config)# interface gigabitethernet 0/0
ciscoasa(config-if)#
```

Step 2 Specify the number Duplicate Address Detection (DAD) attempts.

ipv6 nd dad attempts *value*

Valid values for the *value* argument range from 0 to 600. A 0 value disables DAD processing on the specified interface. The default is 1 message.

DAD ensures the uniqueness of new unicast IPv6 addresses before they are assigned, and ensures that duplicate IPv6 addresses are detected in the network on a link basis. The ASA uses neighbor solicitation messages to perform DAD.

When a duplicate address is identified, the state of the address is set to DUPLICATE, the address is not used, and the following error message is generated:

```
325002: Duplicate address ipv6_address/MAC_address on interface
```

If the duplicate address is the link-local address of the interface, the processing of IPv6 packets is disabled on the interface. If the duplicate address is a global address, the address is not used.

Example:

```
ciscoasa(config-if)# ipv6 nd dad attempts 20
```

Step 3 Set the interval between IPv6 neighbor solicitation retransmissions.

ipv6 nd ns-interval *value*

Values for the *value* argument range from 1000 to 3600000 milliseconds.

Neighbor solicitation messages (ICMPv6 Type 135) are sent on the local link by nodes attempting to discover the link-layer addresses of other nodes on the local link. After receiving a neighbor solicitation message, the destination node replies by sending a neighbor advertisement message (ICPMv6 Type 136) on the local link.

After the source node receives the neighbor advertisement, the source node and destination node can communicate. Neighbor solicitation messages are also used to verify the reachability of a neighbor after the link-layer address of a neighbor is identified. When a node wants to verifying the reachability of a neighbor, the destination address in a neighbor solicitation message is the unicast address of the neighbor.

Neighbor advertisement messages are also sent when there is a change in the link-layer address of a node on a local link.

Example:

```
ciscoasa(config-if)# ipv6 nd ns-interval 9000
```

Step 4 Set the amount of time that a remote IPv6 node is reachable.

ipv6 nd reachable-time *value*

Values for the *value* argument range from 0 to 3600000 milliseconds. When 0 is used for the value, the reachable time is sent as undetermined. It is up to the receiving devices to set and track the reachable time value.

The neighbor reachable time enables detecting unavailable neighbors. Shorter configured times enable detecting unavailable neighbors more quickly, however, shorter times consume more IPv6 network bandwidth and processing resources in all IPv6 network devices. Very short configured times are not recommended in normal IPv6 operation.

Example:

```
ciscoasa config-if)# ipv6 nd reachable-time 1700000
```

Step 5 Set the interval between IPv6 router advertisement transmissions.

ipv6 nd ra-interval [**msec**] *value*

The **msec** keyword indicates that the value provided is in milliseconds. If this keyword is not present, the value provided is in seconds. Valid values for the *value* argument range from 3 to 1800 seconds or from 500 to 1800000 milliseconds if the **msec** keyword is provided. The default is 200 seconds.

The interval value is included in all IPv6 router advertisements that are sent out of this interface.

The interval between transmissions should be less than or equal to the IPv6 router advertisement lifetime if the ASA is configured as a default router.

To prevent synchronization with other IPv6 nodes, the ASA randomly adjusts the value that you set (jitter).

Example:

```
ciscoasa(config-if)# ipv6 nd ra-interval 201
```

Step 6 Specify the length of time that nodes on the local link should consider the ASA as the default router on the link.

ipv6 nd ra-lifetime [**msec**] *value*

The optional **msec** keyword indicates that the value provided is in milliseconds. Otherwise, the value is in seconds. Values for the *value* argument range from 0 to 9000 seconds. Entering 0 indicates that the ASA should not be considered a default router on the selected interface.

The router lifetime value is included in all IPv6 router advertisements sent out of the interface. The value indicates the usefulness of the ASA as a default router on this interface.

Example:

```
ciscoasa(config-if)# ipv6 nd ra-lifetime 2000
```

Step 7 Suppress router advertisements.

ipv6 nd suppress-ra

Router advertisement messages (ICMPv6 Type 134) are automatically sent in response to router solicitation messages (ICMPv6 Type 133). Router solicitation messages are sent by hosts at system startup so that the host can immediately autoconfigure without needing to wait for the next scheduled router advertisement message.

You may want to disable these messages on any interface for which you do not want the ASA to supply the IPv6 prefix (for example, the outside interface).

Entering this command causes the ASA to appear as a regular IPv6 neighbor on the link and not as an IPv6 router.

Step 8 Add a flag to IPv6 router advertisements to inform IPv6 StateLess Address Auto Configuration (SLAAC) clients to use DHCPv6 to obtain an IPv6 address, in addition to the derived SLAAC address.

ipv6 nd managed-config-flag

This option sets the Managed Address Config flag in the IPv6 router advertisement packet.

Step 9 Add a flag to IPv6 router advertisements to inform IPv6 SLAAC clients to use DHCPv6 to obtain the DNS server address, or other information.

ipv6 nd other-config-flag

This option sets the Other Address Config flag in the IPv6 router advertisement packet.

Step 10 Configure which IPv6 prefixes are included in IPv6 router advertisements:

ipv6 nd prefix {*ipv6_prefix*/*prefix_length* | **default**} [*valid_lifetime preferred_lifetime* | **at** *valid_date preferred_date*] [**no-advertise**] [**no-autoconfig**] [ ] [**off-link**]

The prefix advertisement can be used by neighboring devices to autoconfigure their interface addresses. SLAAC uses IPv6 prefixes provided in router advertisement messages to create the global unicast address from the link-local address.

By default, prefixes configured as addresses on an interface using the **ipv6 address** command are advertised in router advertisements. If you configure prefixes for advertisement using the **ipv6 nd prefix** command, then only these prefixes are advertised.

For SLAAC to work correctly, the advertised prefix length in router advertisement messages must always be 64 bits.

- **default**—Indicates that the default prefix is used.
- *valid_lifetime preferred_lifetime* —Specifies the amount of time that the specified IPv6 prefix is advertised as being valid and preferred. An address has no restrictions during the preferred lifetime. After the preferred lifetime expires, the address goes into a deprecated state; while an address is in a deprecated state, its use is discouraged, but not strictly forbidden. After the valid lifetime expires, the address becomes invalid and cannot be used. The valid lifetime must be greater than or equal to the preferred lifetime. Values range from 0 to 4294967295 seconds. The maximum value represents infinity, which can also be specified with the **infinite** keyword. The valid lifetime default is 2592000 (30 days). The preferred lifetime default is 604800 (7 days).

- **at** *valid_date preferred_date*—Indicates a specific date and time at which the prefix expires. Specify the date as the *month_name day hh*:*mm*. For example, enter **dec 1 13:00**.
- **no-advertise**—Disables advertisement of the prefix. For the **default** prefix, this setting only applies to on-link prefixes. Off-link prefixes will still be advertised unless you specify **no-advertise** for a specific off-link prefix.
- **no-autoconfig**—Specifies that the prefix cannot be used for IPv6 SLAAC.
- **off-link**—Configures the specified prefix as off-link. The prefix will be advertised with the L-bit clear. The prefix will not be inserted into the routing table as a Connected prefix.

  When onlink is on (by default), the specified prefix is assigned to the link. Nodes sending traffic to such addresses that contain the specified prefix consider the destination to be locally reachable on the link.

Example:

```
ciscoasa(config-if)# ipv6 nd prefix 2001:DB8::/32 1000 900
```

Step 11 Configure a static entry in the IPv6 neighbor discovery cache.

ipv6 neighbor *ipv6_address if_name mac_address*

The following guidelines and limitations apply for configuring a static IPv6 neighbor:

- The **ipv6 neighbor** command is similar to the **arp** command. If an entry for the specified IPv6 address already exists in the neighbor discovery cache—learned through the IPv6 neighbor discovery process—the entry is automatically converted to a static entry. These entries are stored in the configuration when the copy command is used to store the configuration.
- Use the **show ipv6 neighbor** command to view static entries in the IPv6 neighbor discovery cache.
- The **clear ipv6 neighbor** command deletes all entries in the IPv6 neighbor discovery cache except static entries. The **no ipv6 neighbor** command deletes a specified static entry from the neighbor discovery cache; the command does not remove dynamic entries—entries learned from the IPv6 neighbor discovery process—from the cache. Disabling IPv6 on an interface by using the **no ipv6 enable** command deletes all IPv6 neighbor discovery cache entries configured for that interface except static entries (the state of the entry changes to INCMP [Incomplete]).
- Static entries in the IPv6 neighbor discovery cache are not modified by the neighbor discovery process.
- The **clear ipv6 neighbor** command does not remove static entries from the IPv6 neighbor discovery cache; it only clears the dynamic entries.
- The ICMP syslogs generated are caused by a regular refresh of IPv6 neighbor entries. The ASA default timer for IPv6 neighbor entry is 30 seconds, so the ASA would generate ICMPv6 neighbor discovery and response packets about every 30 seconds. If the ASA has both failover LAN and state interfaces configured with IPv6 addresses, then every 30 seconds, ICMPv6 neighbor discovery and response packets will be generated by both ASAs for both configured and link-local IPv6 addresses. In addition, each packet will generate several syslogs (ICMP connection and local-host creation or teardown), so it may appear that constant ICMP syslogs are being generated. The refresh time for IPV6 neighbor entry is configurable on the regular data interface, but not configurable on the failover interface. However, the CPU impact for this ICMP neighbor discovery traffic is minimal.

Example:

```
ciscoasa(config)# ipv6 neighbor 3001:1::45A inside 002.7D1A.9472
```

Monitoring Routed and Transparent Mode Interfaces

You can monitor interface statistics, status, PPPoE.

Note For Firepower and Secure Firewall models, some statistics are not shown using the ASA commands. You must view more detailed interface statistics using FXOS commands.

- /eth-uplink/fabric# **show interface**
- /eth-uplink/fabric# **show port-channel**
- /eth-uplink/fabric/interface# **show stats**

For the Firepower 2100 in Platform mode, see also the following FXOS connect local-mgmt commands:

- (local-mgmt)# **show portmanager counters**
- (local-mgmt)# **show lacp**
- (local-mgmt)# **show portchannel**

See the FXOS troubleshooting guide for more information.

Interface Statistics and Information

- **show interface**

  Displays interface statistics.

- **show interface ip brief**

  Displays interface IP addresses and status.

- **show bridge-group**

  Displays bridge group information such as interfaces assigned, MAC addresses, and IP addresses.

DHCP Information

- **show ipv6 dhcp interface** [*ifc_name* [**statistics**]]

  The **show ipv6 dhcp interface** command displays DHCPv6 information for all interfaces. If the interface is configured for DHCPv6 stateless server configuration (see Configure the DHCPv6 Stateless Server, on page 767), this command lists the DHCPv6 pool that is being used by the server. If the interface has DHCPv6 address client or Prefix Delegation client configuration, this command shows the state of each

client and the values received from the server. For a specific interface, you can show message statistics for the DHCP server or client. The following examples show information provided by this command:

```
ciscoasa(config-if)# show ipv6 dhcp interface
GigabitEthernet1/1 is in server mode
  Using pool: Sample-Pool

GigabitEthernet1/2 is in client mode
  Prefix State is OPEN
  Renew will be sent in 00:03:46
  Address State is OPEN
  Renew for address will be sent in 00:03:47
  List of known servers:
    Reachable via address: fe80::20c:29ff:fe96:1bf4
    DUID: 000100011D9D1712005056A07E06
    Preference: 0
    Configuration parameters:
      IA PD: IA ID 0x00030001, T1 250, T2 400
        Prefix: 2005:abcd:ab03::/48
                preferred lifetime 500, valid lifetime 600
                expires at Nov 26 2014 03:11 PM (577 seconds)
      IA NA: IA ID 0x00030001, T1 250, T2 400
        Address: 2004:abcd:abcd:abcd:abcd:abcd:abcd:f2cb/128
                preferred lifetime 500, valid lifetime 600
                expires at Nov 26 2014 03:11 PM (577 seconds)
      DNS server: 2004:abcd:abcd:abcd::2
      DNS server: 2004:abcd:abcd:abcd::4
      Domain name: relay.com
      Domain name: server.com
      Information refresh time: 0
  Prefix name: Sample-PD

Management1/1 is in client mode
  Prefix State is IDLE
  Address State is OPEN
  Renew for address will be sent in 11:26:44
  List of known servers:
    Reachable via address: fe80::4e00:82ff:fe6f:f6f9
    DUID: 000300014C00826FF6F8
    Preference: 0
    Configuration parameters:
      IA NA: IA ID 0x000a0001, T1 43200, T2 69120
        Address: 2308:2308:210:1812:2504:1234:abcd:8e5a/128
                preferred lifetime INFINITY, valid lifetime INFINITY
      Information refresh time: 0


ciscoasa(config-if)# show ipv6 dhcp interface outside statistics

DHCPV6 Client PD statistics:

Protocol Exchange Statistics:


 Number of Solicit messages sent:             1
 Number of Advertise messages received:       1
 Number of Request messages sent:             1
 Number of Renew messages sent:               45
 Number of Rebind messages sent:              0
 Number of Reply messages received:           46
 Number of Release messages sent:             0
 Number of Reconfigure messages received:     0
```

```
 Number of Information-request messages sent:  0

Error and Failure Statistics:


 Number of Re-transmission messages sent:                 1
 Number of Message Validation errors in received messages: 0

DHCPV6 Client address statistics:

Protocol Exchange Statistics:


 Number of Solicit messages sent:              1
 Number of Advertise messages received:        1
 Number of Request messages sent:              1
 Number of Renew messages sent:                45
 Number of Rebind messages sent:               0
 Number of Reply messages received:            46
 Number of Release messages sent:              0
 Number of Reconfigure messages received:      0
 Number of Information-request messages sent:  0

Error and Failure Statistics:


 Number of Re-transmission messages sent:                 1
 Number of Message Validation errors in received messages: 0
```

- **show ipv6 dhcp client** [**pd**] **statistics**

  The **show ipv6 dhcp client statistics** command shows DHCPv6 client statistics and shows the output of the number of messages sent and received. The **show ipv6 dhcp client pd statistics** command shows the Prefix Delegation client statistics. The following examples show information provided by this command:

```
ciscoasa(config)# show ipv6 dhcp client statistics

Protocol Exchange Statistics:
  Total number of Solicit messages sent:              4
  Total number of Advertise messages received:        4
  Total number of Request messages sent:              4
  Total number of Renew messages sent:                92
  Total number of Rebind messages sent:               0
  Total number of Reply messages received:            96
  Total number of Release messages sent:              6
  Total number of Reconfigure messages received:      0
  Total number of Information-request messages sent:  0

Error and Failure Statistics:
  Total number of Re-transmission messages sent:                   8
  Total number of Message Validation errors in received messages: 0


ciscoasa(config)# show ipv6 dhcp client pd statistics

Protocol Exchange Statistics:


 Total number of Solicit messages sent:              1
 Total number of Advertise messages received:        1
```

```
 Total number of Request messages sent:              1
 Total number of Renew messages sent:                92
 Total number of Rebind messages sent:               0
 Total number of Reply messages received:            93
 Total number of Release messages sent:              0
 Total number of Reconfigure messages received:      0
 Total number of Information-request messages sent:  0

Error and Failure Statistics:


 Total number of Re-transmission messages sent:                  1
 Total number of Message Validation errors in received messages: 0
```

- **show ipv6 dhcp ha statistics**

  The **show ipv6 dhcp ha statistics** command shows the transaction statistics between failover units, including how many times the DUID information was synced between the units. The following examples show information provided by this command.

  On an active unit:

```
ciscoasa(config)# show ipv6 dhcp ha statistics

DHCPv6 HA global statistics:
  DUID sync messages sent:              1
  DUID sync messages received:          0

DHCPv6 HA error statistics:
  Send errors:                          0
```

  On an standby unit:

```
ciscoasa(config)# show ipv6 dhcp ha statistics

DHCPv6 HA global statistics:
  DUID sync messages sent:              0
  DUID sync messages received:          1

DHCPv6 HA error statistics:
  Send errors:                          0
```

- **show ipv6 general-prefix**

  The **show ipv6 general-prefix** command shows all the prefixes acquired by the DHCPv6 Prefix Delegation clients and the ASA distribution of that prefix to other processes ("Consumer List"). The following example shows information provided by this command:

```
ciscoasa(config)# show ipv6 general-prefix
IPv6 Prefix Sample-PD, acquired via DHCP PD
  2005:abcd:ab03::/48 Valid lifetime 524, preferred lifetime 424
   Consumer List               Usage count
    BGP network command        1
    inside (Address command)   1
```

PPPoE

- **show ip address** *interface_name* **pppoe**

  Displays the current PPPoE client configuration information.

- **debug pppoe** {**event** | **error** | **packet**}

  Enables debugging for the PPPoE client.

- **show vpdn session** [**l2tp** | **pppoe**] [**id** *sess_id* | **packets** | **state** | **window**]

  Views the status of PPPoE sessions.

  The following examples show information provided by this command:

```
ciscoasa# show vpdn

Tunnel id 0, 1 active sessions
     time since change 65862 secs
     Remote Internet Address 10.0.0.1
    Local Internet Address 199.99.99.3
     6 packets sent, 6 received, 84 bytes sent, 0 received
Remote Internet Address is 10.0.0.1
     Session state is SESSION_UP
       Time since event change 65865 secs, interface outside
       PPP interface id is 1
       6 packets sent, 6 received, 84 bytes sent, 0 received
ciscoasa#
ciscoasa# show vpdn session
PPPoE Session Information (Total tunnels=1 sessions=1)
Remote Internet Address is 10.0.0.1
  Session state is SESSION_UP
    Time since event change 65887 secs, interface outside
    PPP interface id is 1
    6 packets sent, 6 received, 84 bytes sent, 0 received
ciscoasa#
ciscoasa# show vpdn tunnel
PPPoE Tunnel Information (Total tunnels=1 sessions=1)
Tunnel id 0, 1 active sessions
   time since change 65901 secs
   Remote Internet Address 10.0.0.1
   Local Internet Address 199.99.99.3
   6 packets sent, 6 received, 84 bytes sent, 0 received
ciscoasa#
```

IPv6 Neighbor Discovery

To monitor IPv6 neighbor discovery parameters, enter the following commands.

- **show ipv6 interface**

  This command displays the usability status of interfaces configured for IPv6, including the interface name, such as “outside,” and displays the settings for the specified interface. However, it excludes the name from the command and displays the settings for all interfaces that have IPv6 enabled on them. Output for the command shows the following:

  - The name and status of the interface.

- The link-local and global unicast addresses.
- The multicast groups to which the interface belongs.
- ICMP redirect and error message settings.
- Neighbor discovery settings.
- The actual time when the command is set to 0.
- The neighbor discovery reachable time that is being used.

Examples for Routed and Transparent Mode Interfaces

Transparent Mode Example with 2 Bridge Groups

The following example for transparent mode includes two bridge groups of three interfaces each, plus a management-only interface:

```
interface gigabitethernet 0/0
  nameif inside1
  security-level 100
  bridge-group 1
  no shutdown
interface gigabitethernet 0/1
  nameif outside1
  security-level 0
  bridge-group 1
  no shutdown
interface gigabitethernet 0/2
  nameif dmz1
  security-level 50
  bridge-group 1
  no shutdown
interface bvi 1
  ip address 10.1.3.1 255.255.255.0 standby 10.1.3.2

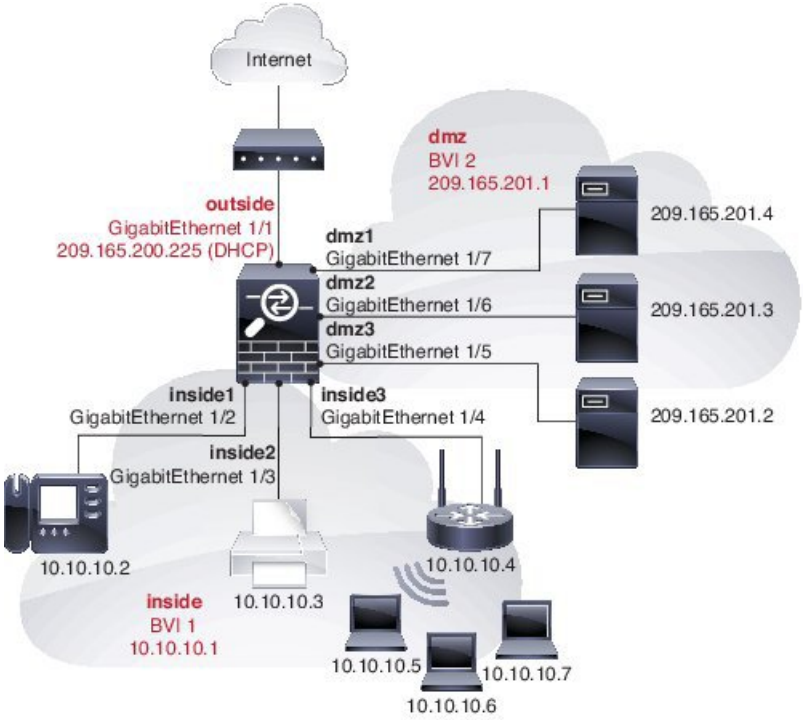
interface gigabitethernet 1/0
  nameif inside2
  security-level 100
  bridge-group 2
  no shutdown
interface gigabitethernet 1/1
  nameif outside2
  security-level 0
  bridge-group 2
  no shutdown
interface gigabitethernet 1/2
  nameif dmz2
  security-level 50
  bridge-group 2
  no shutdown
interface bvi 2
  ip address 10.3.5.8 255.255.255.0 standby 10.3.5.9

interface management 0/0
  nameif mgmt
```

```
  security-level 100
  ip address 10.2.1.1 255.255.255.0 standby 10.2.1.2
  no shutdown
```

Switched LAN Segment Example with 2 Bridge Groups

The following example configures 2 bridge groups with 3 interfaces each and one regular routed interface for outside. Bridge group 1 is inside and bridge group 2 is dmz with public web servers. The bridge group member interfaces can communicate freely within the bridge group because each member is at the same security level, and we enabled same security communication. Although the inside member security level is 100 and the dmz member security level is also 100, these security levels do not apply to inter-BVI communications; only the BVI security levels affect inter-BVI traffic. The security levels of the BVIs and outside (100, 50, and 0) implicitly permit traffic from inside to dmz and inside to outside; and from dmz to outside. An access rule is applied to outside to allow traffic to the servers on dmz.




```
interface gigabitethernet 1/1
  nameif outside
  security-level 0
  ip address dhcp setroute
  no shutdown
!
interface gigabitethernet 1/2
  nameif inside1
  security-level 100
  bridge-group 1
  no shutdown
interface gigabitethernet 1/3
  nameif inside2
  security-level 100
  bridge-group 1
```

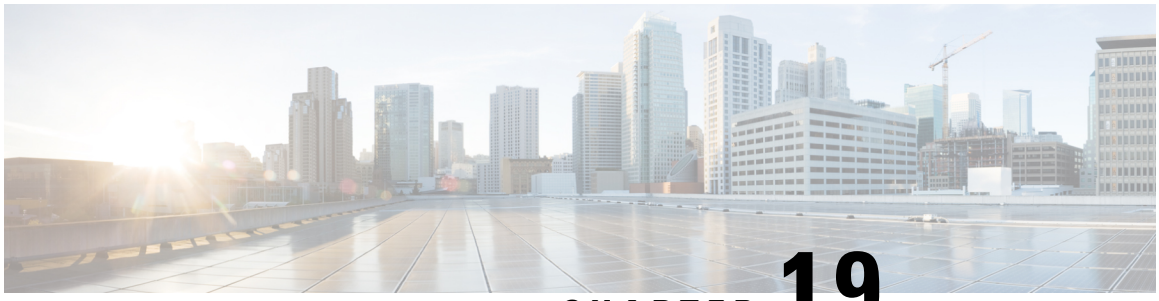
```
  no shutdown
interface gigabitethernet 1/4
  nameif inside3
  security-level 100
  bridge-group 1
  no shutdown
!
interface bvi 1
  nameif inside
  security-level 100
  ip address 10.10.10.1 255.255.255.0
!
interface gigabitethernet 1/5
  nameif dmz1
  security-level 100
  bridge-group 2
  no shutdown
interface gigabitethernet 1/6
  nameif dmz2
  security-level 100
  bridge-group 2
  no shutdown
interface gigabitethernet 1/7
  nameif dmz3
  security-level 100
  bridge-group 2
  no shutdown
!
interface bvi 2
  nameif dmz
  security-level 50
  ip address 209.165.201.1 255.255.255.224
!
same-security-traffic permit inter-interface
!
# Assigns IP addresses to inside hosts
dhcpd address 10.10.10.2-10.10.10.200 inside
dhcpd enable inside
!
# Applies interface PAT for inside traffic going outside
nat (inside1,outside) source dynamic any interface
nat (inside2,outside) source dynamic any interface
nat (inside3,outside) source dynamic any interface
!
# Allows outside traffic to each server for specific applications
object network server1
  host 209.165.201.2
object network server2
  host 209.165.201.3
object network server3
  host 209.165.201.4
!
# Defines mail services allowed on server3
object-group service MAIL
  service-object tcp destination eq pop3
  service-object tcp destination eq imap4
  service-object tcp destination eq smtp
!
# Allows access from outside to servers on the DMZ
access-list SERVERS extended permit tcp any object server1 eq www
access-list SERVERS extended permit tcp any object server2 eq ftp
access-list SERVERS extended permit tcp any object server3 object-group MAIL
access-group SERVERS in interface outside
```

History for Routed and Transparent Mode Interfaces

Feature Name	Platform Releases	Feature Information
IPv6 Neighbor Discovery	7.0(1)	We introduced this feature. We introduced the following commands: **ipv6 nd ns-interval**, **ipv6 nd ra-lifetime**, **ipv6 nd suppress-ra**, **ipv6 neighbor**, **ipv6 nd prefix**, **ipv6 nd dad-attempts**, **ipv6 nd reachable-time**, **ipv6 address**, **ipv6 enforce-eui64**.
IPv6 support for transparent mode	8.2(1)	IPv6 support was introduced for transparent firewall mode.
Bridge groups for transparent mode	8.4(1)	If you do not want the overhead of security contexts, or want to maximize your use of security contexts, you can group interfaces together in a bridge group, and then configure multiple bridge groups, one for each network. Bridge group traffic is isolated from other bridge groups. You can configure up to eight bridge groups of four interfaces each in single mode or per context. We introduced the following commands: **interface bvi**, **show bridge-group**
Address Config Flags for IPv6 DHCP Relay	9.0(1)	We introduced the following commands: **ipv6 nd managed-config-flag**, **ipv6 nd other-config-flag**.
Transparent mode bridge group maximum increased to 250	9.3(1)	The bridge group maximum was increased from 8 to 250 bridge groups. You can configure up to 250 bridge groups in single mode or per context in multiple mode, with 4 interfaces maximum per bridge group. We modified the following commands: **interface bvi**, **bridge-group**
Transparent mode maximum interfaces per bridge group increased to 64	9.6(2)	The maximum interfaces per bridge group was increased from 4 to 64. We did not modify any commands.

Feature Name	Platform Releases	Feature Information
IPv6 DHCP	9.6(2)	The ASA now supports the following features for IPv6 addressing: • DHCPv6 Address client—The ASA obtains an IPv6 global address and optional default route from the DHCPv6 server. • DHCPv6 Prefix Delegation client—The ASA obtains delegated prefix(es) from a DHCPv6 server. The ASA can then use these prefixes to configure other ASA interface addresess so that StateLess Address Auto Configuration (SLAAC) clients can autoconfigure IPv6 addresses on the same network. • BGP router advertisement for delegated prefixes • DHCPv6 stateless server—The ASA provides other information such as the domain name to SLAAC clients when they send Information Request (IR) packets to the ASA. The ASA only accepts IR packets, and does not assign addresses to the clients. We added or modified the following commands: **clear ipv6 dhcp statistics, domain-name, dns-server, import, ipv6 address autoconfig, ipv6 address dhcp, ipv6 dhcp client pd, ipv6 dhcp client pd hint, ipv6 dhcp pool, ipv6 dhcp server, network, nis address, nis domain-name, nisp address, nisp domain-name, show bgp ipv6 unicast, show ipv6 dhcp, show ipv6 general-prefix, sip address, sip domain-name, sntp address**
Integrated Routing and Bridging	9.7(1)	Integrated Routing and Bridging provides the ability to route between a bridge group and a routed interface. A bridge group is a group of interfaces that the ASA bridges instead of routes. The ASA is not a true bridge in that the ASA continues to act as a firewall: access control between interfaces is controlled, and all of the usual firewall checks are in place. Previously, you could only configure bridge groups in transparent firewall mode, where you cannot route between bridge groups. This feature lets you configure bridge groups in routed firewall mode, and to route between bridge groups and between a bridge group and a routed interface. The bridge group participates in routing by using a Bridge Virtual Interface (BVI) to act as a gateway for the bridge group. Integrated Routing and Bridging provides an alternative to using an external Layer 2 switch if you have extra interfaces on the ASA to assign to the bridge group. In routed mode, the BVI can be a named interface and can participate separately from member interfaces in some features, such as access rules and DHCP server. The following features that are supported in transparent mode are not supported in routed mode: multiple context mode, ASA clustering. The following features are also not supported on BVIs: dynamic routing and multicast routing. We modified the following commands: **access-group, access-list ethertype, arp-inspection, dhcpd, mac-address-table static, mac-address-table aging-time, mac-learn, route, show arp-inspection, show bridge-group, show mac-address-table, show mac-learn**

Feature Name	Platform Releases	Feature Information
31-bit Subnet Mask	9.7(1)	For routed interfaces, you can configure an IP address on a 31-bit subnet for point-to-point connections. The 31-bit subnet includes only 2 addresses; normally, the first and last address in the subnet is reserved for the network and broadcast, so a 2-address subnet is not usable. However, if you have a point-to-point connection and do not need network or broadcast addresses, a 31-bit subnet is a useful way to preserve addresses in IPv4. For example, the failover link between 2 ASAs only requires 2 addresses; any packet that is transmitted by one end of the link is always received by the other, and broadcasting is unnecessary. You can also have a directly-connected management station running SNMP or Syslog. This feature is not supported for BVIs for bridge groups or with multicast routing. We modified the following commands: **ip address, http, logging host, snmp-server, ssh**



CHAPTER 19

Advanced Interface Configuration

This chapter describes how to configure MAC addresses for interfaces, how to set the maximum transmission unit (MTU), and set the TCP maximum segment size (TCP MSS), and how to allow same security level communication. Setting the correct MTU and maximum TCP segment size is essential for the best network performance.

- About Advanced Interface Configuration, on page 707
- Manually Configure the MAC Address, on page 711
- Automatically Assign MAC Addresses, on page 712
- Configure the MTU and TCP MSS, on page 713
- Allow Same Security Level Communication, on page 715
- History for Advanced Interface Configuration, on page 715

About Advanced Interface Configuration

This section describes advanced interface settings.

About MAC Addresses

You can manually assign MAC addresses to override the default. For multiple context mode, you can automatically generate unique MAC addresses (for all interfaces assigned to a context) and single context mode (for subinterfaces)..

Note You might want to assign unique MAC addresses to subinterfaces defined on the ASA, because they use the same burned-in MAC address of the parent interface. For example, your service provider might perform access control based on the MAC address. Also, because IPv6 link-local addresses are generated based on the MAC address, assigning unique MAC addresses to subinterfaces allows for unique IPv6 link-local addresses, which can avoid traffic disruption in certain instances on the ASA device.

Default MAC Addresses

Default MAC address assignments depend on the type of interface.

- Physical interfaces—The physical interface uses the burned-in MAC address.

- VLAN interfaces (Firepower 1010)—Routed firewall mode: All VLAN interfaces share a MAC address. Ensure that any connected switches can support this scenario. If the connected switches require unique MAC addresses, you can manually assign MAC addresses. See Manually Configure the MAC Address, on page 711.

  Transparent firewall mode: Each VLAN interface has a unique MAC address. You can override the generated MAC addresses if desired by manually assigning MAC addresses. See Manually Configure the MAC Address, on page 711.

- EtherChannels—For an EtherChannel, all interfaces that are part of the channel group share the same MAC address. This feature makes the EtherChannel transparent to network applications and users, because they only see the one logical connection; they have no knowledge of the individual links. The port-channel interface uses a unique MAC address from a pool; interface membership does not affect the MAC address.
- EtherChannels (ASA Models)—The port-channel interface uses the lowest-numbered channel group interface MAC address as the port-channel MAC address. Alternatively you can configure a MAC address for the port-channel interface. We recommend configuring a unique MAC address in case the group channel interface membership changes. If you remove the interface that was providing the port-channel MAC address, then the port-channel MAC address changes to the next lowest numbered interface, thus causing traffic disruption.
- Subinterfaces—All subinterfaces of a physical interface use the same burned-in MAC address. You might want to assign unique MAC addresses to subinterfaces. For example, your service provider might perform access control based on the MAC address. Also, because IPv6 link-local addresses are generated based on the MAC address, assigning unique MAC addresses to subinterfaces allows for unique IPv6 link-local addresses, which can avoid traffic disruption in certain instances on the ASA.

Automatic MAC Addresses

In multiple context mode, auto-generation assigns unique MAC addresses to all interfaces assigned to a context.

If you manually assign a MAC address and also enable auto-generation, then the manually assigned MAC address is used. If you later remove the manual MAC address, the auto-generated address is used, if enabled.

In the rare circumstance that the generated MAC address conflicts with another private MAC address in your network, you can manually set the MAC address for the interface.

Because auto-generated addresses (when using a prefix) start with A2, you cannot start manual MAC addresses with A2 if you also want to use auto-generation.

The ASA generates the MAC address using the following format:

A2*xx.yyzz.zzzz*

Where *xx.yy* is a user-defined prefix or an autogenerated prefix based on the last two bytes of the interface MAC address, and *zz.zzzz* is an internal counter generated by the ASA. For the standby MAC address, the address is identical except that the internal counter is increased by 1.

For an example of how the prefix is used, if you set a prefix of 77, then the ASA converts 77 into the hexadecimal value 004D (*yyxx*). When used in the MAC address, the prefix is reversed (*xxyy*) to match the ASA native form:

A2**4D.00***zz.zzzz*

For a prefix of 1009 (03F1), the MAC address is:

A2**F1.03**zz.zzzz

Note The MAC address format without a prefix is a legacy version. See the **mac-address auto** command in the command reference for more information about the legacy format.

About the MTU

The MTU specifies the maximum frame *payload* size that the ASA can transmit on a given Ethernet interface. The MTU value is the frame size *without* Ethernet headers, VLAN tagging, or other overhead. For example, when you set the MTU to 1500, the expected frame size is 1518 bytes including the headers, or 1522 when using VLAN. Do not set the MTU value higher to accommodate these headers.

For VXLAN or Geneve, the entire Ethernet datagram is being encapsulated, so the new IP packet is larger and requires a larger MTU: you should set the ASA VTEP source interface MTU to be the network MTU + 54 bytes (for VXLAN) or + 306 bytes (Geneve).

Path MTU Discovery

The ASA supports Path MTU Discovery (as defined in RFC 1191), which lets all devices in a network path between two hosts coordinate the MTU so they can standardize on the lowest MTU in the path.

Default MTU

The default MTU on the ASA is 1500 bytes. This value does not include the 18-22 bytes for the Ethernet header, VLAN tagging, or other overhead.

When you enable VXLAN on the VTEP source interface, if the MTU is less than 1554 bytes, then the ASA automatically raises the MTU to 1554 bytes. In this case, the entire Ethernet datagram is being encapsulated, so the new packet is larger and requires a larger MTU. In general, you should set the ASA source interface MTU to be the network MTU + 54 bytes.

MTU and Fragmentation

For IPv4, if an outgoing IP packet is larger than the specified MTU, it is fragmented into 2 or more frames. Fragments are reassembled at the destination (and sometimes at intermediate hops), and fragmentation can cause performance degradation. For IPv6, packets are typically not allowed to be fragmented at all. Therefore, your IP packets should fit within the MTU size to avoid fragmentation.

For TCP packets, the endpoints typically use their MTU to determine the TCP maximum segment size (MTU - 40, for example). If additional TCP headers are added along the way, for example for site-to-site VPN tunnels, then the TCP MSS might need to be adjusted down by the tunneling entity. See About the TCP MSS, on page 710.

For UDP or ICMP, the application should take the MTU into account to avoid fragmentation.

Note The ASA can receive frames larger than the configured MTU as long as there is room in memory.

MTU and Jumbo Frames

A larger MTU lets you send larger packets. Larger packets might be more efficient for your network. See the following guidelines:

- Matching MTUs on the traffic path—We recommend that you set the MTU on all ASA interfaces and other device interfaces along the traffic path to be the same. Matching MTUs prevents intermediate devices from fragmenting the packets.
- Accommodating jumbo frames—You can set the MTU 9000 bytes or higher when you enable jumbo frames. The maximum depends on the model.

About the TCP MSS

The TCP maximum segment size (MSS) is the size of the TCP payload *before* any TCP and IP headers are added. UDP packets are not affected. The client and the server exchange TCP MSS values during the three-way handshake when establishing the connection.

You can set the TCP MSS on the ASA for through traffic; by default, the maximum TCP MSS is set to 1380 bytes. This setting is useful when the ASA needs to add to the size of the packet for IPsec VPN encapsulation. However, for non-IPsec endpoints, you should disable the maximum TCP MSS on the ASA.

If you set a maximum TCP MSS, if either endpoint of a connection requests a TCP MSS that is larger than the value set on the ASA, then the ASA overwrites the TCP MSS in the request packet with the ASA maximum. If the host or server does not request a TCP MSS, then the ASA assumes the RFC 793-default value of 536 bytes (IPv4) or 1220 bytes (IPv6), but does not modify the packet. For example, you leave the default MTU as 1500 bytes. A host requests an MSS of 1500 minus the TCP and IP header length, which sets the MSS to 1460. If the ASA maximum TCP MSS is 1380 (the default), then the ASA changes the MSS value in the TCP request packet to 1380. The server then sends packets with 1380-byte payloads. The ASA can then add up to 120 bytes of headers to the packet and still fit in the MTU size of 1500.

You can also configure the minimum TCP MSS; if a host or server requests a very small TCP MSS, the ASA can adjust the value up. By default, the minimum TCP MSS is not enabled.

For to-the-box traffic, including for SSL VPN connections, this setting does not apply. The ASA uses the MTU to derive the TCP MSS: MTU - 40 (IPv4) or MTU - 60 (IPv6).

Default TCP MSS

By default, the maximum TCP MSS on the ASA is 1380 bytes. This default accommodates IPv4 IPsec VPN connections where the headers can equal up to 120 bytes; this value fits within the default MTU of 1500 bytes.

Suggested Maximum TCP MSS Setting

The default TCP MSS assumes the ASA acts as an IPv4 IPsec VPN endpoint and has an MTU of 1500. When the ASA acts as an IPv4 IPsec VPN endpoint, it needs to accommodate up to 120 bytes for TCP and IP headers.

If you change the MTU value, use IPv6, or do not use the ASA as an IPsec VPN endpoint, then you should change the TCP MSS setting.

See the following guidelines:

- Normal traffic—Disable the TCP MSS limit and accept the value established between connection endpoints. Because connection endpoints typically derive the TCP MSS from the MTU, non-IPsec packets usually fit this TCP MSS.

- IPv4 IPsec endpoint traffic—Set the maximum TCP MSS to the MTU - 120. For example, if you use jumbo frames and set the MTU to 9000, then you need to set the TCP MSS to 8880 to take advantage of the new MTU.
- IPv6 IPsec endpoint traffic—Set the maximum TCP MSS to the MTU - 140.

Inter-Interface Communication

Allowing interfaces on the same security level to communicate with each other provides the following benefits:

- You can configure more than 101 communicating interfaces.

  If you use different levels for each interface and do not assign any interfaces to the same security level, you can configure only one interface per level (0 to 100).

- You want traffic to flow freely between all same security interfaces without ACLs.

If you enable same security interface communication, you can still configure interfaces at different security levels as usual.

Intra-Interface Communication (Routed Firewall Mode)

Intra-interface communication might be useful for VPN traffic that enters an interface, but is then routed out the same interface. The VPN traffic might be unencrypted in this case, or it might be reencrypted for another VPN connection. For example, if you have a hub and spoke VPN network, where the ASA is the hub, and remote VPN networks are spokes, for one spoke to communicate with another spoke, traffic must go into the ASA and then out again to the other spoke.



Note All traffic allowed by this feature is still subject to firewall rules. Be careful not to create an asymmetric routing situation that can cause return traffic not to traverse the ASA.

Manually Configure the MAC Address

If you need to manually assign the MAC address, you can do so using this procedure.

You might want to assign unique MAC addresses to subinterfaces defined on the ASA, because they use the same burned-in MAC address of the parent interface. For example, your service provider might perform access control based on the MAC address. Also, because IPv6 link-local addresses are generated based on the MAC address, assigning unique MAC addresses to subinterfaces allows for unique IPv6 link-local addresses, which can avoid traffic disruption in certain instances on the ASA.

Before you begin

In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the **changeto context** *name* command.

Procedure

Step 1 Enter interface configuration mode:

interface *id*

Example:

```
ciscoasa(config)# interface gigabithethernet 0/0
```

Step 2 Assign a private MAC address to this interface:

mac-address *mac_address* [**standby** *mac_address*]

Example:

```
ciscoasa(config-if)# mac-address 000C.F142.4CDE
```

The *mac_address* is in H.H.H format, where H is a 16-bit hexadecimal digit. For example, the MAC address 00-0C-F1-42-4C-DE is entered as 000C.F142.4CDE. The MAC address must not have the multicast bit set, that is, the second hexadecimal digit from the left cannot be an odd number.

The first two bytes of a manual MAC address cannot be A2 if you also want to use auto-generated MAC addresses.

For use with failover, set the **standby** MAC address. If the active unit fails over and the standby unit becomes active, the new active unit starts using the active MAC addresses to minimize network disruption, while the old active unit uses the standby address.

Automatically Assign MAC Addresses

This section describes how to configure auto-generation of MAC addresses. For multiple context mode, this feature assigns unique MAC addresses to all interface types that are assigned to a context. For single mode, this feature assigns unique MAC addresses to VLAN subinterfaces.

Before you begin

- When you configure a **nameif** command for the interface, the new MAC address is generated immediately. If you enable this feature after you configure interfaces, then MAC addresses are generated for all interfaces immediately after you enable it. If you disable this feature, the MAC address for each interface reverts to the default MAC address. For example, subinterfaces of GigabitEthernet 0/1 revert to using the MAC address of GigabitEthernet 0/1.
- In the rare circumstance that the generated MAC address conflicts with another private MAC address in your network, you can manually set the MAC address for the interface.
- For multiple context mode, complete this procedure in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.

Procedure

Automatically assign private MAC addresses to each interface:

mac-address auto [**prefix** *prefix*]

If you do not enter a prefix, then the ASA autogenerates the prefix based on the last two bytes of the interface MAC address.

If you manually enter a prefix, then the *prefix* is a decimal value between 0 and 65535. This prefix is converted to a four-digit hexadecimal number, and used as part of the MAC address.

Example:

```
ciscoasa(config)# mac-address auto prefix 19
```

Configure the MTU and TCP MSS

Before you begin

- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the **changeto context** *name* command.
- To increase the MTU above 1500, enable jumbo frames according to Enable Jumbo Frame Support (ASAv, ISA 3000), on page 600.

Procedure

Step 1 Set the MTU. The minimum and maximum depend on your platform.

mtu *interface_name bytes*

Example:

```
ciscoasa(config-if)# mtu inside ?

configure mode commands/options:
  <64-9198>  MTU bytes
ciscoasa(config)# mtu inside 9000
```

The default is 1500 bytes.

Note
When you set the MTU for a port-channel interface, the ASA applies the setting to all member interfaces.

For some models that support jumbo frames, if you enter a value for any interface that is greater than 1500, then you need to enable jumbo frame support. See Enable Jumbo Frame Support (ASAv, ISA 3000), on page 600.

Step 2 Configure the maximum TCP segment size in bytes, between 48 and any maximum number:

sysopt connection tcpmss [**minimum**] *bytes*

Example:

```
ciscoasa(config)# sysopt connection tcpmss 8500
ciscoasa(config)# sysopt connection tcpmss minimum 1290
```

The default value is 1380 bytes. You can disable this feature by setting bytes to **0**.

For the **minimum** keyword, sets the maximum segment size to be no less than *bytes*, between 48 and 65535. The minimum feature is disabled by default (set to 0).

Step 3 Configure the maximum number of unprocessed TCP segments.

sysopt connection tcp-max-unprocessed-seg *unprocessed segments*

Example:

```
ciscoasa(config)# sysopt connection tcp-max-unprocessed-seg 7
```

The default value is 6. The range is from 6 to 24.

Examples

The following example enables jumbo frames, increases the MTU on all interfaces, and disables the TCP MSS for non-VPN traffic (by setting the TCP MSS to 0, which means there is no limit):

```
mtu inside 9198
mtu outside 9198
sysopt connection tcpmss 0
```

The following example enables jumbo frames, increases the MTU on all interfaces, and changes the TCP MSS for VPN traffic to 9078 (the MTU minus 120):

```
mtu inside 9198
mtu outside 9198
sysopt connection tcpmss 9078
```

Allow Same Security Level Communication

By default, interfaces on the same security level cannot communicate with each other, and packets cannot enter and exit the same interface. This section describes how to enable inter-interface communication when interfaces are on the same security level, and how to enable intra-interface communication.

Procedure

Step 1 Enable interfaces on the same security level so that they can communicate with each other:

same-security-traffic permit inter-interface

Step 2 Enable communication between hosts connected to the same interface:

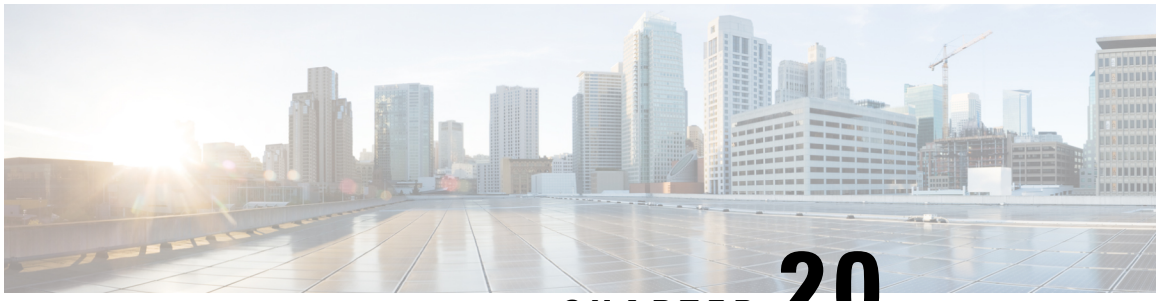
same-security-traffic permit intra-interface

History for Advanced Interface Configuration

Table 28: History for Advanced Interface Configuration

Feature Name	Releases	Feature Information
Maximum MTU is now 9198 bytes	9.1(6), 9.2(1)	The maximum MTU that the ASA can use is 9198 bytes (check for your model's exact limit at the CLI help). This value does not include the Layer 2 header. Formerly, the ASA let you specify the maximum MTU as 65535 bytes, which was inaccurate and could cause problems. If your MTU was set to a value higher than 9198, then the MTU is automatically lowered when you upgrade. In some cases, this MTU change can cause an MTU mismatch; be sure to set any connecting equipment to use the new MTU value. We modified the following command: **mtu**
Increased MTU size for the ASA on the Firepower 4100/9300 chassis	9.6(2)	You can set the maximum MTU to 9184 bytes on the Firepower 4100 and 9300; formerly, the maximum was 9000 bytes. This MTU is supported with FXOS 2.0.1.68 and later. We modified the following command: **mtu**
Unique MAC address generation for single context mode	9.8(3), 9.8(4), 9.9(2)	You can now enable unique MAC address generation for VLAN subinterfaces in single context mode. Normally, subinterfaces share the same MAC address with the main interface. Because IPv6 link-local addresses are generated based on the MAC address, this feature allows for unique IPv6 link-local addresses. New or modified command: **mac-address auto**

Feature Name	Releases	Feature Information
Secure Firewall 3100 auto-negotiation can be enabled or disabled for 1Gigabit and higher interfaces.	9.17(1)	Secure Firewall 3100 auto-negotiation can be enabled or disabled for 1Gigabit and higher interfaces. For other model SFP ports, the **no speed nonegotiate** option sets the speed to 1000 Mbps; the new command means you can set auto-negotiation and speed independently. New/Modified commands: **negotiate-auto**



CHAPTER 20

Traffic Zones

You can assign multiple interfaces to a *traffic zone*, which lets traffic from an existing flow exit or enter the ASA on any interface within the zone. This capability allows Equal-Cost Multi-Path (ECMP) routing on the ASA as well as external load balancing of traffic to the ASA across multiple interfaces.

- About Traffic Zones, on page 717
- Prerequisites for Traffic Zones, on page 723
- Guidelines for Traffic Zones, on page 724
- Configure a Traffic Zone, on page 726
- Monitoring Traffic Zones, on page 727
- Example for Traffic Zones, on page 729
- History for Traffic Zones, on page 732

About Traffic Zones

This section describes how you should use traffic zones in your network.

Non-Zoned Behavior

The Adaptive Security Algorithm takes into consideration the state of a packet when deciding to permit or deny the traffic. One of the enforced parameters for the flow is that traffic enters and exits the same interface. Any traffic for an existing flow that enters a different interface is dropped by the ASA.

Traffic zones let you group multiple interfaces together so that traffic entering or exiting *any* interface in the zone fulfills the Adaptive Security Algorithm security checks.

Related Topics

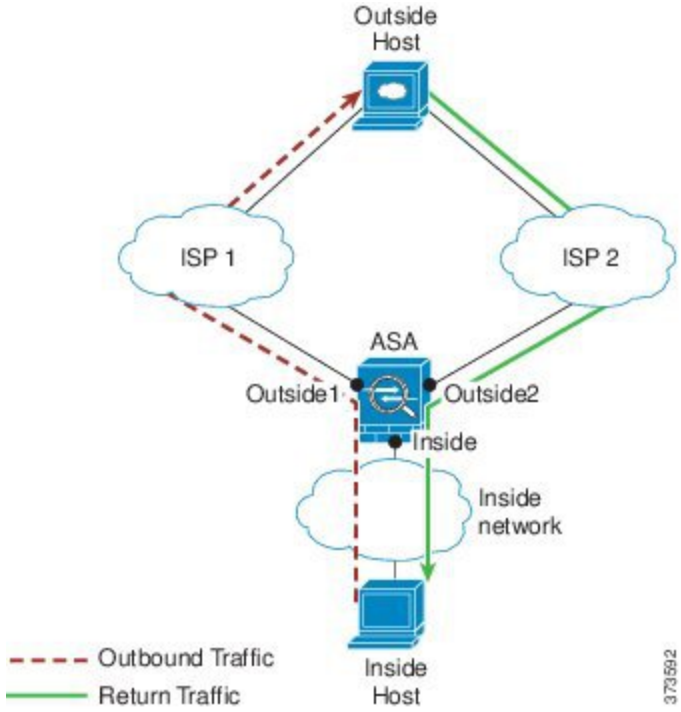
Stateful Inspection Overview, on page 9

Why Use Zones?

You can use zones to accommodate several routing scenarios.

Asymmetric Routing

In the following scenario, a connection was established between an inside host and an outside host through ISP 1 on the Outside1 interface. Due to asymmetric routing on the destination network, return traffic arrived from ISP 2 on the Outside2 interface.

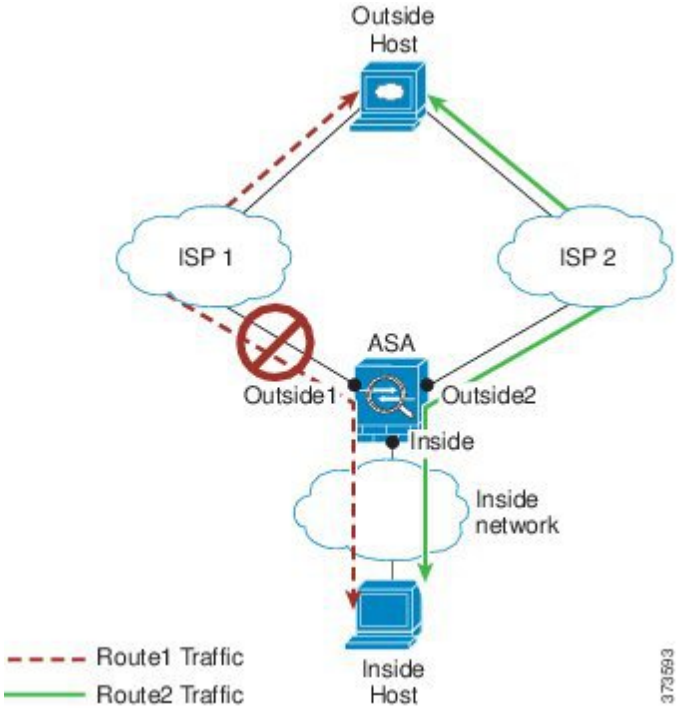


Non-Zoned Problem: The ASA maintains the connection tables on a per-interface basis. When the returning traffic arrives at Outside2, it will not match the connection table and will be dropped. For an ASA cluster, asymmetric routing when the cluster has multiple adjacencies to the same router can lead to unacceptible traffic loss.

Zoned Solution: The ASA maintains connection tables on a per-zone basis. If you group Outside1 and Outside2 into a zone, then when the returning traffic arrives at Outside2, it will match the per-zone connection table, and the connection will be allowed.

Lost Route

In the following scenario, a connection was established between an inside host and an outside host through ISP 1 on the Outside1 interface. Due to a lost or moved route between Outside1 and ISP 1, traffic needs to take a different route through ISP 2.

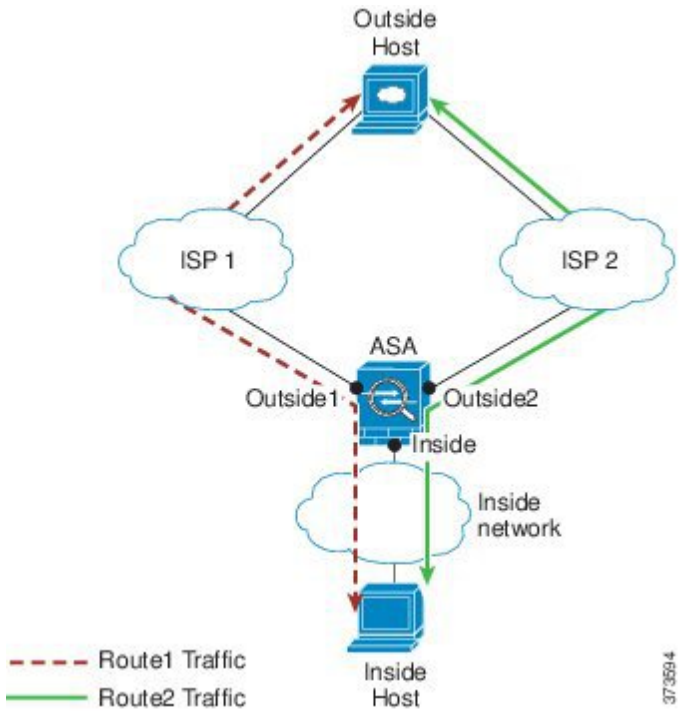


Non-Zoned Problem: The connection between the inside and outside host will be deleted; a new connection must be established using a new next-best route. For UDP, the new route will be used after a single packet drop, but for TCP, a new connection has to be reestablished.

Zoned Solution: The ASA detects the lost route and switches the flow to the new path through ISP 2. Traffic will be seamlessly forwarded without any packet drops.

Load Balancing

In the following scenario, a connection was established between an inside host and an outside host through ISP 1 on the Outside1 interface. A second connection was established through an equal cost route through ISP 2 on Outside2.



Non-Zoned Problem: Load-balancing across interfaces is not possible; you can only load-balance with equal cost routes on one interface.

Zoned Solution: The ASA load-balances connections across up to eight equal cost routes on all the interfaces in the zone.

Per-Zone Connection and Routing Tables

The ASA maintains a per-zone connection table so that traffic can arrive on any of the zone interfaces. The ASA also maintains a per-zone routing table for ECMP support.

ECMP Routing

The ASA supports Equal-Cost Multi-Path (ECMP) routing.

Non-Zoned ECMP Support

Without zones, you can have up to 8 equal cost static or dynamic routes per interface. For example, you can configure three default routes on the outside interface that specify different gateways:

```
route outside 0 0 10.1.1.2
route outside 0 0 10.1.1.3
route outside 0 0 10.1.1.4
```

In this case, traffic is load-balanced on the outside interface between 10.1.1.2, 10.1.1.3, and 10.1.1.4. Traffic is distributed among the specified gateways based on an algorithm that hashes the source and destination IP addresses, incoming interface, protocol, source and destination ports.

ECMP is not supported across multiple interfaces, so you cannot define a route to the same destination on a different interface. The following route is disallowed when configured with any of the routes above:

```
route outside2 0 0 10.2.1.1
```

Zoned ECMP Support

With zones, you can have up to 8 equal cost static or dynamic routes across interface within a zone.

When ECMP is determining how to distribute traffic, it sees each virtual access interface as a single participant or path option, even if that virtual access interface is associated with an interface that has several IPv6 addresses.

For example, you can configure three default routes across three interfaces in the zone:

```
route outside1 0 0 10.1.1.2
route outside2 0 0 10.2.1.2
route outside3 0 0 10.3.1.2
```

Similarly, your dynamic routing protocol can automatically configure equal cost routes. The ASA load-balances traffic across the interfaces with a more robust load balancing mechanism.

When a route is lost, the ASA seamlessly moves the flow to a different route.

How Connections Are Load-Balanced

The ASA load balances connections across equal cost routes using a hash made from the packet 6-tuple (source and destination IP address, source and destination port, protocol, and ingress interface). Unless the route is lost, a connection will stay on the chosen interface for its duration.

Packets within a connection are not load-balanced across routes; a connection uses a single route unless that route is lost.

The ASA does not consider the interface bandwidth or other parameters when load balancing. You should make sure all interfaces within the same zone have the same characteristics such as MTU, bandwidth, and so on.

The load-balancing algorithm is not user configurable.

Falling Back to a Route in Another Zone

When a route is lost on an interface, if there are no other routes available within the zone, then the ASA will use a route from a different interface/zone. If this backup route is used, then you may experience packet drops as with non-zoned routing support.

Interface-Based Security Policy

Zones allow traffic to and from any interface in the zone, but the security policy itself (access rules, NAT, and so on) is still applied per interface, not per zone. If you configure the same security policy for all interfaces within the zone, then you can successfully implement ECMP and load balancing for that traffic. For more information about required parallel interface configuration, see Prerequisites for Traffic Zones, on page 723.

Supported Services for Traffic Zones

The following services are supported with zones:

- Access Rules
- NAT
- Service Rules, except for QoS traffic policing.
- Routing

You can also configure to- and from-the-box services listed in To- and From-the-Box Traffic, on page 723, although full zoned support is not available.

Do not configure other services (such as VPN or Botnet Traffic Filter) for interfaces in a traffic zone; they may not function or scale as expected.



Note For detailed information about how to configure the security policy, see Prerequisites for Traffic Zones, on page 723.

Security Levels

The first interface that you add to a zone determines the security level of the zone. All additional interfaces must have the same security level. To change the security level for interfaces in a zone, you must remove all but one interface, and then change the security levels, and re-add the interfaces.

Primary and Current Interface for the Flow

Each connection flow is built based on the initial ingress and egress interfaces. These interfaces are the *primary* interfaces.

If a new egress interface is used because of route changes or asymmetric routing, then the new interfaces are the *current* interfaces.

Joining or Leaving a Zone

When you assign an interface to a zone, any connections on that interface are deleted. The connections must be reestablished.

If you remove an interface from a zone, any connections that have the interface as the primary interface are deleted. The connections must be reestablished. If the interface is the current interface, the ASA moves the connections back to the primary interface. The zone route table is also refreshed.

Intra-Zone Traffic

To allow traffic to *enter* one interface and *exit* another in the same zone, enable the **same-security permit intra-interface** command, which allows traffic to enter and exit the same interface, as well as the **same-security**

permit inter-interface command, which allows traffic between same-security interfaces. Otherwise, a flow cannot be routed between two interfaces in the same zone.

To- and From-the-Box Traffic

- You cannot add management-only or management-access interfaces to a zone.
- For management traffic on regular interfaces in a zone, only asymmetric routing on existing flows is supported; there is no ECMP support.
- You can configure a management service on only one zone interface, but to take advantage of asymmetric routing support, you need to configure it on all interfaces. Even when the configurations are parallel on all interfaces, ECMP is not supported.
- The ASA supports the following to- and from-the-box services in a zone:
  - Telnet
  - SSH
  - HTTPS
  - SNMP
  - Syslog

Overlapping IP Addresses Within a Zone

For non-zoned interfaces, the ASA supports overlapping IP address networks on interfaces so long as you configure NAT properly. However, overlapping networks are not supported on interfaces in the same zone.

Prerequisites for Traffic Zones

- Configure all interface parameters including the name, IP address, and security level. Note that the security level must match for all interfaces in the zone. You should plan to group together like interfaces in terms of bandwidth and other Layer 2 properties.
- Configure the following services to match on all zone interfaces:
  - Access Rules—Apply the same access rule to all zone member interfaces, or use a global access rule.

    For example:

    ```
    access-list ZONE1 extended permit tcp any host WEBSERVER1 eq 80
    access-group ZONE1 in interface outside1
    access-group ZONE1 in interface outside2
    access-group ZONE1 in interface outside3
    ```

  - NAT—Configure the same NAT policy on all member interfaces of the zone or use a global NAT rule (in other words, use “any” to represent the zone interfaces in the NAT rule).

    Interface PAT is not supported.

For example:

```
object network WEBSERVER1
  host 10.9.9.9 255.255.255.255
  nat (inside,any) static 209.165.201.9
```

Note When you use interface-specific NAT and PAT pools, the ASA cannot switch connections over in case of the original interface failure.

If you use interface-specific PAT pools, multiple connections from the same host might load-balance to different interfaces and use different mapped IP addresses. Internet services that use multiple concurrent connections may not work correctly in this case.

- Service Rules—Use the global service policy, or assign the same policy to each interface in a zone.

  QoS traffic policing is not supported.

  For example:

```
service-policy outside_policy interface outside1
service-policy outside_policy interface outside2
service-policy outside_policy interface outside3
```

Note For VoIP inspections, zone load balancing can cause increased out-of-order packets. This situation can occur because later packets might reach the ASA before earlier packets that take a different path. Symptoms of out-of-order packets include:

- Higher memory utilization at intermediate nodes (firewall and IDS) and the receiving end nodes if queuing is used.
- Poor video or voice quality.

To mitigate these effects, we recommend that you use IP addresses only for load distribution for VoIP traffic.

- Configure routing with ECMP zone capabilities in mind.

Guidelines for Traffic Zones

Firewall Mode

Supported in routed firewall mode only. Does not support transparent firewall mode or bridge group interfaces in routed mode.

Failover

- You cannot add the failover or state link to a zone.
- In Active/Active failover mode, you can assign an interface in each context to an asymmetrical routing (ASR) group. This service allows traffic returning on a similar interface on the peer unit to be restored to the original unit. You cannot configure both ASR groups and traffic zones within a context. If you configure a zone in a context, none of the context interfaces can be part of an ASR group. See Configure Support for Asymmetrically Routed Packets (Active/Active Mode), on page 298 for more information about ASR groups.
- Only the primary interfaces for each connection are replicated to the standby unit; current interfaces are not replicated. If the standby unit becomes active, it will assign a new current interface if necessary.

Clustering

- You cannot add the cluster control link to a zone.

Model Guidelines

You cannot add Firepower 1010 switch ports and VLAN interfaces to a zone.

Additional Guidelines

- You can create a maximum of 256 zones.
- You can add the following types of interfaces to a zone:
  - Physical
  - VLAN
  - EtherChannel
- You cannot add the following types of interfaces:
  - Management-only
  - Management-access
  - Failover or state link
  - Cluster control link
  - Member interfaces in an EtherChannel
  - VNI; also, if a regular data interface is marked as nve-only, it cannot be a member of a zone.
  - BVI, or bridge group member interfaces.
- An interface can be a member of only one zone.
- You can include up to 8 interfaces per zone.
- For ECMP, you can add up to 8 equal cost routes per zone, across all zone interfaces. You can also configure multiple routes on a single interface as part of the 8 route limit.
- When you add an interface to a zone, all static routes for those interfaces are removed.

- You cannot enable DHCP Relay on an interface in a traffic zone.
- The ASA does not support fragmented packet reassembly for fragments that are load-balanced to separate interfaces; those fragments will be dropped.
- PIM/IGMP Multicast routing is not supported on interfaces in a zone.

Configure a Traffic Zone

Configure a named zone, and assign interfaces to the zone.

Procedure

Step 1 Add the zone:

zone *name*

Example:

```
zone outside
```

The zone name can be up to 48 characters in length.

Step 2 Add an interface to the zone:

interface *id* **zone-member** *zone_name*

Example:

```
interface gigabitethernet0/0
  zone-member outside
```

Step 3 Add more interfaces to the zone; ensure they have the same security level as the first interface you added.

Example:

```
interface gigabitethernet0/1
  zone-member outside
interface gigabitethernet0/2
  zone-member outside
interface gigabitethernet0/3
  zone-member outside
```

Examples

The following example configures an outside zone with 4 member interfaces:

```
zone outside
```

```
interface gigabitethernet0/0
  zone-member outside
interface gigabitethernet0/1
  zone-member outside
interface gigabitethernet0/2
  zone-member outside
interface gigabitethernet0/3
  zone-member outside
```

Monitoring Traffic Zones

This section describes how to monitor traffic zones.

Zone Information

- **show zone** [*name*]

  Shows zone ID, context, security level, and members.

  See the following output for the **show zone** command:

```
ciscoasa# show zone outside-zone

Zone: zone-outside id: 2
Security-level: 0
Context: test-ctx
 Zone Member(s) : 2
  outside1       GigabitEthernet0/0
  outside2       GigabitEthernet0/1
```

- **show nameif zone**

  Shows the interface names and zone names.

  See the following output for the **show nameif zone** command:

```
ciscoasa# show nameif zone
Interface                Name                   zone-name      Security
GigabitEthernet0/0       inside-1               inside-zone     100
GigabitEthernet0/1.21    inside                 inside-zone     100
GigabitEthernet0/1.31    4                                        0
GigabitEthernet0/2       outside                outside-zone   0
Management0/0            lan                                      0
```

Zone Connections

- **show conn** [**long** | **detail**] [**zone** *zone_name* [**zone** *zone_name*] [...]]

  The **show conn zone** command displays connections for a zone. The **long** and **detail** keywords show the primary interface on which the connection was built and the one in the brackets is the current interface used to forward the traffic or the interface the last packet came from. Thus, the current interface in case

of a connection coming from multiple interfaces can show different interfaces at different times depending on when the show conn command was issued.

See the following output for the **show conn long zone** command:

```
ciscoasa# show conn long zone zone-inside zone zone-outside

TCP outside-zone:outside1(outside2): 10.122.122.1:1080
inside-zone:inside1(inside2): 10.121.121.1:34254, idle 0:00:02, bytes 10, flags UO
```

- **show asp table zone**

  Shows the accelerated security path tables for debugging purposes.

- **show local-host** [**zone** *zone_name* [**zone** *zone_name*] [...]]

  Shows the network states of local hosts within a zone.

  See the following output for the **show local-host zone** command. The primary interface is listed first, and the current interface is in parentheses.

```
ciscoasa# show local-host zone outside-zone

Zone:outside-zone: 4 active, 5 maximum active, 0 denied
local host: <10.122.122.1>,
    TCP flow count/limit = 3/unlimited
    TCP embryonic count to host = 0
    TCP intercept watermark = unlimited
    UDP flow count/limit = 0/unlimited

  Conn:
 TCP outside-zone:outside1(outside2): 10.122.122.1:1080
inside-zone:inside1(inside2): 10.121.121.1:34254, idle 0:00:02, bytes 10, flags UO
```

Zone Routing

- **show route zone**

  Shows the routes for zone interfaces.

  See the following output for the **show route zone** command:

```
ciscoasa# show route zone

Codes: C - connected, S - static, I - IGRP, 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, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

Gateway of last resort is not set

S    192.168.105.1 255.255.255.255 [1/0] via 172.16.1.1, outside-zone:outside1
C    192.168.212.0 255.255.255.0 is directly connected, lan-zone:inside,
C    172.16.1.0 255.255.255.0 is directly connected, wan-zone:outside2
S    10.5.5.0 255.255.255.0 [1/0] via 172.16.1.1, wan-zone:outside2
```

```
O    10.2.2.1 255.255.255.255 [110/11] via 192.168.212.3, 2:09:24, lan-zone:inside
O    10.1.1.1 255.255.255.255 [110/11] via 192.168.212.2, 2:09:24, lan-zone:inside
```

- **show asp table routing**

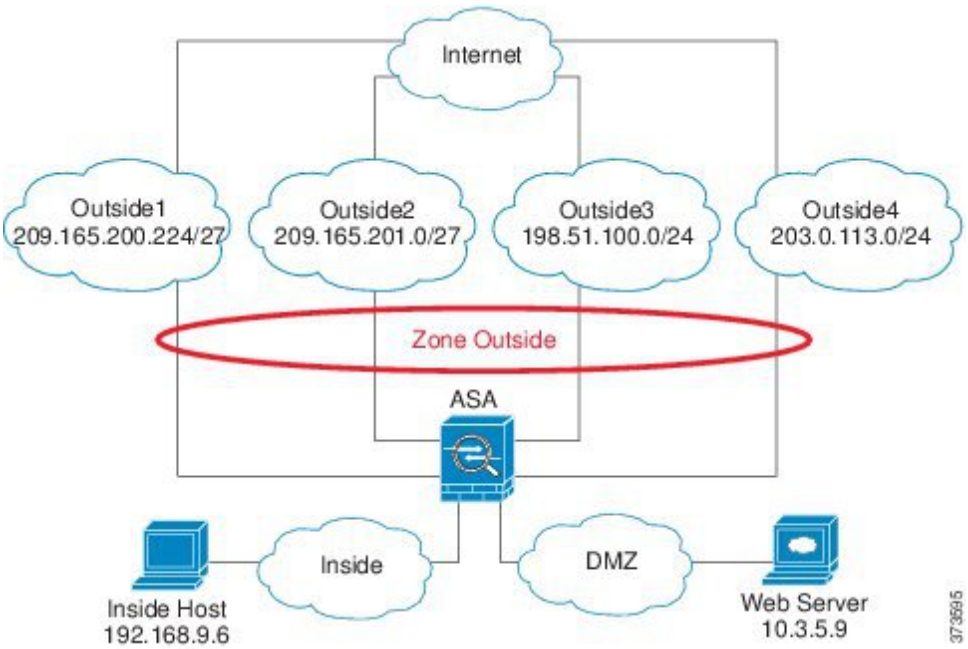
  Shows the accelerated security path tables for debugging purposes, and shows the zone associated with each route.

  See the following output for the **show asp table routing** command:

```
ciscoasa# show asp table routing
route table timestamp: 60
in   255.255.255.255 255.255.255.255 identity
in   10.1.0.1       255.255.255.255 identity
in   10.2.0.1       255.255.255.255 identity
in   10.6.6.4         255.255.255.255 identity
in   10.4.4.4         255.255.255.255 via 10.4.0.10 (unresolved, timestamp: 49)
in   172.0.0.67       255.255.255.255 identity
in   172.0.0.0       255.255.255.0   wan-zone:outside2
in   10.85.43.0      255.255.255.0   via 10.4.0.3 (unresolved, timestamp: 50)
in   10.85.45.0      255.255.255.0   via 10.4.0.20 (unresolved, timestamp: 51)
in   192.168.0.0        255.255.255.0   mgmt
in   192.168.1.0       255.255.0.0     lan-zone:inside
out  255.255.255.255 255.255.255.255 mgmt
out  172.0.0.67       255.255.255.255 mgmt
out  172.0.0.0        255.255.255.0   mgmt
out  10.4.0.0       240.0.0.0       mgmt
out  255.255.255.255 255.255.255.255 lan-zone:inside
out  10.1.0.1       255.255.255.255 lan-zone:inside
out  10.2.0.0       255.255.0.0     lan-zone:inside
out  10.4.0.0       240.0.0.0       lan-zone:inside
```

Example for Traffic Zones

The following example assigns 4 VLAN interfaces to the outside zone, and configures 4 equal cost default routes. PAT is configured for the inside interface, and a web server is available on a DMZ interface using static NAT.




```
interface gigabitethernet0/0
  no shutdown
  description outside switch 1
interface gigabitethernet0/1
  no shutdown
  description outside switch 2

interface gigabitethernet0/2
  no shutdown
  description inside switch

zone outside

interface gigabitethernet0/0.101
  vlan 101
  nameif outside1
  security-level 0
  ip address 209.165.200.225 255.255.255.224
  zone-member outside
  no shutdown

interface gigabitethernet0/0.102
  vlan 102
  nameif outside2
  security-level 0
  ip address 209.165.201.1 255.255.255.224
  zone-member outside
  no shutdown

interface gigabitethernet0/1.201
  vlan 201
  nameif outside3
  security-level 0
  ip address 198.51.100.1 255.255.255.0
  zone-member outside
  no shutdown
```

```
interface gigabitethernet0/1.202
  vlan 202
  nameif outside4
  security-level 0
  ip address 203.0.113.1 255.255.255.0
  zone-member outside
  no shutdown

interface gigabitethernet0/2.301
  vlan 301
  nameif inside
  security-level 100
  ip address 192.168.9.1 255.255.255.0
  no shutdown

interface gigabitethernet0/2.302
  vlan 302
  nameif dmz
  security-level 50
  ip address 10.3.5.1 255.255.255.0
  no shutdown

# Static NAT for DMZ web server on any destination interface
object network WEBSERVER
  host 10.3.5.9 255.255.255.255
  nat (dmz,any) static 209.165.202.129 dns

# Dynamic PAT for inside network on any destination interface
object network INSIDE
  subnet 192.168.9.0 255.255.255.0
  nat (inside,any) dynamic 209.165.202.130

# Global access rule for DMZ web server
access-list WEB-SERVER extended permit tcp any host WEBSERVER eq 80
access-group WEB-SERVER global

# 4 equal cost default routes for outside interfaces
route outside1 0 0 209.165.200.230
route outside2 0 0 209.165.201.10
route outside3 0 0 198.51.100.99
route outside4 0 0 203.0.113.87
# Static routes for NAT addresses - see redistribute static command
route dmz 209.165.202.129 255.255.255.255 10.3.5.99
route inside 209.165.202.130 255.255.255.255 192.168.9.99

# The global service policy
class-map inspection_default
  match default-inspection-traffic
policy-map type inspect dns preset_dns_map
  parameters
    message-length maximum client auto
    message-length maximum 512
    dns-guard
    protocol-enforcement
    nat-rewrite
policy-map global_policy
  class inspection_default
    inspect dns preset_dns_map
    inspect ftp
    inspect h323 h225 _default_h323_map
    inspect h323 ras _default_h323_map
    inspect ip-options _default_ip_options_map
    inspect netbios
    inspect rsh
```

```
    inspect rtsp
    inspect skinny
    inspect esmtp _default_esmtp_map
    inspect sqlnet
    inspect sunrpc
    inspect tftp
    inspect sip
    inspect xdmcp
service-policy global_policy global
```

History for Traffic Zones

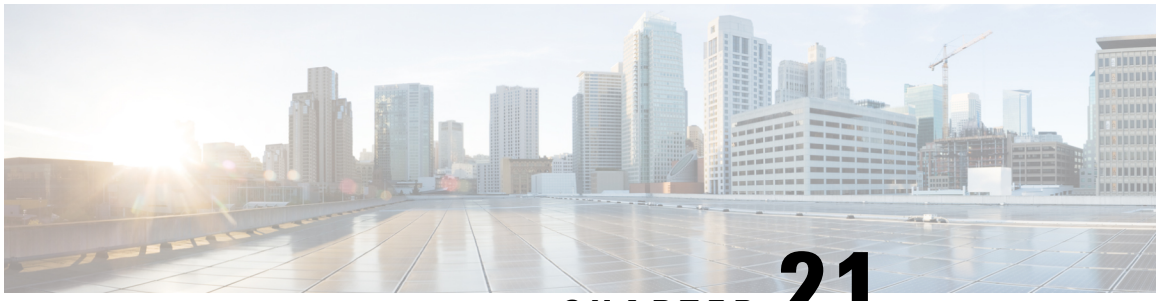
Feature Name	Platform Releases	Description
Traffic Zones	9.3(2)	You can group interfaces together into a traffic zone to accomplish traffic load balancing (using Equal Cost Multi-Path (ECMP) routing), route redundancy, and asymmetric routing across multiple interfaces. **Note** You cannot apply a security policy to a named zone; the security policy is interface-based. When interfaces in a zone are configured with the same access rule, NAT, and service policy, then load-balancing and asymmetric routing operate correctly. We introduced or modified the following commands: **zone**, **zone-member**, **show running-config zone**, **clear configure zone**, **show zone**, **show asp table zone**, **show nameif zone**, **show conn long**, **show local-host zone**, **show route zone**, **show asp table routing**, **clear conn zone**, **clear local-host zone**.
clear local-host command	9.14(1)	The **clear local-host** command and all of its attributes and keywords were deprecated. They will be removed in a future release.



PART IV

Basic Settings

- Basic Settings, on page 735
- DHCP and DDNS Services, on page 759
- Digital Certificates, on page 787
- ARP Inspection and the MAC Address Table, on page 825



CHAPTER 21

Basic Settings

This chapter describes how to configure basic settings on the ASA that are typically required for a functioning configuration.

- Set the Hostname, Domain Name, and the Enable and Telnet Passwords, on page 735
- Set the Date and Time, on page 737
- Configure the Master Passphrase, on page 744
- Configure the DNS Server, on page 748
- Configure the Hardware Bypass and Dual Power Supply (Cisco ISA 3000), on page 750
- Adjust ASP (Accelerated Security Path) Performance and Behavior, on page 752
- Monitoring the DNS Cache, on page 754
- History for Basic Settings, on page 754

Set the Hostname, Domain Name, and the Enable and Telnet Passwords

To set the hostname, domain name, and the enable and Telnet passwords, perform the following steps.

Before you begin

Before you set the hostname, domain name, and the enable and Telnet passwords, check the following requirements:

- In multiple context mode, you can configure the hostname and domain name in both the system and context execution spaces.
- For the enable and Telnet passwords, set them in each context; they are not available in the system.
- To change from the system to a context configuration, enter the **changeto context** *name* command.

Procedure

Step 1 Specify the hostname for the ASA or for a context. The default hostname is “asa.”

hostname *name*

Example:

```
ciscoasa(config)# hostname myhostnamexample12345
```

This name can be up to 63 characters. The hostname must start and end with a letter or digit, and have only letters, digits, or a hyphen.

When you set a hostname for the ASA, that name appears in the command line prompt. If you establish sessions to multiple devices, the hostname helps you keep track of where you enter commands.

For multiple context mode, the hostname that you set in the system execution space appears in the command line prompt for all contexts. The hostname that you optionally set within a context does not appear in the command line, but can be used by the **banner** command **$(hostname)** token.

Step 2 Specify the domain name for the ASA. The default domain name is default.domain.invalid.

domain-name *name*

Example:

```
ciscoasa(config)# domain-name example.com
```

The ASA appends the domain name as a suffix to unqualified names. For example, if you set the domain name to "example.com" and specify a syslog server by the unqualified name of "jupiter," then the ASA qualifies the name to "jupiter.example.com."

Step 3 Change the enable password. By default, the enable password is blank, but you are prompted to change it the first time you enter the **enable** command.

enable password *password*

Example:

```
ciscoasa(config)# enable password Pa$$w0rd
```

The enable password lets you enter privileged EXEC mode if you do not configure enable authentication. The enable password also lets you log into ASDM with a blank username if you do not configure HTTP authentication.

The *password* argument is a case-sensitive password of 8 to 127 characters long, and can be any combination of ASCII printable characters (character codes 32-126), with the following exceptions:

- No spaces
- No question marks
- You cannot use three or more consecutive sequential or repetitive ASCII characters. For example, the following passwords will be rejected:
  - **abc**user1
  - user**543**
  - user**aaaa**
  - user2**666**

This command changes the password for the highest privilege level (15). If you configure local command authorization, you can set enable passwords for each privilege level from 0 to 15 using the following syntax:

enable password *password* **level** *number*

The **encrypted** keyword (for passwords 32 characters and fewer in 9.6 and earlier) or the **pbkdf2** keyword (for passwords longer than 32 characters in 9.6 and later, and passwords of all lengths in 9.7 and later) indicates that the password is encrypted (using an MD5-based hash or a PBKDF2 (Password-Based Key Derivation Function 2) hash using SHA-512). Note that already existing passwords continue to use the MD5-based hash unless you enter a new password. When you define a password in the **enable password** command, the ASA encrypts it when it saves it to the configuration for security purposes. When you enter the **show running-config** command, the **enable password** command does not show the actual password; it shows the encrypted password followed by the **encrypted** or **pbkdf2** keyword. For example, if you enter the password "test," the **show running-config** command output would appear as something similar to the following:

```
username user1 password DLaUiAX3l78qgoB5c7iVNw== encrypted
```

The only time you would actually enter the **encrypted** or **pbkdf2** keyword at the CLI is if you are cutting and pasting a configuration file for use in another ASA, and you are using the same password.

You cannot reset the password to a blank value.

Step 4 Set the login password for Telnet access. There is no default password.

The login password is used for Telnet access when you do not configure Telnet authentication.

passwd *password* [**encrypted**]

Example:

```
ciscoasa(config)# passwd cisco12345
```

The *password* is a case-sensitive password of up to 16 alphanumeric and special characters. You can use any character in the password except a question mark or a space.

The password is saved in the configuration in encrypted form, so you cannot view the original password after you enter it. If for some reason you need to copy the password to another ASA but do not know the original password, you can enter the **passwd** command with the encrypted password and the **encrypted** keyword. Normally, you only see this keyword when you enter the **show running-config passwd** command.

Set the Date and Time



Note Do not set the date and time for the Firepower 2100 in Platform mode, 4100/9300; the ASA receives these settings from the chassis.

Set the Time Zone and Daylight Saving Dates

To set the time zone and daylight saving date range, perform the following steps.

Procedure

Step 1 Set the time zone. By default, the time zone is UTC.

- Firepower and Secure Firewall models:

  clock timezone *zone*

  - *zone*—Enter the **clock timezone ?** command to see a list of acceptable time zone names.

  Example:

```
ciscoasa(config)# clock timezone ?
Available timezones:
CET
CST6CDT
Cuba
EET
Egypt
Eire
EST
EST5EDT
Factory
GB
GB-Eire
GMT
GMT0
GMT-0
GMT+0
Greenwich
Hongkong
HST
Iceland
Iran
Israel
Jamaica
Japan
[...]

ciscoasa(config)# clock timezone US/?

configure mode commands/options:
  US/Alaska          US/Aleutian    US/Arizona    US/Central
  US/East-Indiana    US/Eastern     US/Hawaii     US/Indiana-Starke
  US/Michigan        US/Mountain    US/Pacific

ciscoasa(config)# clock timezone US/Mountain
```

- All other models:

  clock timezone *zone* [-]*hours* [*minutes*]

  - *zone*—Specifies the time zone as a string, for example, PST for Pacific Standard Time.
  - [-]*hours*—Sets the number of hours of offset from UTC. For example, PST is -8 hours.

- *minutes*—Sets the number of minutes of offset from UTC.

Example:

```
ciscoasa(config)# clock timezone PST -8
```

Step 2 (ASAv, and ISA 3000) Enter one of the following commands to change the date range for daylight saving time from the default. The default recurring date range is from 2:00 a.m. on the second Sunday in March to 2:00 a.m. on the first Sunday in November.

- Set the start and end dates for daylight saving time as a specific date in a specific year. If you use this command, you need to reset the dates every year.

  clock summer-time *zone* **date** {*day month* | *month day*} *year hh***:***mm* {*day month* | *month day*} *year hh***:***mm* [*offset*]

  - *zone* —Specifies the time zone as a string, for example, PDT for Pacific Daylight Time.
  - *day* —Sets the day of the month, from 1 to 31. You can enter the day and month as **A**pril 1 or as 1 April, for example, depending on your standard date format.
  - *month* —Sets the month as a string. You can enter the day and month as April 1 or as 1 April, depending on your standard date format.
  - *year* —Sets the year using four digits, for example, 2004. The year range is 1993 to 2035.
  - *hh:mm* —Sets the hour and minutes in 24-hour time.
  - *offset* —Sets the number of minutes to change the time for daylight saving time. By default, the value is 60 minutes.

  Example:

  ```
  ciscoasa(config)# clock summer-time PDT 1 April 2010 2:00 60
  ```

- Specify the start and end dates for daylight saving time, in the form of a day and time of the month, and not a specific date in a year. This command enables you to set a recurring date range that you do not need to change yearly.

  clock summer-time *zone* **recurring** [*week weekday month hh***:***mm week weekday month hh***:***mm*] [*offset*]

  - *zone*—Specifies the time zone as a string, for example, PDT for Pacific Daylight Time.
  - *week* —Specifies the week of the month as an integer between 1 and 4 or as the words first or last. For example, if the day might fall in the partial fifth week, then specify last.
  - *weekday* —Specifies the day of the week: Monday, Tuesday, Wednesday, and so on.
  - *month* —Sets the month as a string.
  - *hh:mm* —Sets the hour and minutes in 24-hour time.
  - *offset* —Sets the number of minutes to change the time for daylight savings time. By default, the value is 60 minutes.

  Example:

```
ciscoasa(config)# clock summer-time PDT recurring first Monday April 2:00 60
```

Set the Date and Time Using an NTP Server

NTP is used to implement a hierarchical system of servers that provide a precisely synchronized time among network systems. This kind of accuracy is required for time-sensitive operations, such as validating CRLs, which include a precise time stamp. You can configure multiple NTP servers. The ASA chooses the server with the lowest stratum—a measure of how reliable the data is.

Time derived from an NTP server overrides any time set manually.

The ASA supports NTPv4.

Before you begin

In multiple context mode, you can set the time in the system configuration only.

Procedure

Step 1 (Optional) Enable authentication with an NTP server.

a) Enable authentication.

ntp authenticate

Example:

```
ciscoasa(config)# ntp authenticate
```

When you enable NTP authentication, you must also specify a key ID in the **ntp trusted-key** command and associate that key with the server with the **ntp server key** command. Configure the actual key for the ID with the **ntp authentication-key** command. If you have multiple servers, configure a separate ID for each sever.

b) Specify an authentication key ID to be a trusted key, which is required for authentication with an NTP server.

ntp trusted-key *key_id*

Example:

```
ciscoasa(config)# ntp trusted-key 1
ciscoasa(config)# ntp trusted-key 2
ciscoasa(config)# ntp trusted-key 3
ciscoasa(config)# ntp trusted-key 4
```

The *key_id* argument is a value between 1 and 4294967295. You can enter multiple trusted keys for use with multiple servers.

c) Set a key to authenticate with an NTP server.

ntp authentication-key *key_id* {**md5** | **sha1** | **sha256** | **sha512** | **cmac**} *key*

Example:

```
ciscoasa(config)# ntp authentication-key 1 md5 aNiceKey1
ciscoasa(config)# ntp authentication-key 2 md5 aNiceKey2
ciscoasa(config)# ntp authentication-key 3 md5 aNiceKey3
ciscoasa(config)# ntp authentication-key 4 md5 aNiceKey4
```

- *key_id*—Sets the ID that you set using the **ntp trusted-key** command.
- {**md5** | **sha1** | **sha256** | **sha512** | **cmac**} —Sets the algorithm.
- *key*—Sets the key as a string up to 32 characters long.

Step 2 Identify an NTP server.

ntp server {*ipv4_address* | *ipv6_address* }[**key** *key_id*] [**source** *interface_name*] [**prefer**]

Example:

```
ciscoasa(config)# ntp server 10.1.1.1 key 1 prefer
ciscoasa(config)# ntp server 10.2.1.1 key 2
ciscoasa(config)# ntp server 2001:DB8::178 key 3
ciscoasa(config)# ntp server 2001:DB8::8945:ABCD key 4
```

If you enabled NTP authentication (**ntp authenticate**), you must specify the **key** *key_id* argument using the ID that you set using the **ntp trusted-key** command.

The **source** *interface_name* keyword-argument pair identifies the outgoing interface for NTP packets if you do not want to use the default interface in the routing table. Because the system does not include any interfaces in multiple context mode, specify an interface name defined in the admin context.

The **prefer** keyword sets this NTP server as the preferred server if multiple servers have similar accuracy. NTP uses an algorithm to determine which server is the most accurate and synchronizes to that one. If servers are of similar accuracy, then the **prefer** keyword specifies which of those servers to use. However, if a server is significantly more accurate than the preferred one, the ASA uses the more accurate one. For example, the ASA uses a server of stratum 2 over a server of stratum 3 that is preferred.

You can identify multiple servers; the ASA uses the most accurate server.

Set the Date and Time Manually

To set the date and time manually, perform the following steps:

Before you begin

In multiple context mode, you can set the time in the system configuration only.

Procedure

Set the date time manually.

clock set *hh*:*mm*:*ss* {*month day* | *day month*} *year*

Example:

```
ciscoasa# clock set 20:54:00 april 1 2004
```

The *hh*:*mm*:*ss* argument sets the hour, minutes, and seconds in 24-hour time. For example, enter 20:54:00 for 8:54 pm.

The day value sets the day of the month, from 1 to 31. You can enter the day and month as april 1 or as 1 april, for example, depending on your standard date format.

The month value sets the month. Depending on your standard date format, you can enter the day and month as april 1 or as 1 april.

The year value sets the year using four digits, for example, 2004. The year range is from 1993 to 2035.

The default time zone is UTC. If you change the time zone after you enter the **clock set** command using the **clock timezone** command, the time automatically adjusts to the new time zone.

This command sets the time in the hardware chip, and does not save the time in the configuration file. This time endures reboots. Unlike the other **clock** commands, this command is a privileged EXEC command. To reset the clock, you need to set a new time with the clock set command.

Configure Precision Time Protocol (ISA 3000)

The Precision Time Protocol (PTP) is a time-synchronization protocol developed to synchronize the clocks of various devices in a packet-based network. These device clocks are generally of varying precision and stability. The protocol is designed specifically for industrial, networked measurement and control systems, and is optimal for use in distributed systems because it requires minimal bandwidth and little processing overhead.

A PTP system is a distributed, networked system consisting of a combination of PTP and non-PTP devices. PTP devices include ordinary clocks, boundary clocks and transparent clocks. Non-PTP devices include network switches, routers and other infrastructure devices.

You can configure the ASA device to be a transparent clock. The ASA device does not synchronize its clock with the PTP clocks. The ASA device will use the PTP default profile, as defined on the PTP clocks.

When you configure the PTP devices, you define a domain number for the devices that are meant to function together.Thus, you can configure multiple PTP domains, and then configure each non-PTP device to use the PTP clocks for one specific domain.

Before you begin

- This feature is only available on the ISA 3000.
- Use of PTP is supported in single context mode only.

- Cisco PTP supports multicast PTP messages only.
- PTP is enabled on all ISA 3000 interfaces in transparent mode by default. In routed mode, you must add the necessary configuration to ensure that the PTP packets are allowed to flow through the device.
- PTP is available only for IPv4 networks, not for IPv6 networks.
- PTP configuration is supported on physical Ethernet interfaces, whether stand-alone or bridge group members. It is not supported on:
  - Management interface.
  - Subinterfaces, EtherChannels, BVIs. or any other virtual interfaces.
- PTP flows on VLAN subinterfaces are supported, assuming the appropriate PTP configuration is present on the parent interface.
- You must ensure that PTP packets are allowed to flow through the device. In transparent firewall mode, the access list configuration to allow PTP traffic is configured by default. PTP traffic is identified by UDP ports 319 and 320, and destination IP address 224.0.1.129, so in routed firewall mode any ACL that allows this traffic should be acceptable.
- In routed firewall mode, you must also enable multicast routing for PTP multicast groups:
  - Enter the global configuration mode command **multicast-routing**.
  - And for each interface that is not a bridge group member, and on which PTP is enabled, enter the interface configuration command **igmp join-group 224.0.1.129** to statically enable PTP multicast group membership. This command is not supported or needed for bridge group members.

Procedure

Step 1 Specify the domain number of all ports of the device:

ptp domain *domain_num*

Example:

```
ciscoasa(config)# ptp domain 54
```

The *domain_num* argument is the domain number for all ports on the device. Packets received on a different domain are treated like regular multicast packets and will not undergo any PTP processing. This value can be from zero to 255; the default value is zero. Enter the domain number that is configured on the PTP devices in your network.

Step 2 (Optional) Configure the PTP clock mode on the device:

ptp mode e2etransparent

Example:

```
ciscoasa(config)# ptp mode e2etransparent
```

This command enables End-to-End Transparent mode on all PTP-enabled interfaces.

Step 3 Enable PTP on an interface:

ptp enable

Enable PTP on each interface through which the system can contact a PTP clock in the configured domain.

Example:

```
ciscoasa(config)# interface gigabitethernet1/2
ciscoasa(config-if)# ptp enable
```

Configure the Master Passphrase

The master passphrase allows you to securely store plain text passwords in encrypted format and provides a key that is used to universally encrypt or mask all passwords, without changing any functionality. Features that use the master passphrase include the following:

- OSPF
- EIGRP
- VPN load balancing
- VPN (remote access and site-to-site)
- Failover
- AAA servers
- Logging
- Shared licenses

Add or Change the Master Passphrase

To add or change the master passphrase, perform the following steps.

Before you begin

- This procedure will only be accepted in a secure session, for example by console, SSH, or ASDM via HTTPS.
- If failover is enabled but no failover shared key is set, an error message appears if you change the master passphrase, informing you that you must enter a failover shared key to protect the master passphrase changes from being sent as plain text.
- Enabling or changing password encryption in Active/Standby failover causes a **write standby**, which replicates the active configuration to the standby unit. Without this replication, the encrypted passwords on the standby unit will differ even though they use the same passphrase; configuration replication ensures that the configurations are the same. For Active/Active failover, you must manually enter **write standby**. A **write standby** can cause traffic interruption in Active/Active mode, because the configuration is cleared on the secondary unit before the new configuration is synced. You should make all contexts active on the primary ASA using the **failover active group 1** and **failover active group 2** commands, enter **write standby**, and then restore the group 2 contexts to the secondary unit using the **no failover active group 2** command.

Procedure

Step 1 Set the passphrase used for generating the encryption key. The passphrase must be between 8 and 128 characters long. All characters except a backspace and double quotes are accepted for the passphrase. If you do not enter the new passphrase in the command, you are prompted for it. To change the passphrase, you must enter the old passphrase.

key config-key password-encryption [*new_passphrase* [*old_passphrase*]]

Example:

```
ciscoasa(config)# key config-key password-encryption
Old key: bumblebee
New key: haverford
Confirm key: haverford
```

Note
Use the interactive prompts to enter passwords to avoid having the passwords logged in the command history buffer.

Use the **no key config-key password-encrypt** command with caution, because it changes the encrypted passwords into plain text passwords. You may use the **no** form of this command when downgrading to a software version that does not support password encryption.

Step 2 Enable password encryption.

password encryption aes

Example:

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

As soon as password encryption is enabled and the master passphrase is available, all the user passwords will be encrypted. The running configuration will show the passwords in the encrypted format.

If the passphrase is not configured at the time that password encryption is enabled, the command will succeed in anticipation that the passphrase will be available in the future.

If you later disable password encryption using the **no password encryption aes** command, all existing encrypted passwords are left unchanged, and as long as the master passphrase exists, the encrypted passwords will be decrypted, as required by the application.

Step 3 Save the runtime value of the master passphrase and the resulting configuration.

write memory

Example:

```
ciscoasa(config)# write memory
```

If you do not enter this command, passwords in startup configuration may still be visible if they were not saved with encryption previously. In addition, in multiple context mode the master passphrase is changed in the system context configuration. As a result, the passwords in all contexts will be affected. If the write memory

command is not entered in the system context mode, but not in all user contexts, then the encrypted passwords in user contexts may be stale. Alternatively, use the write memory all command in the system context to save all configurations.

Examples

The following example shows that no previous key was present:

```
ciscoasa(config)# key config-key password-encryption 12345678
```

The following example shows that a key already exists:

```
ciscoasa(config)# key config-key password-encryption 23456789
Old key: 12345678
```

In the following example, you enter the command without parameters so that you will be prompted for keys. Because a key already exists, you are prompted for it.

```
ciscoasa(config)# key config-key password-encryption
Old key: 12345678
New key: 23456789
Confirm key: 23456789
```

In the following example, there is no existing key, so you are not prompted to supply it.

```
ciscoasa(config)# key config-key password-encryption
New key: 12345678
Confirm key: 12345678
```

Disable the Master Passphrase

Disabling the master passphrase reverts encrypted passwords into plain text passwords. Removing the passphrase might be useful if you downgrade to a previous software version that does not support encrypted passwords.

Before you begin

- You must know the current master passphrase to disable it. See Remove the Master Passphrase, on page 747 if you do not know the passphrase.
- This procedure works only in a secure session; that is, by Telnet, SSH, or ASDM via HTTPS.

To disable the master passphrase, perform the following steps:

Procedure

Step 1 Remove the master passphrase. If you do not enter the passphrase in the command, you are prompted for it.

no key config-key password-encryption [*old_passphrase*]]

Example:

```
ciscoasa(config)# no key config-key password-encryption

Warning! You have chosen to revert the encrypted passwords to plain text.
This operation will expose passwords in the configuration and therefore
exercise caution while viewing, storing, and copying configuration.

Old key: bumblebee
```

Step 2 Save the runtime value of the master passphrase and the resulting configuration.

write memory

Example:

```
ciscoasa(config)# write memory
```

The non-volatile memory containing the passphrase will be erased and overwritten with the 0xFF pattern.

In multiple mode, the master passphrase is changed in the system context configuration. As a result, the passwords in all contexts will be affected. If the write memory command is entered in the system context mode, but not in all user contexts, then the encrypted passwords in user contexts may be stale. Alternatively, use the write memory all command in the system context to save all configurations.

Remove the Master Passphrase

You cannot recover the master passphrase. If the master passphrase is lost or unknown, you can remove it.

To remove the master passphrase, perform the following steps:

Procedure

Step 1 Remove the master key and the configuration that includes the encrypted passwords.

write erase

Example:

```
ciscoasa(config)# write erase
```

Step 2 Reload the ASA with the startup configuration, without any master key or encrypted passwords.

reload

Example:

```
ciscoasa(config)# reload
```

Configure the DNS Server

You need to configure a DNS server so that the ASA can resolve host names to IP addresses. You also must configure a DNS server to use fully qualified domain names (FQDN) network objects in access rules.

Some ASA features require use of a DNS server to access external servers by domain name. Other features, such as the **ping** or **traceroute** command, let you enter a name that you want to ping or traceroute, and the ASA can resolve the name by communicating with a DNS server. Many SSL VPN and certificate commands also support names.

By default, there is a default DNS server group called DefaultDNS. You can create multiple DNS server groups: only one group can be active at a time, however.. Other DNS server groups can be configured for VPN tunnel groups. See the **tunnel-group** command in the command reference for more information.

Note The ASA has limited support for using the DNS server, depending on the feature. For example, most commands require you to enter an IP address and can only use a name when you manually configure the **name** command to associate a name with an IP address and enable use of the names using the names command.

Before you begin

Make sure that you configure the appropriate routing and access rules for any interface on which you enable DNS domain lookup so you can reach the DNS server.

Procedure

Step 1 Enable the ASA to send DNS requests to a DNS server to perform a name lookup for supported commands.

dns domain-lookup *interface_name*

If you do not enable DNS lookup on an interface, then the ASA will not communicate with the DNS server on that interface. Make sure to enable DNS lookup on all interfaces that will be used to access DNS servers.

Example:

```
ciscoasa(config)# dns domain-lookup inside
ciscoasa(config)# dns domain-lookup outside
```

Step 2 Create one or more DNS server groups and add servers to the groups.

a) Name the DNS server group.

dns server-group *name*

To configure the default DefaultDNS server group, specify DefaultDNS for the name.

Example:

```
ciscoasa(config)# dns server-group DefaultDNS
```

b) Specify one or more DNS servers for the group.

name-server *ip_address* [*ip_address2*] [...] [*ip_address6*] *[interface_name]*

You can enter all six IP addresses in the same command, separated by spaces, or you can enter each command separately.

(Optional) Specify the *interface_name* through which the ASA communicates with the server. If you do not specify the interface, the ASA checks the data routing table; if there are no matches, it then checks the management-only routing table.

The ASA tries each DNS server in order until it receives a response.

Example:

```
ciscoasa(config-dns-server-group)# name-server 10.1.1.5 192.168.1.67 209.165.201.6 outside
```

c) Configure the domain name appended to the hostname if it is not fully qualified.

domain-name *name*

Example:

```
ciscoasa(config-dns-server-group)# domain-name example.com
```

d) (Optional) Configure additional properties of the DNS server group.

Use the following commands to change the characteristics of the group, if the default settings are not appropriate for your network.

- **timeout** *seconds*—The number of seconds, from 1 to 30, to wait before trying the next DNS server. The default is 2 seconds. Each time the ASA retries the list of servers, this timeout doubles.
- **retries** *number*—The number of times, from 0 to 10, to retry the list of DNS servers when the ASA does not receive a response.
- **expire-entry-timer minutes** *number*—The minimum TTL for the DNS entry, in minutes. If the expiration timer is longer than the entry's TTL, the TTL is increased to the expire entry time value. If the TTL is longer than the expiration timer, the expire entry time value is ignored: no additional time is added to the TTL in this case. Upon expiration, the entry is removed from the DNS lookup table. Removing an entry requires that the table be recompiled, so frequent removals can increase the processing load on the device. Because some DNS entries can have very short TTL (as short as three seconds), you can use this setting to virtually extend the TTL. The default is 1 minute (that is, the minimum TTL for all resolutions is 1 minute). The range is 1 to 65535 minutes. This option is used when resolving FQDN network objects only.
- **poll-timer minutes** *number*—The time, in minutes, of the polling cycle used to resolve FQDN network/host objects to IP addresses. FQDN objects are resolved only if they are used in a firewall

policy. The timer determines the maximum time between resolutions; the DNS entry's time-to-live (TTL) value is also used to determine when to update to IP address resolution, so individual FQDNs might be resolved more frequently than the polling cycle. The default is 240 (four hours). The range is 1 to 65535 minutes.

e) Repeat the above steps to add additional DNS server groups.

Step 3 Specify the default DNS group.

dns-group *name*

By default, DefaultDNS is specified. If you configured other groups, you can specify a different default group using this command. You can only have one active group.

Example:

```
ciscoasa(config)# dns-group new_default_group
```

Configure the Hardware Bypass and Dual Power Supply (Cisco ISA 3000)

You can enable the hardware bypass so that traffic continues to flow between an interface pair during a power outage. Supported interface pairs are copper GigabitEthernet 1/1 & 1/2; and GigabitEthernet 1/3 & 1/4. When the hardware bypass is active, no firewall functions are in place, so make sure you understand the risks of allowing traffic through. See the following hardware bypass guidelines:

- This feature is only available on the Cisco ISA 3000 appliance.
- If you have a fiber Ethernet model, only the copper Ethernet pair (GigabitEthernet 1/1 & 1/2) supports hardware bypass.
- When the ISA 3000 loses power and goes into hardware bypass mode, only the supported interface pairs can communicate; when using the default configuration, inside1 <---> inside2, and outside1 <---> outside2 can no longer communicate. Any existing connections between these interfaces will be lost.
- We suggest that you disable TCP sequence randomization (as described in this procedure). If randomization is enabled (the default), then when the hardware bypass is activated, TCP sessions will need to be re-established. By default, the ISA 3000 rewrites the initial sequence number (ISN) of TCP connections passing through it to a random number. When the hardware bypass is activated, the ISA 3000 is no longer in the data path and does not translate the sequence numbers; the receiving client receives an unexpected sequence number and drops the connection. Even with TCP sequence randomization disabled, some TCP connections will have to be re-established because of the link that is temporarily down during the switchover.
- Cisco TrustSec connections on hardware bypass interfaces are dropped when hardware bypass is activated. When the ISA 3000 powers on and hardware bypass is deactivated, the connections are renegotiated.
- When the hardware bypass is deactivated, and traffic resumes going through the ISA 3000 data path, some existing TCP sessions need to be re-established because of the link that is temporarily down during the switchover.

- When hardware bypass is active, the Ethernet PHYs are disconnected, so the ASA is unable to determine the interface status. Interfaces may appear to be in a down state.

For dual power supplies in the ISA 3000, you can establish dual power supplies as the expected configuration in the ASA OS. If one power supply fails, the ASA issues an alarm. By default, the ASA expects a single power supply and won't issue an alarm as long as it includes one working power supply.

Before you begin

- You must attach the hardware bypass interfaces to access ports on the switch. Do not attach them to trunk ports.

Procedure

Step 1 Configure the hardware bypass to activate during a power failure:

hardware-bypass GigabitEthernet {**1/1-1/2** | **1/3-1/4**} [**sticky**]

Example:

```
ciscoasa(config)# hardware-bypass GigabitEthernet 1/1-1/2
ciscoasa(config)# hardware-bypass GigabitEthernet 1/3-1/4
```

The **sticky** keyword keeps the appliance in hardware bypass mode after the power comes back and the appliance boots up. In this case, you need to manually turn off the hardware bypass when you are ready; this option lets you control when the brief interruption in traffic occurs.

Step 2 Manually activate or deactivate the hardware bypass:

[**no**] **hardware-bypass manual GigabitEthernet** {**1/1-1/2** | **1/3-1/4**}

Example:

```
ciscoasa# hardware-bypass manual GigabitEthernet 1/1-1/2
ciscoasa# no hardware-bypass manual GigabitEthernet 1/1-1/2
```

Step 3 (Optional) Configure the hardware bypass to remain active until after the ASA FirePOWER module boots up:

hardware-bypass boot-delay module-up sfr

You must enable hardware bypass without the **sticky** option for the boot delay to operate. Without the **hardware-bypass boot-delay** command, the hardware bypass is likely to become inactive before the ASA FirePOWER module finishes booting up. This scenario can cause traffic to be dropped if you configured the module to fail-close, for example.

Step 4 Disable TCP sequence randomization. This example shows how to disable randomization for all traffic by adding the setting to the default configuration.

policy-map global_policy

class sfrclass

set connection random-sequence-number disable

If you later decide to turn it back on, replace "disable" with **enable**.

Step 5 Establish dual power supplies as the expected configuration:

power-supply dual

Step 6 Save the configuration.

write memory

The behavior of hardware bypass after the system comes online is determined by the configuration setting in the startup configuration, so you must save your running configuration.

Adjust ASP (Accelerated Security Path) Performance and Behavior

The ASP is an implementation layer that puts your policies and configurations into action. It is not of direct interest except during troubleshooting with the Cisco Technical Assistance Center. However, there are a few behaviors related to performance and reliability that you can adjust.

Choose a Rule Engine Transactional Commit Model

By default, when you change a rule-based policy (such as access rules), the changes become effective immediately. However, this immediacy comes with a slight cost in performance. The performance cost is more noticeable for very large rule lists in a high connections-per-second environment, for example, when you change a policy with 25,000 rules while the ASA is handling 18,000 connections per second.

The performance is affected because the rule engine compiles rules to enable faster rule lookup. By default, the system also searches uncompiled rules when evaluating a connection attempt so that new rules can be applied; because the rules are not compiled, the search takes longer.

You can change this behavior so that the rule engine uses a transactional model when implementing rule changes, continuing to use the old rules until the new rules are compiled and ready for use. With the transactional model, performance should not drop during the rule compilation. The following table clarifies the behavioral difference.

Model	Before Compilation	During Compilation	After Compilation
Default	Matches old rules.	Match new rules. (The rate for connections per second decreases.)	Matches new rules.
Transactional	Matches old rules.	Match old rules. (The rate for connections per second is unaffected.)	Matches new rules.

An additional benefit of the transactional model is that, when replacing an ACL on an interface, there is no gap between deleting the old ACL and applying the new one. This feature reduces the chances that acceptable connections may be dropped during the operation.

Before you begin

- Transactional commit is not recommended for access control rules if you use FQDN objects for hostnames whose resolution can frequently change, as the access group compilation might never completely resolve due to DNS churn. If you still want to use transactional commit, consider lengthening the time for DNS expiration.
- If you enable the transactional model for a rule type, syslogs to mark the beginning and the end of the compilation are generated. These syslogs are numbered 780001 through 780004.

Procedure

Enable the transactional commit model for the rule engine:

asp rule-engine transactional-commit *option*

Where the options are:

- **access-group**—Access rules applied globally or to interfaces.
- **nat**—Network Address Translation rules.

Example:

```
ciscoasa(config)# asp rule-engine transactional-commit access-group
```

Enable ASP Load Balancing

The ASP load balancing mechanism helps avoid the following issues:

- Overruns caused by sporadic traffic spikes on flows
- Overruns caused by bulk flows oversubscribing specific interface receive rings
- Overruns caused by relatively heavily overloaded interface receive rings, in which a single core cannot sustain the load.

ASP load balancing allows multiple cores to work simultaneously on packets that were received from a single interface receive ring. If the system drops packets, and the **show cpu** command output is far less than 100%, then this feature may help your throughput if the packets belong to many unrelated connections.

Note ASP load balancing is disabled on the ASAv. With the integration of DPDK (Dataplane Development Kit) into the ASAv's accelerated security path (ASP), the ASAv shows better performance with this feature disabled.

Procedure

Step 1 Enable the automatic switching on and off of ASP load balancing:

asp load-balance per-packet auto

Step 2 Manually enable ASP load balancing:

asp load-balance per-packet

ASP load balancing is enabled until you manually disable it, even if you also have the **auto** command enabled.

Step 3 Manually disable ASP load balancing:

no asp load-balance per-packet

This command only applies if you manually enabled ASP load blancing. If you also enabled the **auto** command, then the system reverts to automatically enabling or disabling ASP load balancing.

Monitoring the DNS Cache

The ASA provides a local cache of DNS information from external DNS queries that are sent for certain clientless SSL VPN and certificate commands. Each DNS translation request is first looked for in the local cache. If the local cache has the information, the resulting IP address is returned. If the local cache can not resolve the request, a DNS query is sent to the various DNS servers that have been configured. If an external DNS server resolves the request, the resulting IP address is stored in the local cache with its corresponding hostname.

See the following command for monitoring the DNS cache:

- **show dns-hosts**

  This command shows the DNS cache, which includes dynamically learned entries from a DNS server as well as manually entered name and IP addresses using the name command.

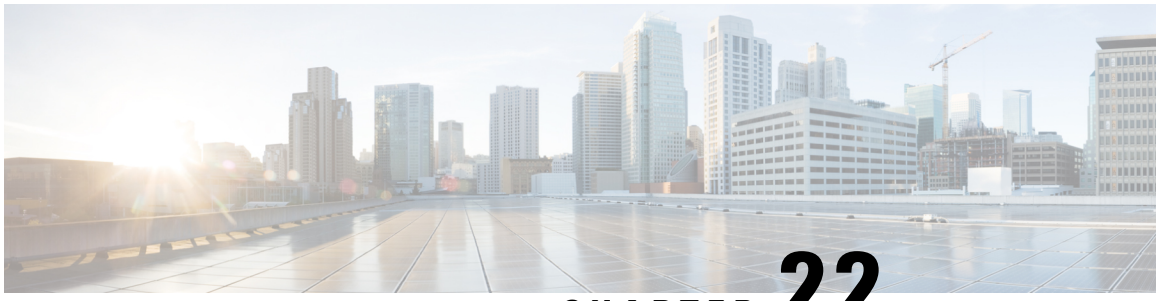
History for Basic Settings

Feature Name	Platform Releases	Description
Trusted DNS servers for network-service object domain resolution.	9.17(1)	You can specify which DNS servers the system should trust when resolving domain names in network-service objects. This feature ensures that any DNS domain name resolutions acquire IP addresses from trusted sources. New/Modified commands: **dns trusted-source**, **show dns trusted-source**

Feature Name	Platform Releases	Description
Change in DNS entry TTL behavior	9.17(1)	Formerly, the configured value was added to the existing TTL of each entry (the default was 1 minute). Now, if the expiration timer is longer than the entry's TTL, the TTL is increased to the expire entry time value. If the TTL is longer than the expiration timer, the expire entry time value is ignored; no additional time is added to the TTL in this case. New/Modified commands: **expire-entry-timer minutes**
Stronger local user and enable password requirements	9.17(1)	For local users and the enable password, the following password requirements were added: • Password length—Minimum 8 characters. Formerly, the minimum was 3 characters. • Repetitive and sequential characters—Three or more consecutive sequential or repetitive ASCII characters are disallowed. For example, the following passwords will be rejected: • **abc**user1 • user**543** • user**aaaa** • user2**666** New/Modified commands: **enable password**, **username**
NTPv4 support	9.14(1)	The ASA now supports NTPv4. No modified commands.
Additional NTP authentication algorithms	9.13(1)	Formerly, only MD5 was supported for NTP authentication. The ASA now supports the following algorithms: • MD5 • SHA-1 • SHA-256 • SHA-512 • AES-CMAC New/Modified commands: **ntp authentication-key**
NTP support on IPv6	9.12(1)	You can now specify an IPv6 address for the NTP server. New/Modified commands: **ntp server**

Feature Name	Platform Releases	Description
enable password change now required on login	9.12(1)	The default **enable** password is blank. When you try to access privileged EXEC mode on the ASA, you are now required to change the password to a value of 3 to 127 characters. You cannot keep it blank. The **no enable password** command is no longer supported. At the CLI, you can access privileged EXEC mode using the **enable** command, the **login** command (with a user at privilege level 2+), or an SSH or Telnet session when you enable **aaa authorization exec auto-enable**. All of these methods require you to set the enable password. This password change requirement is not enforced for ASDM logins. In ASDM, by default you can log in without a username and with the **enable** password. New/Modified commands: **enable password**
ASP load balancing is disabled on the ASAv	9.10(1)	With the recent integration of DPDK (Dataplane Development Kit) into the ASAv's accelerated security path (ASP), the ASAv shows better performance with this feature disabled.
Automatic ASP load balancing now supported for the ASAv	9.8(1)	Formerly, you could only manually enable and disable ASP load balancing. We modified the following command: **asp load-balance per-packet auto**
PBKDF2 hashing for all local **username** and **enable** passwords	9.7(1)	Local **username** and **enable** passwords of all lengths are stored in the configuration using a PBKDF2 (Password-Based Key Derivation Function 2) hash using SHA-512. Previously, passwords 32 characters and shorter used the MD5-based hashing method. Already existing passwords continue to use the MD5-based hash unless you enter a new password. See the "Software and Configurations" chapter in the General Operations Configuration Guide for downgrading guidelines. We modified the following commands: **enable, username**
Dual power supply support for the ISA 3000	9.6(1)	For dual power supplies in the ISA 3000, you can establish dual power supplies as the expected configuration in the ASA OS. If one power supply fails, the ASA issues an alarm. By default, the ASA expects a single power supply and won't issue an alarm as long as it includes one working power supply. We introduced the following command: **power-supply dual**
Longer password support for local **username** and **enable** passwords (up to 127 characters)	9.6(1)	You can now create local **username** and **enable** passwords up to 127 characters (the former limit was 32). When you create a password longer than 32 characters, it is stored in the configuration using a PBKDF2 (Password-Based Key Derivation Function 2) hash. Shorter passwords continue to use the MD5-based hashing method. We modified the following commands: **enable, username**
ISA 3000 hardware bypass	9.4(1.225)	The ISA 3000 supports a hardware bypass function to allow traffic to continue flowing through the appliance when there is a loss of power. We introduced the following commands: **hardware-bypass, hardware-bypass manual, hardware-bypass boot-delay, show hardware-bypass** *This feature is not available in Version 9.5(1).*

Feature Name	Platform Releases	Description
Automatic ASP Load Balancing	9.3(2)	You can now enable automatic switching on and off of the ASP load balancing feature. **Note** The automatic feature is not supported on the ASAv; only manual enabling and disabling is supported. We introduced the following command: **asp load-balance per-packet auto**.
Removal of the default Telnet password	9.0(2), 9.1(2)	To improve security for management access to the ASA, the default login password for Telnet was removed; you must manually set the password before you can log in using Telnet. **Note** The login password is only used for Telnet if you do not configure Telnet user authentication (the **aaa authentication telnet console** command). Previously, when you cleared the password, the ASA restored the default of "cisco." Now when you clear the password, the password is removed. The login password is also used for Telnet sessions from the switch to the ASASM (see the **session** command). For initial ASASM access, you must use the **service-module session** command, until you set a login password. We modified the following command: **password**
Password Encryption Visibility	8.4(1)	We modified the **show password encryption** command.
Master Passphrase	8.3(1)	We introduced this feature. The master passphrase allows you to securely store plain text passwords in encrypted format and provides a key that is used to universally encrypt or mask all passwords, without changing any functionality. We introduced the following commands: **key config-key password-encryption**, **password encryption aes**, **clear configure password encryption aes**, **show running-config password encryption aes, show password encryption**



CHAPTER 22

DHCP and DDNS Services

This chapter describes how to configure the DHCP server or DHCP relay as well as dynamic DNS (DDNS) update methods.

- About DHCP and DDNS services, on page 759
- Guidelines for DHCP and DDNS services, on page 762
- Configure the DHCP Server, on page 763
- Configure the DHCP Relay Agent, on page 769
- Configure Dynamic DNS, on page 772
- Monitoring DHCP and DDNS Services, on page 778
- History for DHCP and DDNS Services, on page 783

About DHCP and DDNS services

This topic describes the DHCP server, DHCP relay agent, and DDNS update.

DHCPv4 servers

A DHCPv4 server provides network configuration parameters, such as IP addresses, to DHCP clients.

- delivers configuration parameters directly to DHCP clients attached to ASA interfaces, and
- uses UDP port communication for client-server messaging.

DHCPv4 server communication details

The DHCP communication uses specific network protocols and port assignments:

- IPv4 DHCP client: Uses broadcast rather than multicast address to reach the server and listens for messages on UDP port 68
- DHCP server: Listens for messages on UDP port 67

DHCP options

DHCP provides a framework for passing configuration information to hosts on a TCP/IP network. The configuration parameters are carried in tagged items that are stored in the Options field of the DHCP message

and the data are also called options. Vendor information is also stored in Options, and all of the vendor information extensions can be used as DHCP options.

Common DHCP options

Cisco IP Phones download their configuration from a TFTP server. When a Cisco IP Phone starts, if it does not have both the IP address and TFTP server IP address preconfigured, it sends a request with option 150 or 66 to the DHCP server to obtain this information.

- DHCP option 150 provides the IP addresses of a list of TFTP servers.
- DHCP option 66 gives the IP address or the hostname of a single TFTP server.
- DHCP option 3 sets the default route.

A single request might include both options 150 and 66. In this case, the DHCP server provides values for both options in the response if they are already configured on the ASA.

You can use advanced DHCP options to provide DNS, WINS, and domain name parameters to DHCP clients. DHCP option 15 is used for the DNS domain suffix. You can also use the DHCP automatic configuration setting to obtain these values or define them manually. When you use more than one method to define this information, it is passed to DHCP clients in the following sequence:

1. Manually configured settings.
2. Advanced DHCP options settings.
3. DHCP automatic configuration settings.

Domain name configuration precedence

You can manually define the domain name that you want the DHCP clients to receive and then enable DHCP automatic configuration. Although DHCP automatic configuration discovers the domain together with the DNS and WINS servers, the manually defined domain name is passed to DHCP clients with the discovered DNS and WINS server names, because the domain name discovered by the DHCP automatic configuration process is superseded by the manually defined domain name.

DHCPv6 stateless server

A DHCPv6 stateless server is a DHCP IPv6 configuration that

- provides information such as DNS server or domain name when clients send Information Request (IR) packets,
- accepts IR packets but does not assign addresses to clients, and
- works in conjunction with StateLess Address Auto Configuration (SLAAC) and Prefix Delegation features.

DHCPv6 stateless server operation

For clients that use StateLess Address Auto Configuration (SLAAC) in conjunction with the Prefix Delegation feature (Enable the IPv6 Prefix Delegation Client, on page 686), you can configure the ASA to provide information such as the DNS server or domain name when they send Information Request (IR) packets to the

ASA. The ASA only accepts IR packets and does not assign addresses to the clients. You will configure the client to generate its own IPv6 address by enabling IPv6 autoconfiguration on the client. Enabling stateless autoconfiguration on a client configures IPv6 addresses based on prefixes received in Router Advertisement messages; in other words, based on the prefix that the ASA received using Prefix Delegation.

DHCP relay agents

A DHCP relay agent forwards DHCP requests received on an interface to one or more DHCP servers.

DHCP relay agent operation

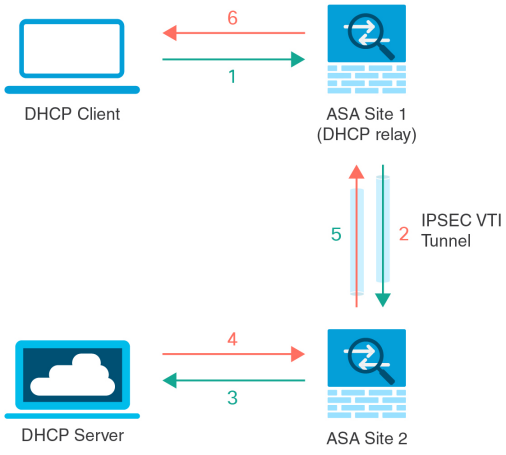
DHCP clients use UDP broadcasts to send their initial DHCPDISCOVER messages because they do not have information about the network to which they are attached. If the client is on a network segment that does not include a server, UDP broadcasts normally are not forwarded by the ASA because it does not forward broadcast traffic. The DHCP relay agent lets you configure the interface of the ASA that is receiving the broadcasts to forward DHCP requests to a DHCP server on another device.

DHCP Relay Server Support on VTI

You can configure DHCP relay agent on an ASA interface to receive and forward DHCP messages between a DHCP client and a DHCP server. However, a DHCP relay server to forward messages through a logical interface was not supported.

Following figure shows the DISCOVER process of the DHCP Client and DHCP Server using DHCP relay over VTI VPN. The DHCP relay agent, configured on VTI interface of ASA Site 1, receives DHCPDISCOVER packet from the DHCP Client and sends the packet through the VTI tunnel. ASA Site 2 forwards the DHCPDISCOVER packet to the DHCP Server. The DHCP Server replies with a DHCPOFFER to ASA Site 2. ASA Site 2 forwards it to DHCP relay (ASA Site1), which forwards it to the DHCP Client.

Figure 55: DHCP Relay Server over VTI



The same procedure is followed for a DHCPREQUEST and DHCPACK/NACK requirements.

Guidelines for DHCP and DDNS services

Context mode restrictions

DHCPv6 stateless server is not supported in multiple context mode.

Firewall mode restrictions

Consider these firewall mode limitations when configuring DHCP and DDNS services:

- DHCP Relay is not supported in transparent firewall mode or in routed mode on the BVI or bridge group member interface.
- DHCP Server is supported in transparent firewall mode on a bridge group member interface. In routed mode, the DHCP server is supported on the BVI interface, not the bridge group member interface. The BVI must have a name for the DHCP server to operate.
- DDNS is not supported in transparent firewall mode or in routed mode on the BVI or bridge group member interface.
- DHCPv6 stateless server is not supported in transparent firewall mode or in routed mode on the BVI or bridge group member interface.

Clustering limitations

DHCPv6 stateless server is not supported with clustering.

IPv6 support

DHCPv4 server guidelines

Follow these guidelines when configuring DHCPv4 server:

- The maximum available DHCP pool is 256 addresses.
- You can configure a DHCP server on any interface with a name and IP address, such as a physical interface, a subinterface, or a BVI in routed mode.
- Do not select an interface that uses DHCP or PPPoE to obtain it's address as a DHCP server interface.
- You can configure only one DHCP server on each interface. Each interface can have its own pool of addresses to use. However the other DHCP settings, such as DNS servers, domain name, options, ping timeout, and WINS servers, are configured globally and used by the DHCP server on all interfaces.
- You cannot configure an interface as a DHCP client if that interface also has DHCP server enabled; you must use a static IP address.
- You cannot configure both a DHCP server and DHCP relay on the same device, even if you want to enable them on different interfaces; you can only configure one type of service.
- You can reserve a DHCP address for an interface. The ASA assigns a specific address from the address pool to a DHCP client based on the client's MAC address.
- The ASA does not support DHCP clients behind DHCP relay servers; the client must be directly connected to the ASA.
- The DHCP server does not support BOOTP requests.

DHCPv6 server guidelines

Consider these guidelines when configuring DHCPv6 server:

- The DHCPv6 Stateless server cannot be configured on an interface where the DHCPv6 address, Prefix Delegation client, or DHCPv6 relay is configured.

DHCP relay guidelines

Follow these guidelines when configuring DHCP relay:

- You can configure a maximum of 10 DHCPv4 relay servers, with a maximum of 4 servers per interface.
- You can configure a maximum of 10 DHCPv6 relay servers. Interface-specific servers for IPv6 are not supported.
- You cannot configure both a DHCP server and DHCP relay on the same device, even if you want to enable them on different interfaces; you can only configure one type of service.
- Do not select an interface that uses DHCP or PPPoE to obtain it's address as a DHCP relay interface.
- DHCP relay services are not available in transparent firewall mode or in routed mode on the BVI or bridge group member interface. You can, however, allow DHCP traffic through using an access rule. To allow DHCP requests and replies through the ASA, you need to configure two access rules, one that allows DCHP requests from the inside interface to the outside (UDP destination port 67), and one that allows the replies from the server in the other direction (UDP destination port 68).
- For IPv4, clients must be directly-connected to the ASA and cannot send requests through another relay agent or a router. For IPv6, the ASA supports packets from another relay server.
- The DHCP clients must be on different interfaces from the DHCP servers to which the ASA relays requests.
- You cannot enable DHCP Relay on an interface in a traffic zone.

DDNS service configuration

The firewall's DDNS supports only DynDNS service. Hence, ensure that the DDNS is configured with update URL in the following syntax:

https://*username***:***password***@***provider-domain***/***path***?hostname=<h>&myip=<a>**.

Configure the DHCP Server

This section describes how to configure a DHCP server provided by the ASA.

Procedure

Step 1 Enable the DHCPv4 Server, on page 764.

Step 2 Configure Advanced DHCPv4 Options, on page 766.

Step 3 Configure the DHCPv6 Stateless Server, on page 767.

Enable the DHCPv4 Server

To enable the DHCP server on an ASA interface, perform the following steps:

Procedure

Step 1 Create a DHCP address pool for an interface. The ASA assigns a client one of the addresses from this pool to use for a given period of time. These addresses are the local, untranslated addresses for the directly connected network.

dhcpd address *ip_address_start-ip_address_end if_name*

Example:

```
ciscoasa(config)# dhcpd address 10.0.1.101-10.0.1.110 inside
```

The address pool must be on the same subnet as the ASA interface. In transparent mode, specify a bridge group member interface. In routed mode, specify a routed interface or a BVI; do not specify the bridge group member interface.

Step 2 (Optional) (Routed mode) Automatically configure DNS, WINS, and domain name values obtained from an interface running a DHCP or PPPoE client, or from a VPN server.

dhcpd auto_config *client_if_name* [[**vpnclient-wins-override**] **interface** *if_name*]

Example:

```
ciscoasa(config)# dhcpd auto_config outside interface inside
```

If you specify DNS, WINS, or domain name parameters using the following commands, then they overwrite the parameters obtained by automatic configuration.

Step 3 (Optional) Reserve a DHCP address for a client. The ASA assigns a specific address from the configured address pool to a DHCP client based on the client's MAC address.

dhcpd reserve-address *ip_address mac_address if_name*

Example:

```
ciscoasa(config)# dhcpd reserve-address 10.0.1.109 030c.f142.4cde inside
```

The reserved address must come from the configured address pool, and the address pool must be on the same subnet as the ASA interface. In transparent mode, specify a bridge group member interface. In routed mode, specify a routed interface or a BVI; do not specify the bridge group member interface.

Step 4 (Optional) Specify the IP address(es) of the DNS server(s).

dhcpd dns *dns1* [*dns2*]

Example:

```
ciscoasa(config)# dhcpd dns 209.165.201.2 209.165.202.129
```

Step 5 (Optional) Specify the IP address(es) of the WINS server(s). You may specify up to two WINS servers.

dhcpd wins *wins1* [*wins2*]

Example:

```
ciscoasa(config)# dhcpd wins 209.165.201.5
```

Step 6 (Optional) Change the lease length to be granted to the client. The lease length equals the amount of time in seconds that the client can use its allocated IP address before the lease expires. Enter a value from 0 to 1,048,575. The default value is 3600 seconds.

dhcpd lease *lease_length*

Example:

```
ciscoasa(config)# dhcpd lease 3000
```

Step 7 (Optional) Configure the domain name.

dhcpd domain *domain_name*

Example:

```
ciscoasa(config)# dhcpd domain example.com
```

Step 8 (Optional) Configure the DHCP ping timeout value for ICMP packets. To avoid address conflicts, the ASA sends two ICMP ping packets to an address before assigning that address to a DHCP client. The default is 50 milliseconds.

dhcpd ping timeout *milliseconds*

Example:

```
ciscoasa(config)# dhcpd ping timeout 20
```

Step 9 Define a default gateway that is sent to the DHCP clients. For routed mode, if you do not use the **dhcpd option 3 ip** command, then the ASA sends the DHCP server-enabled interface IP address as the default gateway. For transparent mode, you must set **dhcpd option 3 ip** if you want to set a default gateway; the ASA itself cannot act as the default gateway.

dhcpd option 3 ip *gateway_ip*

Example:

```
ciscoasa(config)# dhcpd option 3 ip 10.10.1.1
```

Step 10 Enable the DHCP daemon within the ASA to listen for DHCP client requests on the enabled interface.

dhcpd enable *interface_name*

Example:

```
ciscoasa(config)# dhcpd enable inside
```

Specify the same interface as the **dhcpd address** range.

Configure Advanced DHCPv4 Options

The ASA supports the DHCP options listed in RFC 2132, RFC 2562, and RFC 5510 to send information. All DHCP options (1 through 255) are supported except for 1, 12, 50–54, 58–59, 61, 67, and 82.

Procedure

Step 1 Configure a DHCP option that returns one or two IP addresses:

dhcpd option *code* **ip** *addr_1* [*addr_2*]

Example:

```
ciscoasa(config)# dhcpd option 150 ip 10.10.1.1
ciscoasa(config)# dhcpd option 3 ip 10.10.1.10
```

Option 150 provides the IP address or names of one or two TFTP servers for use with Cisco IP phones. Option 3 sets the default route for Cisco IP phones.

Step 2 Configure a DHCP option that returns a text string:

dhcpd option *code* **ascii** *text*

Example:

```
ciscoasa(config)# dhcpd option 66 ascii exampleserver
```

Option 66 provides the IP address or name of a TFTP server for use with Cisco IP phones.

Step 3 Configure a DHCP option that returns a hexadecimal value.

dhcpd option code hex *value*

Example:

```
ciscoasa(config)# dhcpd option 2 hex 22.0011.01.FF1111.00FF.0000.AAAA.1111.1111.1111.11
```

Note
The ASA does not verify that the option type and value that you provide match the expected type and value for the option code as defined in RFC 2132. For example, you can enter the **dhcpd option 46 ascii hello** command, and the ASA accepts the configuration, although option 46 is defined in RFC 2132 to expect a single-digit, hexadecimal value. For more information about option codes and their associated types and expected values, see RFC 2132.

The following table shows the DHCP options that are not supported by the **dhcpd option** command.

Table 29: Unsupported DHCP Options

Option Code	Description
0	DHCPOPT_PAD
1	HCPOPT_SUBNET_MASK
12	DHCPOPT_HOST_NAME
50	DHCPOPT_REQUESTED_ADDRESS
51	DHCPOPT_LEASE_TIME
52	DHCPOPT_OPTION_OVERLOAD
53	DHCPOPT_MESSAGE_TYPE
54	DHCPOPT_SERVER_IDENTIFIER
58	DHCPOPT_RENEWAL_TIME
59	DHCPOPT_REBINDING_TIME
61	DHCPOPT_CLIENT_IDENTIFIER
67	DHCPOPT_BOOT_FILE_NAME
82	DHCPOPT_RELAY_INFORMATION
255	DHCPOPT_END

Configure the DHCPv6 Stateless Server

For clients that use StateLess Address Auto Configuration (SLAAC) in conjunction with the Prefix Delegation feature (Enable the IPv6 Prefix Delegation Client, on page 686), you can configure the ASA to provide information such as the DNS server or domain name when they send Information Request (IR) packets to the ASA. The ASA only accepts IR packets and does not assign addresses to the clients. You will configure the client to generate its own IPv6 address by enabling IPv6 autoconfiguration on the client. Enabling stateless autoconfiguration on a client configures IPv6 addresses based on prefixes received in Router Advertisement messages; in other words, based on the prefix that the ASA received using Prefix Delegation.

Before you begin

This feature is only supported in single, routed mode. This feature is not supported in clustering.

Procedure

Step 1 Configure the IPv6 DHCP pool that contains the information you want the DHCPv6 server to provide:

ipv6 dhcp pool *pool_name*

Example:

```
ciscoasa(config)# ipv6 dhcp pool Inside-Pool
ciscoasa(config)#
```

You can configure separate pools for each interface if you want, or you can use the same pool on multiple interfaces.

Step 2 Configure one or more of the following parameters to be provided to clients in responses to IR messages:

dns-server *dns_ipv6_address*

domain-name *domain_name*

nis address *nis_ipv6_address*

nis domain-name *nis_domain_name*

nisp address *nisp_ipv6_address*

nisp domain-name *nisp_domain_name*

sip address *sip_ipv6_address*

sip domain-name *sip_domain_name*

sntp address *sntp_ipv6_address*

import {[**dns-server**] [**domain-name**] [**nis address**] [**nis domain-name**] [**nisp address**] [**nisp domain-name**] [**sip address**] [**sip domain-name**] [**sntp address**]}

Example:

```
ciscoasa(config-dhcpv6)# domain-name example.com
ciscoasa(config-dhcpv6)# import dns-server
```

The **import** command uses one or more parameters that the ASA obtained from the DHCPv6 server on the Prefix Delegation client interface. You can mix and match manually-configured parameters with imported parameters; however, you cannot configure the same parameter manually and in the **import** command.

Step 3 Enter interface configuration mode for the interface where you want the ASA to listen for IR messages:

interface *id*

Example:

```
ciscoasa(config)# interface gigabithethernet 0/0
ciscoasa(config-if)#
```

Step 4 Enable the DHCPv6 server:

ipv6 dhcp server *pool_name*

Example:

```
ciscoasa(config-if)# ipv6 dhcp server Inside-Pool
ciscoasa(config-if)#
```

Step 5 Configure the Router Advertisement to inform SLAAC clients about the DHCPv6 server:

ipv6 nd other-config-flag

This flag informs IPv6 autoconfiguration clients that they should use DHCPv6 to obtain additional information from DHCPv6, such as the DNS server address.

Example

The following example creates two IPv6 DHCP pools, and enables the DHCPv6 server on two interfaces:

```
ipv6 dhcp pool Eng-Pool
  domain-name eng.example.com
  import dns-server
ipv6 dhcp pool IT-Pool
  domain-name it.example.com
  import dns-server
interface gigabitethernet 0/0
  ipv6 address dhcp setroute default
  ipv6 dhcp client pd Outside-Prefix
interface gigabitethernet 0/1
  ipv6 address Outside-Prefix ::1:0:0:0:1/64
  ipv6 dhcp server Eng-Pool
  ipv6 nd other-config-flag
interface gigabitethernet 0/2
  ipv6 address Outside-Prefix ::2:0:0:0:1/64
  ipv6 dhcp server IT-Pool
  ipv6 nd other-config-flag
```

Configure the DHCP Relay Agent

When a DHCP request enters an interface, the DHCP servers to which the ASA relays the request depends on your configuration. You may configure the following types of servers:

- Interface-specific DHCP servers—When a DHCP request enters a particular interface, then the ASA relays the request only to the interface-specific servers.
- Global DHCP servers—When a DHCP request enters an interface that does not have interface-specific servers configured, the ASA relays the request to all global servers. If the interface has interface-specific servers, then the global servers are not used.

Configure the DHCPv4 Relay Agent

When a DHCP request enters an interface, the ASA relays the request to the DHCP server.

Procedure

Step 1 Do one or both of the following:

- Specify a global DHCP server IP address and the interface through which it is reachable.

  dhcprelay server *ip_address if_name*

  Example:

```
ciscoasa(config)# dhcprelay server 209.165.201.5 outside
ciscoasa(config)# dhcprelay server 209.165.201.8 outside
ciscoasa(config)# dhcprelay server 209.165.202.150 it
```

- Specify the interface ID connected to the DHCP client network, and the DHCP server IP address to be used for DHCP requests that enter that interface.

```
interface interface_id
  dhcprelay server ip_address
```

  Example:

```
ciscoasa(config)# interface gigabitethernet 0/0
ciscoasa(config)# dhcprelay server 209.165.201.6
ciscoasa(config)# dhcprelay server 209.165.201.7
ciscoasa(config)# interface gigabitethernet 0/1
ciscoasa(config)# dhcprelay server 209.165.202.155
ciscoasa(config)# dhcprelay server 209.165.202.156
```

  Note that you do not specify the egress interface for the requests, as in the global **dhcprelay server** command; instead, the ASA uses the routing table to determine the egress interface.

Step 2 Enable the DHCP relay service on the interface connected to the DHCP clients. You can enable DHCP relay on multiple interfaces.

dhcprelay enable *interface*

Example:

```
ciscoasa(config)# dhcprelay enable inside
ciscoasa(config)# dhcprelay enable dmz
ciscoasa(config)# dhcprelay enable eng1
ciscoasa(config)# dhcprelay enable eng2
ciscoasa(config)# dhcprelay enable mktg
```

Step 3 (Optional) Set the number of seconds allowed for DHCP relay address handling.

dhcprelay timeout *seconds*

Example:

```
ciscoasa(config)# dhcprelay timeout 25
```

Step 4 (Optional) Change the first default router address in the packet sent from the DHCP server to the address of the ASA interface.

dhcprelay setroute *interface_name*

Example:

```
ciscoasa(config)# dhcprelay setroute inside
```

This action allows the client to set its default route to point to the ASA even if the DHCP server specifies a different router.

If there is no default router option in the packet, the ASA adds one containing the interface address.

Step 5 (Optional) Configure interfaces as trusted interfaces. Do one of the following:

- Specify a DHCP client interface that you want to trust:

  ```
  interface interface_id
    dhcprelay information trusted
  ```

  Example:

  ```
  ciscoasa(config)# interface gigabitethernet 0/0
  ciscoasa(config-if)# dhcprelay information trusted
  ```

  You can configure interfaces as trusted interfaces to preserve DHCP Option 82. DHCP Option 82 is used by downstream switches and routers for DHCP snooping and IP Source Guard. Normally, if the ASA DHCP relay agent receives a DHCP packet with Option 82 already set, but the giaddr field (which specifies the DHCP relay agent address that is set by the relay agent before it forwards the packet to the server) is set to 0, then the ASA will drop that packet by default. You can now preserve Option 82 and forward the packet by identifying an interface as a trusted interface.

- Configure all client interfaces as trusted:

  dhcprelay information trust-all

  Example:

  ```
  ciscoasa(config)# dhcprelay information trust-all
  ```

Configure the DHCPv6 Relay Agent

When a DHCPv6 request enters an interface, the ASA relays the request to all DHCPv6 global servers.

Procedure

Step 1 Specify the IPv6 DHCP server destination address to which client messages are forwarded.

ipv6 dhcprelay server *ipv6_address* [*interface*]

Example:

```
ciscoasa(config)# ipv6 dhcprelay server 3FFB:C00:C18:6:A8BB:CCFF:FE03:2701
```

The *ipv6-address* argument can be a link-scoped unicast, multicast, site-scoped unicast, or global IPv6 address. Unspecified, loopback, and node-local multicast addresses are not allowed as the relay destination. The optional *interface* argument specifies the egress interface for a destination. Client messages are forwarded to the destination address through the link to which the egress interface is connected. If the specified address is a link-scoped address, then you must specify the interface.

Step 2 Enable DHCPv6 relay service on an interface.

ipv6 dhcprelay enable *interface*

Example:

```
ciscoasa(config)# ipv6 dhcprelay enable inside
```

Step 3 (Optional) Specify the amount of time in seconds that is allowed for responses from the DHCPv6 server to pass to the DHCPv6 client through the relay binding for relay address handling.

ipv6 dhcprelay timeout *seconds*

Example:

```
ciscoasa(config)# ipv6 dhcprelay timeout 25
```

Valid values for the *seconds* argument range from 1 to 3600. The default is 60 seconds.

Configure Dynamic DNS

When an interface uses DHCP IP addressing, the assigned IP address can change when the DHCP lease is renewed. When the interface needs to be reachable using a fully qualified domain name (FQDN), the IP address change can cause the DNS server resource records (RRs) to become stale. Dynamic DNS (DDNS) provides a mechanism to update DNS RRs whenever the IP address or hostname changes. You can also use DDNS for static or PPPoE IP addressing.

DDNS updates the following RRs on the DNS server: the A RR includes the name-to-IP address mapping, while the PTR RR maps addresses to names.

The ASA supports the following DDNS update methods:

- Standard DDNS—The standard DDNS update method is defined by RFC 2136.

With this method, the ASA and the DHCP server use DNS requests to update the DNS RRs. The ASA or DHCP server sends a DNS request to its local DNS server for information about the hostname and, based on the response, determines the main DNS server that owns the RRs. The ASA or DHCP server then sends an update request directly to the main DNS server. See the following typical scenarios.

- The ASA updates the A RR, and the DHCP server updates the PTR RR.

  Typically, the ASA "owns" the A RR, while the DHCP server "owns" the PTR RR, so both entities need to request updates separately. When the IP address or hostname changes, the ASA sends a DHCP request (including the FQDN option) to the DHCP server to inform it that it needs to request a PTR RR update.

- The DHCP server updates both the A and PTR RR.

  Use this scenario if the ASA does not have the authority to update the A RR. When the IP address or hostname changes, the ASA sends a DHCP request (including the FQDN option) to the DHCP server to inform it that it needs to request an A and PTR RR update.

You can configure different ownership depending on your security needs and the requirements of the main DNS server. For example, for a static address, the ASA should own the updates for both records.

- Web—The Web update method uses the DynDNS Remote API specification (https://help.dyn.com/remote-access-api/).

  With this method when the IP address or hostname changes, the ASA sends an HTTP request directly to a DNS provider with which you have an account.

Note DDNS is not supported on the BVI or bridge group member interfaces.

Before you begin

- Configure a DNS server on **Configuration** > **Device Management** > **DNS** > **DNS Client**. See Configure the DNS Server, on page 748.
- Configure the device hostname and domain name on **Configuration** > **Device Setup** > **Device Name/Password**. See Set the Hostname, Domain Name, and the Enable and Telnet Passwords, on page 735. If you do not specify the hostname per interface, then the device hostname is used. If you do not specify an FQDN, then for static or PPPoE IP addressing, the system domain name or the DNS server domain name is appended to the hostname.

Procedure

Step 1 Standard DDNS method: Configure a DDNS update method to enable DNS requests from the ASA.

You do not need to configure a DDNS update method if the DHCP server will perform all requests.

a) Create an update method.

ddns update method *name*

Example:

```
ciscoasa(config)# ddns update method ddns1
ciscoasa(DDNS-update-method)#
```

b) Specify the standard DDNS method.

ddns [**both**]

By default, the ASA updates the A RR only. Use this setting if you want the DHCP server to update the PTR RR. If you want the ASA to update both the A and PTR RR, specify **both**. Use the **both** keyword for static or PPPoE IP addressing.

Example:

```
ciscoasa(DDNS-update-method)# ddns
```

c) (Optional) Configure the update interface between DNS requests.

interval maximum *days hours minutes seconds*

By default when all values are set to 0, update requests are sent whenever the IP address or hostname changes. To send requests regularly, set the *days* (0-364), *hours*, *minutes*, and *seconds*.

Example:

```
ciscoasa(DDNS-update-method)# interval maximum 0 0 15 0
```

d) Associate this method with an interface. See Step 3, on page 775.

Step 2 Web method: Configure a DDNS update method to enable HTTP update requests from the ASA.

a) Create an update method.

ddns update method *name*

Example:

```
ciscoasa(config)# ddns update method web1
ciscoasa(DDNS-update-method)#
```

b) Specify the web method and the update URL.

web update-url https://*username***:***password***@***provider-domain***/***path***?hostname=<h>&myip=<a>**

Before entering the question mark (?) character, press the control (Ctrl) key and the v key together on your keyboard. This will allow you to enter the ? without the software interpreting the ? as a help query.

Example:

```
ciscoasa(DDNS-update-method)#
web update-url
https://jcrichton:pa$$w0rd17@domains.example.com/nic/update?hostname=<h>&myip=<a>
```

c) (Optional) Specify the address types (IPv4 or IPv6) that you want to update.

By default, the ASA updates all IPv4 and IPv6 addresses. If you want to limit the addresses, enter the following command.

web update-type {**ipv4** | **ipv6** [**all**] | **both** [**all**]}

- **both all**—(Default) Updates all IPv4 and IPv6 addresses.
- **both**—Updates the IPv4 address and the latest IPv6 address.
- **ipv4**—Updates only the IPv4 address.
- **ipv6**—Updates only the latest IPv6 address.
- **ipv6 all**—Updates all IPv6 addresses.

Example:

```
ciscoasa(DDNS-update-method)# web update-type ipv4
```

d) (Optional) Configure the update interface between DNS requests.

interval maximum *days hours minutes seconds*

By default when all values are set to 0, update requests are sent whenever the IP address or hostname changes. To send requests regularly, set the *days* (0-364), *hours*, *minutes*, and *seconds*.

Example:

```
ciscoasa(DDNS-update-method)# interval maximum 0 0 15 0
```

e) Associate this method with an interface. See Step 3, on page 775.

f) The web type method for DDNS also requires you to identify the DDNS server root CA to validate the DDNS server certificate for the HTTPS connection. See step Step 4, on page 776.

Step 3 Configure interface settings for DDNS, including setting the update method, DHCP client settings, and the hostname for this interface.

a) Enter interface configuration mode.

interface *id*

Example:

```
ciscoasa(config)# interface gigabitethernet1/1
ciscoasa(config-if)#
```

b) Assign an update method.

ddns update *name*

Standard DDNS method: You do not need to assign a method if you want the DHCP server to perform all updates. This command is required for the web update method.

Example:

```
ciscoasa(config-if)# ddns update ddns1
```

c) Assign a hostname for this interface.

ddns update hostname *hostname*

If you do not set the hostname, the device hostname is used. If you do not specify an FQDN, then the system domain name or the default domain from the DNS server group is appended (for static or PPPoE IP addressing) or the domain name from the DHCP server is appended (for DHCP IP addressing).

Example:

```
ciscoasa(config-if)# ddns update hostname asa1.example.com
```

d) Standard DDNS method: Determine which records you want the DHCP server to update.

dhcp client update dns [**server** {**both** | **none**}]

The ASA sends DHCP client requests to the DHCP server. Note that the DHCP server must also be configured to support DDNS. The server can be configured to honor the client requests, or it can override the client (in which case, it will reply to the client so the client does not also try to perform updates that the server is performing). Even if the client does not request DDNS updates, the DHCP server can be configured to send updates anyway.

For static or PPPoE IP addressing, these settings are ignored.

Note
You can also set these values globally for all interfaces using the **dhcp-client update dns** command. The per-interface settings take precedence over the global settings.

- Default (no keywords)—Requests that the DHCP server perform the PTR RR update. This setting works in conjunction with a DDNS update method with **ddns** A Records enabled.
- **server both**—Requests that the DHCP server perform both A and PTR RR updates. This setting does not require a DDNS update method to be associated with the interface.
- **server none**—Requests the DHCP server not to perform updates. This setting works in conjunction with a DDNS update method with **ddns both** A and PTR records enabled.

Example:

```
ciscoasa(config-if)# ddns client update dns
```

Step 4 The web method for DDNS also requires you to identify the DDNS server root CA to validate the DDNS server certificate for the HTTPS connection. See Configure Trustpoints, on page 799.

Example:

```
crypto ca trustpoint DDNS_Trustpoint
  enrollment terminal
crypto ca authenticate DDNS_Trustpoint nointeractive
   MIIFWjCCA0KgAwIBAgIQbkepxUtHDA3sM9CJuRz04TANBgkqhkiG9w0BAQwFADBH
   MQswCQYDVQQGEwJVUzEiMCAGA1UEChMZR29vZ2xlIFRydXN0IFNlcnZpY2VzIExM
   [...]
   quit
```

Standard DDNS Method for a Static IP Address

The following example shows how to configure the standard DDNS method for use with a static IP address. Note that you do not configure DHCP client settings for this scenario.

```
! Define the DDNS method to update both RRs:
ddns update method ddns-2
 ddns both
interface gigabitethernet1/1
 ip address 209.165.200.225
! Associate the method with the interface:
 ddns update ddns-2
 ddns update hostname asa1.example.com
```

Example: Standard DDNS Method;ASA Updates A RR and DHCP Server Updates PTR RR

The following example configures the ASA to update the A RR and the DHCP server to update the PTR RR.

```
! Define the DDNS method to update the A RR:
ddns update method ddns-1
 ddns
interface gigabitethernet1/1
  ip address dhcp
! Associate the method with the interface:
  ddns update ddns-1
  ddns update hostname asa
! Set the client to update the A RR, and the server to update the PTR RR:
  dhcp client update dns
```

Example: Standard DDNS Method; No DHCP Server Update of RRs

The following example configures the ASA to update both the A and PTR RR, while requesting the DHCP server to update no RRs.

```
! Define the DDNS method to update both RRs:
ddns update method ddns-2
  ddns both
! Associate the method with the interface:
interface gigabitethernet1/1
  ip address dhcp
  ddns update ddns-2
  ddns update hostname asa1.example.com
! Set the client to update both RRs, and the server to update none:
  dhcp client update dns server none
```

Example: Standard DDNS Method; DHCP Server Updates all RRs

The following example configures the DHCP client to request that the DHCP server to update both the A and PTR RRs. Because the server performs all updates, you do not need to associated an update method with the interface.

```
interface gigabitethernet1/1
  ip address dhcp
  ddns update hostname asa
! Configure the DHCP server to update both RRs:
  dhcp client update dns server both
```

Example: Web Type

The following example configures the web type method.

```
! Define the web type method:
ddns update method web-1
  web update-url https://captainkirk:enterpr1s3@domains.cisco.com/ddns?hostname=<h>&myip=<a>
! Associate the method with the interface:
interface gigabitethernet1/1
  ip address dhcp
  ddns update web-1
  ddns update hostname asa2.example.com
```

Monitoring DHCP and DDNS Services

This section includes the procedures to monitor both DHCP and DDNS services.

Monitoring DHCP Services

- **show dhcpd** {**binding** [*IP_address*] | **state** | **statistics**}

  This command shows the current DHCP server client binding, state, and statistics.

- **show dhcprelay** {**state** | **statistics**}

  This command displays the DHCP relay status and statistics.

- **show ipv6 dhcprelay binding**

  This command shows the relay binding entries that were created by the relay agent.

- **show ipv6 dhcprelay statistics**

  This command shows DHCP relay agent statistics for IPv6.

- **show ipv6 dhcp server statistics**

  This command shows the DHCPv6 stateless server statistics. The following example shows information provided by this command:

```
ciscoasa(config)# show ipv6 dhcp server statistics

Protocol Exchange Statistics:
  Total number of Solicit messages received:             0
  Total number of Advertise messages sent:               0
  Total number of Request messages received:             0
  Total number of Renew messages received:               0
  Total number of Rebind messages received:              0
  Total number of Reply messages sent:                  10
  Total number of Release messages received:             0
  Total number of Reconfigure messages sent:             0
  Total number of Information-request messages received: 10
  Total number of Relay-Forward messages received:       0
  Total number of Relay-Reply messages sent:             0

Error and Failure Statistics:
  Total number of Re-transmission messages sent:                 0
```

```
  Total number of Message Validation errors in received messages: 0
```

- **show ipv6 dhcp pool** [*pool_name*]
- **show ipv6 dhcp interface** [*ifc_name* [**statistics**]]

  The **show ipv6 dhcp interface** command displays DHCPv6 information for all interfaces. If the interface is configured for DHCPv6 stateless server configuration (see Configure the DHCPv6 Stateless Server, on page 767), this command lists the DHCPv6 pool that is being used by the server. If the interface has DHCPv6 address client or Prefix Delegation client configuration, this command shows the state of each client and the values received from the server. For a specific interface, you can show message statistics for the DHCP server or client. The following examples show information provided by this command:

```
ciscoasa(config-if)# show ipv6 dhcp interface
GigabitEthernet1/1 is in server mode
  Using pool: Sample-Pool

GigabitEthernet1/2 is in client mode
  Prefix State is OPEN
  Renew will be sent in 00:03:46
  Address State is OPEN
  Renew for address will be sent in 00:03:47
  List of known servers:
    Reachable via address: fe80::20c:29ff:fe96:1bf4
    DUID: 000100011D9D1712005056A07E06
    Preference: 0
    Configuration parameters:
      IA PD: IA ID 0x00030001, T1 250, T2 400
        Prefix: 2005:abcd:ab03::/48
                preferred lifetime 500, valid lifetime 600
                expires at Nov 26 2014 03:11 PM (577 seconds)
      IA NA: IA ID 0x00030001, T1 250, T2 400
        Address: 2004:abcd:abcd:abcd:abcd:abcd:abcd:f2cb/128
                preferred lifetime 500, valid lifetime 600
                expires at Nov 26 2014 03:11 PM (577 seconds)
      DNS server: 2004:abcd:abcd:abcd::2
      DNS server: 2004:abcd:abcd:abcd::4
      Domain name: relay.com
      Domain name: server.com
      Information refresh time: 0
  Prefix name: Sample-PD

Management1/1 is in client mode
  Prefix State is IDLE
  Address State is OPEN
  Renew for address will be sent in 11:26:44
  List of known servers:
    Reachable via address: fe80::4e00:82ff:fe6f:f6f9
    DUID: 000300014C00826FF6F8
    Preference: 0
    Configuration parameters:
      IA NA: IA ID 0x000a0001, T1 43200, T2 69120
        Address: 2308:2308:210:1812:2504:1234:abcd:8e5a/128
                preferred lifetime INFINITY, valid lifetime INFINITY
      Information refresh time: 0


ciscoasa(config-if)# show ipv6 dhcp interface outside statistics

DHCPV6 Client PD statistics:
```

```
Protocol Exchange Statistics:


 Number of Solicit messages sent:              1
 Number of Advertise messages received:        1
 Number of Request messages sent:              1
 Number of Renew messages sent:                45
 Number of Rebind messages sent:               0
 Number of Reply messages received:            46
 Number of Release messages sent:              0
 Number of Reconfigure messages received:      0
 Number of Information-request messages sent:  0

Error and Failure Statistics:


 Number of Re-transmission messages sent:                 1
 Number of Message Validation errors in received messages: 0

DHCPV6 Client address statistics:

Protocol Exchange Statistics:


 Number of Solicit messages sent:              1
 Number of Advertise messages received:        1
 Number of Request messages sent:              1
 Number of Renew messages sent:                45
 Number of Rebind messages sent:               0
 Number of Reply messages received:            46
 Number of Release messages sent:              0
 Number of Reconfigure messages received:      0
 Number of Information-request messages sent:  0

Error and Failure Statistics:


 Number of Re-transmission messages sent:                 1
 Number of Message Validation errors in received messages: 0
```

- **show ipv6 dhcp ha statistics**

  The **show ipv6 dhcp ha statistics** command shows the transaction statistics between failover units, including how many times the DUID information was synced between the units. The following examples show information provided by this command.

  On an active unit:

```
ciscoasa(config)# show ipv6 dhcp ha statistics

DHCPv6 HA global statistics:
  DUID sync messages sent:              1
  DUID sync messages received:          0

DHCPv6 HA error statistics:
  Send errors:                          0
```

  On an standby unit:

```
ciscoasa(config)# show ipv6 dhcp ha statistics
```

```
DHCPv6 HA global statistics:
  DUID sync messages sent:              0
  DUID sync messages received:          1

DHCPv6 HA error statistics:
  Send errors:                          0
```

Troubleshooting DHCP Relay over VTI

If the DHCP client fails to get an IP address:

- Verify for Tunnel interface/VTI configuration in both the ASA sites.
- Verify the packets transferred between the sites using the **show crypto ipsec sa** command:

  Example

```
ciscoasa(config)# show crypto ipsec sa
interface: outside
Crypto map tag: cmap, seq num: 10, local addr: 192.168.2.111
access-list CSM_IPSEC_ACL_0 extended permit ip any4 any4
local ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
remote ident (addr/mask/prot/port): (0.0.0.0/0.0.0.0/0/0)
current_peer: 192.168.2.110
#pkts encaps: 1, #pkts encrypt: 1, #pkts digest: 1
#pkts decaps: 2, #pkts decrypt: 2, #pkts verify: 2
```

Enable Debug Commands

Enabling DHCP relay debugs helps you to know whether the DISCOVER/REQUEST packets were forwarded to DHCP relay server:

- **debug dhcprelay event 255**
- **debug dhcprelay packet 255**
- **debug dhcprelay error 255**

Example

```
ciscoasa(config)# DHCPD/RA: Relay msg received, fip=ANY, fport=0 on inside interface
DHCP: Received a BOOTREQUEST from interface 2 (size = 548)
DHCPRA: relay binding found for client xxxx.xxxx.xxxx.
DHCPRA: setting giaddr to 192.168.1.111. dhcpd_forward_request: request from xxxx.xxxx.xxxx
forwarded to 192.168.3.112.
DHCPD/RA: Relay msg received, fip=ANY, fport=0 on vti interface
DHCP: Received a BOOTREPLY from relay interface 5 (size = 300, xid = xxxxxxxxx) at 04:40:52
UTC Tue Sep 10 2019
DHCPRA: relay binding found for client xxxx.xxxx.xxxx.
DHCPD/RA: creating ARP entry (192.168.1.88, xxxx.xxxx.xxxx).
DHCPRA: Adding rule to allow client to respond using offered address 192.168.1.95
DHCPRA: forwarding reply to client xxxx.xxxx.xxxx.
DHCPD/RA: Relay msg received, fip=ANY, fport=0 on inside interface
```

Monitoring DDNS Status

See the following command for monitoring DDNS status.

- **show ddns update** {**interface** *if_name* | **method** [*name*]}

  This command shows the DDNS update status.

  The following example show details about the DDNS update method:

```
ciscoasa# show ddns update method ddns1

Dynamic DNS Update Method: ddns1
  IETF standardized Dynamic DNS 'A' record update
```

  The following example shows details about the web update method:

```
ciscoasa# show ddns update method web1

Dynamic DNS Update Method: web1
 Dynamic DNS updated via HTTP(s) protocols
  URL used to update record:
https://cdarwin:*****@ddns.cisco.com/update?hostname=<h>&myip=<a>
```

  The following example shows information about the DDNS interface:

```
ciscoasa# show ddns update interface outside

Dynamic DNS Update on outside:
  Update Method Name            Update Destination
  test                          not available
```

  The following example shows a successful web type update:

```
ciscoasa# show ddns update interface outside

Dynamic DNS Update on outside:
  Update Method Name            Update Destination
  test                          not available

Last Update attempted on 09:01:52.729 UTC Mon Mar 23 2020
Status : Success
FQDN : asa1.example.com
IP addresses(s): 10.10.32.45,2001:DB8::1
```

  The following example shows a web type failure:

```
ciscoasa# show ddns update interface outside

Dynamic DNS Update on outside:
  Update Method Name            Update Destination
  test                          not available

Last Update attempted on 09:01:52.729 UTC Mon Mar 23 2020
Status : Failed
Reason : Could not establish a connection to the server
```

  The following example shows that the DNS server returned an error for the web type update:

```
ciscoasa# show ddns update interface outside

Dynamic DNS Update on outside:
  Update Method Name            Update Destination
```

```
  test                          not available

Last Update attempted on 09:01:52.729 UTC Mon Mar 23 2020
Status : Failed
Reason : Server error (Error response from server)
```

The following example shows that a web update was not yet attempted due to the IP address unconfigured or the DHCP request failed, for example:

```
ciscoasa# show ddns update interface outside

Dynamic DNS Update on outside:
  Update Method Name            Update Destination
  test                          not available

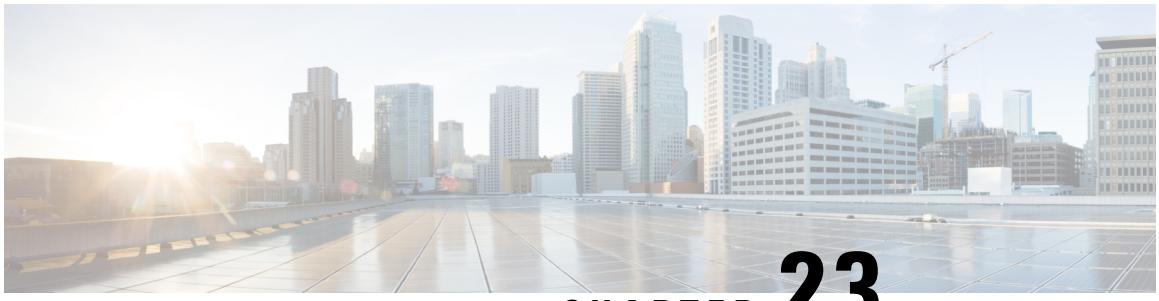
Last Update Not attempted
```

History for DHCP and DDNS Services

Feature Name	Platform Releases	Description
DDNS support for the web update method	9.15(1)	You can now configure an interface to use DDNS with the web update method. New/Modified commands: **show ddns update interface**, **show ddns update method**, **web update-url**, **web update-type**
DHCP relay server support on VTIs	9.14(1)	You can now enable DHCP relay on VTIs. New/Modified commands: **dhcprelay server** .
DHCP reservation	9.13(1)	ASA supports DHCP reservation. The DHCP server assigns a static IP address from the defined address pool to a DHCP client based on the client's MAC address. New/Modified commands: **dhcpd reserve-address** .

Feature Name	Platform Releases	Description
IPv6 DHCP	9.6(2)	The ASA now supports the following features for IPv6 addressing: • DHCPv6 Address client—The ASA obtains an IPv6 global address and optional default route from the DHCPv6 server. • DHCPv6 Prefix Delegation client—The ASA obtains delegated prefix(es) from a DHCPv6 server. The ASA can then use these prefixes to configure other ASA interface addresess so that StateLess Address Auto Configuration (SLAAC) clients can autoconfigure IPv6 addresses on the same network. • BGP router advertisement for delegated prefixes • DHCPv6 stateless server—The ASA provides other information such as the domain name to SLAAC clients when they send Information Request (IR) packets to the ASA. The ASA only accepts IR packets, and does not assign addresses to the clients. We added or modified the following commands: **clear ipv6 dhcp statistics, domain-name, dns-server, import, ipv6 address, ipv6 address dhcp, ipv6 dhcp client pd, ipv6 dhcp client pd hint, ipv6 dhcp pool, ipv6 dhcp server, network, nis address, nis domain-name, nisp address, nisp domain-name, show bgp ipv6 unicast, show ipv6 dhcp, show ipv6 general-prefix, sip address, sip domain-name, sntp address**
DHCPv6 monitoring	9.4(1)	You can now monitor DHCP statistics for IPv6 and DHCP bindings for IPv6.
DHCP Relay server validates the DHCP Server identifier for replies	9.2(4)/ 9.3(3)	If the ASA DHCP relay server receives a reply from an incorrect DHCP server, it now verifies that the reply is from the correct server before acting on the reply. We did not introduce or modify any commands. We did not modify any ASDM screens. We did not introduce or modify any commands.
DHCP rebind function	9.1(4)	During the DHCP rebind phase, the client now tries to rebind to other DHCP servers in the tunnel group list. Before this release, the client did not rebind to an alternate server when the DHCP lease fails to renew. We did not introduce or modify any commands.
DHCP trusted interfaces	9.1(2)	You can now configure interfaces as trusted interfaces to preserve DHCP Option 82. DHCP Option 82 is used by downstream switches and routers for DHCP snooping and IP Source Guard. Normally, if the ASA DHCP relay agent receives a DHCP packet with Option 82 already set, but the giaddr field (which specifies the DHCP relay agent address that is set by the relay agent before it forwards the packet to the server) is set to 0, then the ASA will drop that packet by default. You can now preserve Option 82 and forward the packet by identifying an interface as a trusted interface. We introduced or modified the following commands: **dhcprelay information trusted**, **dhcprelay information trust-all**, **show running-config dhcprelay**.
DHCP relay servers per interface (IPv4 only)	9.1(2)	You can now configure DHCP relay servers per-interface, so requests that enter a given interface are relayed only to servers specified for that interface. IPv6 is not supported for per-interface DHCP relay. We introduced or modified the following commands: **dhcprelay server** (interface config mode), **clear configure dhcprelay**, **show running-config dhcprelay**.

Feature Name	Platform Releases	Description
DHCP relay for IPv6 (DHCPv6)	9.0(1)	DHCP relay support for IPv6 was added. We introduced the following commands: **ipv6 dhcprelay server**, **ipv6 dhcprelay enable**, **ipv6 dhcprelay timeout**, **clear config ipv6 dhcprelay**, **ipv6 nd managed-config-flag**, **ipv6 nd other-config-flag**, **debug ipv6 dhcp**, **debug ipv6 dhcprelay**, **show ipv6 dhcprelay binding**, **clear ipv6 dhcprelay binding**, **show ipv6 dhcprelay statistics**, and **clear ipv6 dhcprelay statistics**.
DDNS	7.0(1)	We introduced this feature. We introduced the following commands: **ddns**, **ddns update**, **dhcp client update dns**, **dhcpd update dns**, **show running-config ddns,** and **show running-config dns server-group**.
DHCP	7.0(1)	The ASA can provide a DHCP server or DHCP relay services to DHCP clients attached to ASA interfaces. We introduced the following commands: **dhcp client update dns**, **dhcpd address**, **dhcpd domain**, **dhcpd enable**, **dhcpd lease**, **dhcpd option**, **dhcpd ping timeout**, **dhcpd update dns**, **dhcpd wins**, **dhcp-network-scope**, **dhcprelay enable**, **dhcprelay server**, **dhcprelay setroute**, **dhcp-server**. **show running-config dhcpd**, and **show running-config dhcprelay**.



CHAPTER 23

Digital Certificates

This chapter describes how to configure digital certificates.

- About Digital Certificates, on page 787
- Guidelines for Digital Certificates, on page 794
- Configure Digital Certificates, on page 797
- Set a Certificate Expiration Alert (for Identity or CA Certificates), on page 819
- Monitoring Digital Certificates, on page 820
- History for Certificate Management, on page 822

About Digital Certificates

Digital certificates provide digital identification for authentication. A digital certificate includes information that identifies a device or user, such as the name, serial number, company, department, or IP address. CAs are responsible for managing certificate requests and issuing digital certificates. CAs are trusted authorities that "sign" certificates to verify their authenticity, thereby guaranteeing the identity of the device or user.

A digital certificate also includes a copy of the public key for the user or device. A CA can be a trusted third party, such as VeriSign, or a private (in-house) CA that you establish within your organization. CAs issue digital certificates in the context of a PKI, which uses public-key or private-key encryption to ensure security.

For authentication using digital certificates, at least one identity certificate and its issuing CA certificate must exist on an ASA. This configuration allows multiple identities, roots, and certificate hierarchies. The ASA evaluates third-party certificates against CRLs, also called authority revocation lists, all the way from the identity certificate up the chain of subordinate certificate authorities.

Descriptions of several different types of available digital certificates follow:

- A CA certificate is used to sign other certificates. It is self-signed and called a root certificate. A certificate that is issued by another CA certificate is called a subordinate certificate.
- CAs also issue identity certificates, which are certificates for specific systems or hosts.
- Code-signer certificates are special certificates that are used to create digital signatures to sign code, with the signed code itself revealing the certificate origin.

In a hierarchy with a root and two intermediate CA certificates, a CRL validation for remote access fail on the headend running 9.13 or later when the ID certificate is signed by one intermediate CA, but the CRL is signed by another intermediate CA. The failure happens even if the headend trusts both the intermediates where both are signed by the same root.

Hence, each signer must maintain their own CRL. Each signer would then specify the location of the CRL in the url list of each certificate it signs. Alternatively, you can configure a url override in the trustpoint of each signer pointing to the correct CRL location.

The local CA integrates an independent certificate authority feature on the ASA, deploys certificates, and provides secure revocation checking of issued certificates. The local CA provides a secure, configurable, in-house authority for certificate authentication with user enrollment through a website login page.



Note CA certificates and identity certificates apply to both site-to-site VPN connections and remote access VPN connections. Procedures in this document refer to remote access VPN use in the ASDM GUI.



Tip For an example of a scenario that includes certificate configuration and load balancing, see the following URL: https://supportforums.cisco.com/docs/DOC-5964.

Public Key Cryptography

Digital signatures, enabled by public key cryptography, provide a way to authenticate devices and users. In public key cryptography, such as the RSA encryption system, each user has a key pair containing both a public and a private key. The keys act as complements, and anything encrypted with one of the keys can be decrypted with the other.

In simple terms, a signature is formed when data is encrypted with a private key. The signature is attached to the data and sent to the receiver. The receiver applies the public key of the sender to the data. If the signature sent with the data matches the result of applying the public key to the data, the validity of the message is established.

This process relies on the receiver having a copy of the public key of the sender and a high degree of certainty that this key belongs to the sender, not to someone pretending to be the sender.

Obtaining the public key of a sender is normally handled externally or through an operation performed at installation. For example, most web browsers are configured with the root certificates of several CAs by default. For VPN, the IKE protocol, a component of IPsec, can use digital signatures to authenticate peer devices before setting up security associations.

Certificate Scalability

Without digital certificates, you must manually configure each IPsec peer for each peer with which it communicates; as a result, each new peer that you add to a network would require a configuration change on each peer with which it needs to communicate securely.

When you use digital certificates, each peer is enrolled with a CA. When two peers try to communicate, they exchange certificates and digitally sign data to authenticate each other. When a new peer is added to the network, you enroll that peer with a CA and none of the other peers need modification. When the new peer attempts an IPsec connection, certificates are automatically exchanged and the peer can be authenticated.

With a CA, a peer authenticates itself to the remote peer by sending a certificate to the remote peer and performing some public key cryptography. Each peer sends its unique certificate, which was issued by the CA. This process works because each certificate encapsulates the public key for the associated peer, each

certificate is authenticated by the CA, and all participating peers recognize the CA as an authenticating authority. The process is called IKE with an RSA signature.

The peer can continue sending its certificate for multiple IPsec sessions, and to multiple IPsec peers, until the certificate expires. When its certificate expires, the peer administrator must obtain a new one from the CA.

CAs can also revoke certificates for peers that no longer participate in IPsec. Revoked certificates are not recognized as valid by other peers. Revoked certificates are listed in a CRL, which each peer may check before accepting a certificate from another peer.

Some CAs have an RA as part of their implementation. An RA is a server that acts as a proxy for the CA, so that CA functions can continue when the CA is unavailable.

Key Pairs

Key pairs are RSA or Elliptic Curve Signature Algorithm (ECDSA) keys, which have the following characteristics:

- RSA keys can be used for SSH or SSL.
- SCEP enrollment supports the certification of RSA keys.
- The maximum RSA key size is 4096, and the default is 2048.
- The maximum ECDSA key length is 521, and the default is 384.
- You can generate a general purpose RSA key pair, used for both signing and encryption, or you can generate separate RSA key pairs for each purpose. Separate signing and encryption keys help to reduce exposure of the keys, because SSL uses a key for encryption but not signing. However, IKE uses a key for signing but not encryption. By using separate keys for each, exposure of the keys is minimized.

Trustpoints

Trustpoints let you manage and track CAs and certificates. A trustpoint is a representation of a CA or identity pair. A trustpoint includes the identity of the CA, CA-specific configuration parameters, and an association with one, enrolled identity certificate.

After you have defined a trustpoint, you can reference it by name in commands requiring that you specify a CA. You can configure many trustpoints.



Note If the ASA has multiple trustpoints that share the same CA, only one of these trustpoints sharing the CA can be used to validate user certificates. To control which trustpoint sharing a CA is used for validation of user certificates issued by that CA, use the **support-user-cert-validation** command.

For automatic enrollment, a trustpoint must be configured with an enrollment URL, and the CA that the trustpoint represents must be available on the network and must support SCEP.

You can export and import the keypair and issued certificates associated with a trustpoint in PKCS12 format. This format is useful to manually duplicate a trustpoint configuration on a different ASA.

Certificate Enrollment

The ASA needs a CA certificate for each trustpoint and one or two certificates for itself, depending upon the configuration of the keys used by the trustpoint. If the trustpoint uses separate RSA keys for signing and encryption, the ASA needs two certificates, one for each purpose. In other key configurations, only one certificate is needed.

The ASA supports automatic enrollment with SCEP and with manual enrollment, which lets you paste a base-64-encoded certificate directly into the terminal. For site-to-site VPNs, you must enroll each ASA. For remote access VPNs, you must enroll each ASA and each remote access VPN client.

Proxy for SCEP Requests

The ASA can proxy SCEP requests between AnyConnect Client and a third-party CA. The CA only needs to be accessible to the ASA if it is acting as the proxy. For the ASA to provide this service, the user must authenticate using any of the methods supported by AAA before the ASA sends an enrollment request. You can also use host scan and dynamic access policies to enforce rules of eligibility to enroll.

The ASA supports this feature only with an AnyConnect Client SSL or IKEv2 VPN session. It supports all SCEP-compliant CAs, including Cisco IOS CS, Windows Server 2003 CA, and Windows Server 2008 CA.

Clientless (browser-based) access does not support SCEP proxy, although WebLaunch—clientless-initiated AnyConnect Client—does support it.

The ASA does not support polling for certificates.

The ASA supports load balancing for this feature.

Revocation Checking

When a certificate is issued, it is valid for a fixed period of time. Sometimes a CA revokes a certificate before this time period expires; for example, because of security concerns or a change of name or association. CAs periodically issue a signed list of revoked certificates. Enabling revocation checking forces the ASA to check that the CA has not revoked a certificate each time that it uses the certificate for authentication.

When you enable revocation checking, the ASA checks certificate revocation status during the PKI certificate validation process, which can use either CRL checking, OCSP, or both. OCSP is only used when the first method returns an error (for example, indicating that the server is unavailable).

With CRL checking, the ASA retrieves, parses, and caches CRLs, which provide a complete list of revoked (and unrevoked) certificates with their certificate serial numbers. The ASA evaluates certificates according to CRLs, also called authority revocation lists, from the identity certificate up the chain of subordinate certificate authorities.

OCSP offers a more scalable method of checking revocation status in that it localizes certificate status through a validation authority, which it queries for status of a specific certificate.

Supported CA Servers

The ASA supports the following CA servers:

Cisco IOS CS, ASA Local CA, and third-party X.509 compliant CA vendors including, but not limited to:

- Baltimore Technologies
- Entrust

- Digicert
- Geotrust
- GoDaddy
- iPlanet/Netscape
- Microsoft Certificate Services
- RSA Keon
- Thawte
- VeriSign

CRLs

CRLs provide the ASA with one way of determining whether a certificate that is within its valid time range has been revoked by the issuing CA. CRL configuration is part of configuration of a trustpoint.

You can configure the ASA to make CRL checks mandatory when authenticating a certificate by using the **revocation-check crl** command. You can also make the CRL check optional by using the **revocation-check crl none** command, which allows the certificate authentication to succeed when the CA is unavailable to provide updated CRL data.



Note The **revocation-check crl none**, which was removed in 9.13(1), was restored.

The ASA can retrieve CRLs from CAs using HTTP, SCEP, or LDAP. CRLs retrieved for each trustpoint are cached for a configurable amount of time for each trustpoint.



Note Though the CRL server responds with HTTP flag "Connection: Keep-alive" to indicate a persistent connection, ASA does not request support for persistent connection. Change the settings on the CRL server to respond with "Connection: Close" when the list is sent.

When the ASA has cached a CRL for longer than the amount of time it is configured to cache CRLs, the ASA considers the CRL too old to be reliable, or “stale.” The ASA tries to retrieve a newer version of the CRL the next time that a certificate authentication requires a check of the stale CRL.

You could receive a *revocation check* failure for a user connection/certificate if you exceed the CRL size limit of 16 MB.

The ASA caches CRLs for an amount of time determined by the following two factors:

- The number of minutes specified with the **cache-time** command. The default value is 60 minutes.
- The NextUpdate field in the CRLs retrieved, which may be absent from CRLs. You control whether the ASA requires and uses the NextUpdate field with the **enforcenextupdate** command.

The ASA uses these two factors in the following ways:

- If the NextUpdate field is not required, the ASA marks CRLs as stale after the length of time defined by the **cache-time** command.
- If the NextUpdate field is required, the ASA marks CRLs as stale at the sooner of the two times specified by the **cache-time** command and the NextUpdate field. For example, if the **cache-time** command is set to 100 minutes and the NextUpdate field specifies that the next update is 70 minutes away, the ASA marks CRLs as stale in 70 minutes.

If the ASA has insufficient memory to store all CRLs cached for a given trustpoint, it deletes the least recently used CRL to make room for a newly retrieved CRL. Large CRLs require significant computational overhead to parse them. Hence, for better performance, use many CRLs of smaller size rather than few large CRLs, or preferably, use OCSP.

See the following the cache sizes:

- Single context mode—128 MB
- Multiple context mode—16 MB per context

OCSP

OCSP provides the ASA with a way of determining whether a certificate that is within its valid time range has been revoked by the issuing CA. OCSP configuration is part of trustpoint configuration.

OCSP localizes certificate status on a validation authority (an OCSP server, also called the *responder*) which the ASA queries for the status of a specific certificate. This method provides better scalability and more up-to-date revocation status than does CRL checking, and helps organizations with large PKI installations deploy and expand secure networks.



Note The ASA allows a five-second time skew for OCSP responses.

You can configure the ASA to make OCSP checks mandatory when authenticating a certificate by using the **revocation-check ocsp** command. You can also make the OCSP check optional by using the **revocation-check ocsp none** command, which allows the certificate authentication to succeed when the validation authority is unavailable to provide updated OCSP data.



Note The **revocation-check ocsp none**, which was removed in 9.13(1), was restored.

OCSP provides three ways to define the OCSP server URL. The ASA uses these servers in the following order:

1. The OCSP URL defined in a match certificate override rule by using the **match certificate** command).
2. The OCSP URL configured by using the **ocsp url** command.
3. The AIA field of the client certificate.

Note To configure a trustpoint to validate a self-signed OCSP responder certificate, you import the self-signed responder certificate into its own trustpoint as a trusted CA certificate. Then you configure the **match certificate** command in the client certificate validating trustpoint to use the trustpoint that includes the self-signed OCSP responder certificate to validate the responder certificate. Use the same procedure for configuring validating responder certificates external to the validation path of the client certificate.

The OCSP server (responder) certificate usually signs the OCSP response. After receiving the response, the ASA tries to verify the responder certificate. The CA normally sets the lifetime of the OCSP responder certificate to a relatively short period to minimize the chance of being compromised. The CA usually also includes an ocsp-no-check extension in the responder certificate, which indicates that this certificate does not need revocation status checking. However, if this extension is not present, the ASA tries to check revocation status using the same method specified in the trustpoint. If the responder certificate is not verifiable, revocation checks fail. To avoid this possibility, use the **revocation-check none** command to configure the responder certificate validating trustpoint, and use the **revocation-check ocsp** command to configure the client certificate.

Certificates and User Login Credentials

The following section describes the different methods of using certificates and user login credentials (username and password) for authentication and authorization. These methods apply to IPsec, AnyConnect Client, and Clientless SSL VPN.

In all cases, LDAP authorization does not use the password as a credential. RADIUS authorization uses either a common password for all users or the username as a password.

User Login Credentials

The default method for authentication and authorization uses the user login credentials.

- Authentication
  - Enabled by the authentication server group setting in the tunnel group (also called ASDM Connection Profile)
  - Uses the username and password as credentials
- Authorization
  - Enabled by the authorization server group setting in the tunnel group (also called ASDM Connection Profile)
  - Uses the username as a credential

Certificates

If user digital certificates are configured, the ASA first validates the certificate. It does not, however, use any of the DNs from certificates as a username for the authentication.

If both authentication and authorization are enabled, the ASA uses the user login credentials for both user authentication and authorization.

- Authentication

- Enabled by the authentication server group setting
- Uses the username and password as credentials

- Authorization
  - Enabled by the authorization server group setting
  - Uses the username as a credential

If authentication is disabled and authorization is enabled, the ASA uses the primary DN field for authorization.

- Authentication
  - DISABLED (set to None) by the authentication server group setting
  - No credentials used

- Authorization
  - Enabled by the authorization server group setting
  - Uses the username value of the certificate primary DN field as a credential

Note If the primary DN field is not present in the certificate, the ASA uses the secondary DN field value as the username for the authorization request.

For example, consider a user certificate that includes the following Subject DN fields and values:

```
Cn=anyuser,OU=sales;O=XYZCorporation;L=boston;S=mass;C=us;ea=anyuser@example.com
```

If the Primary DN = EA (E-mail Address) and the Secondary DN = CN (Common Name), then the username used in the authorization request would be anyuser@example.com.

Guidelines for Digital Certificates

This section includes guidelines and limitations that you should check before configuring digital certificates.

Context Mode Guidelines

- Supported in single context mode only for third-party CAs.

Failover Guidelines

- Does not support replicating sessions in Stateful Failover.
- Does not support failover for local CAs.
- Certificates are automatically copied to the standby unit if you configure stateful failover. If you find a certificate is missing, use the **write standby** command on the active unit.

IPv6 Guidelines

Does not support IPv6.

Local CA Certificates

- Make sure that the ASA is configured correctly to support certificates. An incorrectly configured ASA can cause enrollment to fail or request a certificate that includes inaccurate information.
- Make sure that the hostname and domain name of the ASA are configured correctly. To view the currently configured hostname and domain name, enter the **show running-config** command.
- Make sure that the ASA clock is set accurately before configuring the CA. Certificates have a date and time that they become valid and expire. When the ASA enrolls with a CA and obtains a certificate, the ASA checks that the current time is within the valid range for the certificate. If it is outside that range, enrollment fails.
- Thirty days before the local CA certificate expires, a rollover replacement certificate is generated, and a syslog message informs the administrator that it is time for local CA rollover. The new local CA certificate must be imported onto all necessary devices before the current certificate expires. If the administrator does not respond by installing the rollover certificate as the new local CA certificate, validations may fail.
- The local CA certificate rolls over automatically after expiration using the same keypair. The rollover certificate is available for export in base 64 format.

  The following example shows a base 64 encoded local CA certificate:

```
MIIXlwIBAzCCF1EGCSqGSIb3DQEHAaCCF0IEghc+MIIXOjCCFzYGCSqGSIb3DQEHBqCCFycwghcjAgEAMIIXHA
YJKoZIhvcNAQcBMBsGCiqGSIb3DQEMAQMwDQQIjph4SxJoyTgCAQGAghbw3v4bFy+GGG2dJnB4OLphsUM+IG3S
DOiDwZG9n1SvtMieoxd7Hxknxbum06JDrujWKtHBIqkrm+td34qlNE1iGeP2YC94/NQ2z+4kS+uZzwcRhl1KEZ
TS1E4L0fSaC3uMTxJq2NUHYWmoc8pi4CIeLj3h7VVMy6qbx2AC8I+q57+QG5vG5l5Hi5imwtYfaWwPEdPQxaWZ
PrzoG1J8BFqdPa1jBGhAzzuSmElm3j/2dQ3Atro1G9nIsRHgV39fcBgwz4fEabHG7/Vanb+fj81d5nlOiJjDYY
bP86tvbZ2yOVZR6aKFVI0b2AfCr6PbwfC9U8Z/aF3BCyM2sN2xPJrXva94CaYrqyotZdAkSYA5KWScyEcgdqmu
BeGDKOncTknfgy0XM+fG5rb3qAXy1GkjyFI5Bm9Do6RUROoG1DSrQrKeq/hj….

END OF CERTIFICATE
```

SCEP Proxy Support

- Ensure that the ASA and the Cisco ISE Policy Service Nodes are synchronized using the same NTP server.
- AnyConnect Client 3.0 or later must be running at the endpoint.
- The authentication method, configured in the connection profile for your group policy, must be set to use both AAA and certificate authentication.
- An SSL port must be open for IKEv2 VPN connections.
- The CA must be in auto-grant mode.

Additional Guidelines

- The type of certificate you can use is constrained by the certificate types supported by the applications that will use the certificate. RSA certificates are generally supported by all applications that use certificates.

But EDDSA certificates might not be supported by workstation operating systems, browsers, ASDM, or AnyConnect Client. For example, you need to use an RSA certificate for remote access VPN identity and authentication. For site-to-site VPN, where the ASA is the application that uses the certificate, EDDSA is supported.

- For ASAs that are configured as CA servers or clients, limit the validity period of the certificate to less than the recommended end date of 03:14:08 UTC, January 19, 2038. This guideline also applies to imported certificates from third-party vendors.
- The ASA establishes LDAP/SSL connection only if one of the following certification criteria is satisfied:
  - The LDAP server certificate is trusted (exists in a trustpoint or the ASA trustpool) and is valid.
  - A CA certificate from servers issuing chain is trusted (exists in a trustpoint or the ASA trustpool) and all subordinate CA certificates in the chain are complete and valid.
- When a certificate enrollment is completed, the ASA stores a PKCS12 file containing the user's keypair and certificate chain, which requires about 2 KB of flash memory or disk space per enrollment. The actual amount of disk space depends on the configured RSA key size and certificate fields. Keep this guideline in mind when adding a large number of pending certificate enrollments on an ASA with a limited amount of available flash memory, because these PKCS12 files are stored in flash memory for the duration of the configured enrollment retrieval timeout. We recommend using a key size of at least 2048.
- You should configure the ASA to use an identity certificate to protect ASDM traffic and HTTPS traffic to the management interface. Identity certificates that are automatically generated with SCEP are regenerated after each reboot, so make sure that you manually install your own identity certificates. For an example of this procedure that applies only to SSL, see the following URL: http://www.cisco.com/en/US/products/ps6120/products_configuration_example09186a00809fcf91.shtml.
- The ASA and the AnyConnect Client can only validate certificates in which the X520Serialnumber field (the serial number in the Subject Name) is in PrintableString format. If the serial number format uses encoding such as UTF8, the certificate authorization will fail.
- Use only valid characters and values for certificate parameters when you import them on the ASA. In ASA, these certificates are decoded to build them into internal data structures. Certificates with blank fields are construed as non-compliant with the decoding standards, and hence the installation validation fails. However, from version 9.16, blank values of optional fields does not impact decoding and installation validation criteria.
- To use a wildcard (*) symbol, make sure that you use encoding on the CA server that allows this character in the string value. Although RFC 5280 recommends using either a UTF8String or PrintableString, you should use UTF8String because PrintableString does not recognize the wildcard as a valid character. The ASA rejects the imported certificate if an invalid character or value is found during the import. For example:

```
ERROR: Failed to parse or verify imported certificate ciscoasa(config)# Read 162*H÷ytes
 as CA certificate:0U0= \Ivr"phÕV°3é¼þ0 CRYPTO_PKI(make trustedCerts list)
CERT-C: E ../cert-c/source/certlist.c(302): Error #711h
CRYPTO_PKI: Failed to verify the ID certificate using the CA certificate in trustpoint
 mm.
CERT-C: E ../cert-c/source/p7contnt.c(169): Error #703h
crypto_certc_pkcs7_extract_certs_and_crls failed (1795):
crypto_certc_pkcs7_extract_certs_and_crls failed
CRYPTO_PKI: status = 1795: failed to verify or insert the cert into storage
```

- Configure DNS server on the device for a sucessful ACME enrollment. For information on DNS server configuration procedure, see Configure the DNS Server, on page 748.

Configure Digital Certificates

The following topics explain how to configure digital certificates.

Configure Key Pairs

To create or remove key pairs, perform the following steps.

Procedure

Step 1 Generate one default, general-purpose RSA key pair.

crypto key generate rsa modulus 2048

Example:

```
ciscoasa(config)# crypto key generate rsa modulus 2048
```

The default key modulus is 2048, but you should specify the modulus explicitly to ensure you get the size you require. The key is named Default-RSA-Key.

For RSA keys, the modulus can be one of the following (in bits): 2048 or 4096.

If you also want an Elliptic Curve Signature Algorithm (ECDSA) key, you can generate the Default-ECDSA-Key. The default length is 384, but you can also use 256 or 521.

crypto key generate ecdsa elliptic-curve 384

If you also want an Edwards Curve Signature Algorithm (EdDSA) key, you can generate the Default-EdDSA-Key. The default length is 256 bits.

Note
EST enrollments on the ASA using keypairs of type EdDSA (Ed25519) is not supported. EST enrollments can use only RSA or ECDSA keys.

crypto key generate eddsa edward-curve Ed25519

Step 2 (Optional) Create additional keys with unique names.

crypto key generate rsa label *key-pair-label* **modulus** *size*

crypto key generate ecdsa label *key-pair-label* **elliptic-curve** *size*

Example:

```
ciscoasa(config)# crypto key generate rsa label exchange modulus 2048
```

The label is referenced by the trustpoint that uses the key pair.

Step 3 Verify key pairs that you have generated.

show crypto key mypubkey {**rsa** | **ecdsa**}

Example:

```
ciscoasa/contexta(config)# show crypto mypubkey key rsa
```

Step 4 Save the key pair that you have generated.

write memory

Example:

```
ciscoasa(config)# write memory
```

Step 5 If necessary, remove existing key pairs so that you can generate new ones.

crypto key zeroize {**rsa** | **ecdsa**}

Example:

```
ciscoasa(config)# crypto key zeroize rsa
```

Step 6 (Optional) Archive the local CA server certificate and key pair.

copy

Example:

```
ciscoasa# copy LOCAL-CA-SERVER_0001.pl2 tftp://10.1.1.22/user6/
```

This command copies the local CA server certificate and key pair and all files from the ASA using either FTP or TFTP.

Note
Make sure that you back up all local CA files as often as possible.

Step 7 For enrollment methods that support auto-enrollment, key pairs are automatically generated and stored in flash at the time of enrollment. To enable this support, following key parameters are specified in the trustpoint:

The following example shows ECDSA keypair configuration:

```
ciscoasa(config-ca-trustpoint)# keypair ?

crypto-ca-trustpoint mode commands/options:
  WORD < 129 char  Name of key pair
  ecdsa            Generate ECDSA keys
  eddsa            Generate EDDSA keys
  rsa              Generate RSA keys
ciscoasa(config-ca-trustpoint)# keypair ecdsa elliptic-curve ?

crypto-ca-trustpoint mode commands/options:
  256  256 bits
  384  384 bits
  521  521 bits
ciscoasa(config-ca-trustpoint)# keypair ecdsa elliptic-curve 521
ciscoasa(config-ca-trustpoint)#
```

The following example shows RSA keypair configuration:

```
ciscoasa(config-ca-trustpoint)# keypair ?

crypto-ca-trustpoint mode commands/options:
  WORD < 129 char  Name of key pair
  ecdsa            Generate ECDSA keys
  eddsa            Generate EDDSA keys
  rsa              Generate RSA keys
ciscoasa(config-ca-trustpoint)# keypair ecdsa elliptic-curve ?

ciscoasa(config-ca-trustpoint)# keypair rsa modulus ?

crypto-ca-trustpoint mode commands/options:
  2048  2048 bits
  3072  3072 bits
  4096  4096 bits
ciscoasa(config-ca-trustpoint)# keypair rsa modulus 2048
ciscoasa(config-ca-trustpoint)#
```

Step 8 You can remove the key pairs by running the following command:

crypto key zeroize rsa

The following example shows how to remove key pairs:

Example:

```
ciscoasa(config)# crypto key zeroize rsa
WARNING: All RSA keys will be removed.
WARNING: All device certs issued using these keys will also be removed.

Do you really want to remove these keys? [yes/no] y
```

Configure Trustpoints

To configure a trustpoint, perform the following steps:

Procedure

Step 1 Create a trustpoint that corresponds to the CA from which the ASA needs to receive a certificate.

crypto ca trustpoint *trustpoint-name*

Example:

```
ciscoasa/contexta(config)# crypto ca trustpoint Main
```

You enter the crypto ca trustpoint configuration mode, which controls CA-specific trustpoint parameters that you may configure starting in Step 3.

Step 2 Specify the source interface for the CA certification:

enrollment interface *interface-name*

```
ciscoasa(config-ca-trustpoint)#  enrollment interface mgmt
```

Step 3 Choose one of the following options:

- Request automatic enrollment using SCEP with the specified trustpoint and configure the enrollment URL by running the following command:

  enrollment protocol scep *url*

  Example:

  ```
  ciscoasa/contexta(config-ca-trustpoint)# enrollment protocol scep url
  http://10.29.67.142:80/certsrv/mscep/mscep.dll
  ```

- Request automatic enrollment using CMP with the specified trustpoint and configure the enrollment URL by running the following command:

  enrollment protocol cmp *url*

  Example

  ```
  ciscoasa/ contexta(config-ca-trustpoint)# enrollment protocol cmp url
  http://10.29.67.142:80/certsrv/mscep/mscep.dll
  ```

  Note
  Carrier license must be included to enable CMP enrollment.

- Request manual enrollment with the specified trustpoint by pasting the certificate received from the CA to the terminal by running the following command:

  enrollment terminal

  ```
  ciscoasa/contexta(config-ca-trustpoint)# enrollment terminal
  ```

- Request self signed certificate by running the following command:

  enrollment self

- Request automatic enrollment using EST with the specified trustpoint and configure the enrollment URL by running the following command:

  enrollment protocol est *url*

  Example

  ```
  asa(config-ca-trustpoint)# enrollment protocol est ?

  crypto-ca-trustpoint mode commands/options:
    url CA server enrollment URL
  asa(config-ca-trustpoint)# enrollment protocol est url ?
   crypto-ca-trustpoint mode commands/options:
    LINE < 477 char  URL
  asa(config-ca-trustpoint)# enrollment protocol est url https://xyz.com/est
  ```

Step 4 If the trustpoint has been configured to use CMP in the step above, you can optionally enable the functionality that automatically requests certificates. Enter a percentage of the absolute lifetime of the certificate after which auto-enroll will be necessary and specify if you want to generate a new key while renewing the certificate.

```
[no] auto-enroll [<percent>] [regenerate]
```

Step 5 Specify the available CRL configuration options.

revocation-check crl none

Note
The **revocation-check crl none**, which was removed in 9.13(1), was restored.

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# revocation-check crl
ciscoasa/contexta(config-ca-trustpoint)# revocation-check none
```

Note
To enable either required or optional CRL checking, make sure that you configure the trustpoint for CRL management after obtaining certificates.

Step 6 Enable or disable the basic constraints extension and CA flag.

[**no**] **ca-check**

The basic constraints extension identifies whether the subject of the certificate is a Certificate Authority (CA), in which case the certificate can be used to sign other certificates. The CA flag is part of this extension. The presence of these items in a certificate indicates that the certificate's public key can be used to validate certificate signatures.

The **ca-check** command is enabled by default, so you need to enter this command only if you want to disable basic constraints and the CA flag.

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# no ca-check
```

Step 7 During enrollment, ask the CA to include the specified e-mail address in the Subject Alternative Name extension of the certificate.

email *address*

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# email example.com
```

Step 8 (Optional) Specify a retry period in minutes, and applies only to SCEP enrollment.

enrollment retry period

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# enrollment retry period 5
```

Step 9 (Optional) Specify a maximum number of permitted retries, and applies only to SCEP enrollment.

enrollment retry count

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# enrollment retry period 2
```

Step 10 During enrollment, ask the CA to include the specified fully qualified domain name in the Subject Alternative Name extension of the certificate.

fqdn *fqdn*

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# fqdn example.com
```

Step 11 During enrollment, ask the CA to include the IP address of the ASA in the certificate.

ip-address *ip-address*

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# ip-address 10.10.100.1
```

Step 12 Specify the key pair whose public key is to be certified.

keypair *name*

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# keypair exchange
```

Step 13 If you have trustpoints configured for CMP, determine if you want to generate EDDSA keys, EDCSA keys or RSA keys for any manual and automatic enrollments.

```
no keypair name | [rsa modulus 2048|4096] | [edcsa elliptic-curve 256|384|521] | [ eddsa
edwards-curve Ed25519 ]
```

Note
EST enrollments on the ASA using keypairs of type EDDSA (Ed25519) is not supported. EST enrollments can use only RSA and ECDSA keys. ACME enrollment with keypair type EDDSA is not supported.

Note
When the ECDHE_ECDSA cipher group is used, configure the trustpoint with a certificate that contain an ECDSA-capable key. Certificates with RSA key are not compatible with ECDSA ciphers.

Step 14 Configure OCSP URL overrides and trustpoints to use for validating OCSP responder certificates.

match certificate map-name override ocsp

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# match certificate examplemap override ocsp
```

Step 15 Configure the source interface for ASA to reach OCSP:

interface *nameif*

Example:

```
ciscoasa(config)# crypto ca trustpoint TP
ciscoasa(config-ca-trustpoint)# ocsp ?

crypto-ca-trustpoint mode commands/options:
  disable-nonce  Disable OCSP Nonce Extension
  interface      Configure Source interface
  url            OCSP server URL
ciscoasa(config-ca-trustpoint)# ocsp  interface
ciscoasa(config-ca-trustpoint)# ocsp interface ?

crypto-ca-trustpoint mode commands/options:
Current available interface(s):
  inside   Name of interface GigabitEthernet0/0.100
  inside1  Name of interface GigabitEthernet0/0.41
  mgmt     Name of interface Management0/0
  outside  Name of interface GigabitEthernet0/0.51
ciscoasa(config-ca-trustpoint)# ocsp interface mgmt
```

Step 16 Disable the nonce extension on an OCSP request. The nonce extension cryptographically binds requests with responses to avoid replay attacks.

ocsp disable-nonce

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# ocsp disable-nonce
```

Step 17 Configure an OCSP server for the ASA to use to check all certificates associated with a trustpoint rather than the server specified in the AIA extension of the client certificate.

ocsp url

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# ocsp url
```

Step 18 Specify a challenge phrase that is registered with the CA during enrollment. The CA usually uses this phrase to authenticate a subsequent revocation request.

password *string*

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# password mypassword
```

Step 19 Set one or more methods for revocation checking: CRL, OCSP, and none.

Note
When you are assigning OCSP URL for revocation checking, you can specify the interface (includes management interface) from where the OCSP is reachable. This interface value determines the routing decision.

revocation check

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# revocation check
```

Step 20 During enrollment, ask the CA to include the specified subject DN in the certificate. If a DN string includes a comma, enclose the value string within double quotes (for example, O=”Company, Inc.”).

subject-name *X.500 name*

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# myname X.500 examplename
```

Step 21 During enrollment, ask the CA to include the ASA serial number in the certificate.

serial-number

Example:

```
ciscoasa/contexta(config-ca-trustpoint)# serial number JMX1213L2A7
```

Step 22 Save the running configuration.

write memory

Example:

```
ciscoasa/contexta(config)# write memory
```

Configure CRLs for a Trustpoint

To use mandatory or optional CRL checking during certificate authentication, you must configure CRLs for each trustpoint. To configure CRLs for a trustpoint, perform the following steps:

Procedure

Step 1 Enter crypto ca trustpoint configuration mode for the trustpoint whose CRL configuration you want to modify.

crypto ca trustpoint *trustpoint-name*

Example:

```
ciscoasa (config)# crypto ca trustpoint Main
```

Note
Make sure that you have enabled CRLs before entering this command. In addition, the CRL must be available for authentication to succeed.

Step 2 Enter crl configuration mode for the current trustpoint.

crl configure

Example:

```
ciscoasa(config-ca-trustpoint)# crl configure
```

Tip
To set all CRL configuration parameters to default values, use the **default** command. At any time during CRL configuration, reenter this command to restart the procedure.

Step 3 Choose one of the following to configure retrieval policy:

- CRLs are retrieved only from the CRL distribution points (CDP) URLs that are specified in authenticated certificates.

  policy cdp

  ```
  ciscoasa(config-ca-crl)# policy cdp
  ```

  Note
  SCEP retrieval is not supported by distribution points specified in certificates.

- CRLs are retrieved only from certificate map match rule that you configure.

  policy static

  ```
  ciscoasa(config-ca-crl)# policy static
  ```

- CRLs are retrieved from CRL distribution points specified in authenticated certificates and from certificate map match rule that you configure.

  policy both

  ```
  ciscoasa(config-ca-crl)# policy both
  ```

Step 4 If you used the **static** or **both** keywords when you configured the CRL policy, you must configure certificate map match rule for CRL retrieval. You can now configure multiple static CDPs to a single map.

enrollment terminal

To remove a specific instance, in the **no** form of the command, include the sequence number or the URL. Ensure the specified values match the configured values. To remove all the entries of the map, simply use the **no** command.

Example:

```
ciscoasa(crypto ca trustpoint)#enrollment terminal

ciscoasa(crypto ca trustpoint)#match certificate Main override cdp  10 url http://192.0.2.10
ciscoasa(crypto ca trustpoint)#match certificate Main override cdp  20 url http://192.0.2.12
ciscoasa(crypto ca trustpoint)#match certificate Main override cdp  30 url http://192.0.2.13
```

Step 5 Specify HTTP, LDAP, or SCEP as the CRL retrieval method.

protocol http | **ldap** | **scep**

Example:

```
ciscoasa(config-ca-crl)# protocol http
```

Step 6 Configure how long the ASA caches CRLs for the current trustpoint. The *refresh-time* argument is the number of minutes that the ASA waits before considering a CRL stale.

cache-time *refresh-time*

Example:

```
ciscoasa(config-ca-crl)# cache-time 420
```

Step 7 Choose one of the following:

- Require the NextUpdate field to be present in CRLs. This is the default setting.

  enforcenextupdate

  ```
  ciscoasa(config-ca-crl)# enforcenextupdate
  ```

- Allow the NextUpdate field to be absent in CRLs.

  no enforcenextupdate

  ```
  ciscoasa(config-ca-crl)# no enforcenextupdate
  ```

Step 8 Identify the LDAP server to the ASA if LDAP is specified as the retrieval protocol. You can specify the server by DNS hostname or by IP address. You can also provide a port number if the server listens for LDAP queries on a port other than the default of 389.

ldap-defaults *server*

Example:

```
ciscoasa (config-ca-crl)# ldap-defaults ldap1
```

Note
If you use a hostname instead of an IP address to specify the LDAP server, make sure that you have configured the ASA to use DNS.

Step 9 Allow CRL retrieval if the LDAP server requires credentials.

ldap-dn *admin-DN password*

Example:

```
ciscoasa (config-ca-crl)# ldap-dn cn=admin,ou=devtest,o=engineering c00lRunZ
```

Step 10 Retrieve the current CRL from the CA represented by the specified trustpoint and test the CRL configuration for the current trustpoint.

crypto ca crl request *trustpoint*

Example:

```
ciscoasa (config-ca-crl)# crypto ca crl request Main
```

Step 11 Save the running configuration.

write memory

Example:

```
ciscoasa (config)# write memory
```

Export or Import a Trustpoint Configuration

To export and import a trustpoint configuration, perform the following steps:

Procedure

Step 1 Export a trustpoint configuration with all associated keys and certificates in PKCS12 format.

crypto ca export *trustpoint*

Example:

```
ciscoasa(config)# crypto ca export Main
```

The ASA displays the PKCS12 data in the terminal. You can copy the data. The trustpoint data is password protected; however, if you save the trustpoint data in a file, make sure that the file is in a secure location.

Step 2 Import keypairs and issued certificates that are associated with a trustpoint configuration.

crypto ca import *trustpoint* **pkcs12***passphrase*

Example:

```
ciscoasa(config)# crypto ca import Main pkcs12 ?
```

The ASA prompts you to paste the text into the terminal in base 64 format. The key pair imported with the trustpoint is assigned a label that matches the name of the trustpoint that you create.

Note
If an ASA has trustpoints that share the same CA, you can use only one of the trustpoints that share the CA to validate user certificates. To control which trustpoint that shares a CA is used for validation of user certificates issued by that CA, use the **support-user-cert-validation** keyword.

Examples

The following example exports PKCS12 data for the trustpoint Main with the passphrase Wh0zits:

```
ciscoasa(config)# crypto ca export Main pkcs12 Wh0zits

Exported pkcs12 follows:

[ PKCS12 data omitted ]

---End - This line not part of the pkcs12---
```

The following example manually imports PKCS12 data to the trustpoint Main with the passphrase Wh0zits:

```
ciscoasa (config)# crypto ca import Main pkcs12 Wh0zits

Enter the base 64 encoded pkcs12.
End with a blank line or the word "quit" on a line by itself:
[ PKCS12 data omitted ]
quit
INFO: Import PKCS12 operation completed successfully
```

The following example manually imports a certificate for the trustpoint Main:

```
ciscoasa (config)# crypto ca import Main certificate
% The fully qualified domain name in the certificate will be: securityappliance.example.com

Enter the base 64 encoded certificate.
End with a blank line or the word "quit" on a line by itself
[ certificate data omitted ]
quit
INFO: Certificate successfully imported
```

Configure CA Certificate Map Rules

You can configure rules based on the Issuer and Subject fields of a certificate. Using the rules you create, you can map IPsec peer certificates to tunnel groups with the **tunnel-group-map** command.

To configure a CA certificate map rule, perform the following steps:

Procedure

Step 1 Enter CA certificate map configuration mode for the rule you want to configure and specify the rule sequence number.

crypto ca certificate map [*map_name*]*sequence-number*

Example:

```
ciscoasa(config)# crypto ca certificate map test-map 10
```

If you do not specify the map name, the rule is added to the default map: DefaultCertificateMap. For each rule number, you can specify one or more fields to match.

Step 2 Specify the issuer name or subject name:

{**issuer-name** | **subject-name**} [**attr** *attribute*] *operator string*

Example:

```
ciscoasa(config-ca-cert-map)# issuer-name cn=asa.example.com
ciscoasa(config-ca-cert-map)# subject-name attr cn eq mycert
ciscoasa(config-ca-cert-map)# subject-name attr uid eq jcrichton
```

You can match the entire value, or specify the attributes that you want to match. The following are valid attributes:

- c—Country
- cn—Common Name
- dc—Domain Component
- dnq—DN Qualifier
- ea—Email Address
- genq—Generational Qualifier
- gn—Given Name
- i—Initials
- ip—IP Address
- l—Locality
- n—Name
- o—Organization Name
- ou—Organizational Unit
- ser—Serial Number

  Note
  Ensure you specify the serial number attribute in the subject-name. The certificate map only matches with a serial number attribute specified in the subject-name.

- sn—Surname
- sp—State/Province
- t—Title
- uid—User ID
- uname—Unstructured Name

The following are valid operators:

- eq—The field or attribute must be identical to the value given.

- ne—The field or attribute cannot be identical to the value given.
- co—Part or all of the field or attribute must match the value given.
- nc—No part of the field or attribute can match the value given.

Step 3 Specify the alternative subject name:

alt-subject-name *operator string*

Example:

```
ciscoasa(config-ca-cert-map)# alt-subject-name eq happydays
```

The following are valid operators:

- eq—The field must be identical to the value given.
- ne—The field cannot be identical to the value given.
- co—Part or all of the field must match the value given.
- nc—No part of the field can match the value given.

Step 4 Specify the extended key usage:

extended-key-usage *operator OID_string*

Example:

```
ciscoasa(config-ca-cert-map)# extended-key-usage nc clientauth
```

The following are valid operators:

- co—Part or all of the field must match the value given.
- nc—No part of the field can match the value given.

The following are valid OID strings:

- *string*—User-defined string.
- clientauth—Client Authentication (1.3.6.1.5.5.7.3.2)
- codesigning—Code Signing (1.3.6.1.5.5.7.3.3)
- emailprotection—Secure Email Protection (1.3.6.1.5.5.7.3.4)
- ocspsigning—OCSP Signing (1.3.6.1.5.5.7.3.9)
- serverauth—Server Authentication (1.3.6.1.5.5.7.3.1)
- timestamping—Time Stamping (1.3.6.1.5.5.7.3.8)

Configure Reference Identities

When the ASA is acting as a TLS client, it supports rules for verification of an application server's identity as defined in RFC 6125. This RFC specifies procedures for representing the reference identities (configured on the ASA) and verifying them against the presented identities (sent from the application server). If the presented identity cannot be matched against the configured reference identity, the connection is not established and an error is logged.

The server presents its identity by including one or more identifiers in the server certificate presented to the ASA while establishing the connection. Reference identities are configured on the ASA, to be compared to the identity presented in a server certificate during connection establishment. These identifiers are specific instances of the four identifier types specified in RFC 6125. The four identifier types are:

- CN_ID: A Relative Distinguished Name (RDN) in a certificate subject field that contains only one attribute-type-and-value pair of type Common Name (CN), where the value matches the overall form of a domain name. The CN value cannot be free text. A CN-ID reference identifier does not identify an application service.
- DNS-ID: A subjectAltName entry of type dNSName. This is a DNS domain name. A DNS-ID reference identifier does not identify an application service.
- SRV-ID: A subjectAltName entry of type otherName whose name form is SRVName as defined in RFC 4985. A SRV-ID identifier may contain both a domain name and an application service type. For example, a SRV-ID of "_imaps.example.net" would be split into a DNS domain name portion of "example.net" and an application service type portion of "imaps."
- URI-ID: A subjectAltName entry of type uniformResourceIdentifier whose value includes both (i) a "scheme" and (ii) a "host" component (or its equivalent) that matches the "reg-name" rule specified in RFC 3986. A URI-ID identifier must contain the DNS domain name, not the IP address, and not just the hostname. For example, a URI-ID of "sip:voice.example.edu" would be split into a DNS domain name portion of "voice.example.edu" and an application service type of "sip."

A reference identity is created when configuring one with a previously unused name. Once a reference identity has been created, the four identifier types and their associated values can be added or deleted from the reference identity. The reference identifiers MAY contain information identifying the application service and MUST contain information identifying the DNS domain name.

Before you begin

- Reference identities are used when connecting to the Syslog Server and the Smart Licensing server only. No other ASA SSL client mode connections currently support the configuration or use of reference identities.
- ASA implements all the rules for matching the identifiers described in RFC 6125 except for pinned certificates and fallback for interactive clients.
- Ability to pin certificates is not implemented. Therefore, `No Match Found, Pinned Certificate` will not occur. Also, a user will not be given the opportunity to pin a certificate if a match is not found since our implementation is not an interactive client.

Procedure

Step 1 Enter the **[no] crypto ca reference-identity** command in global configuration mode to place the ASA in ca-reference-identity mode.

[no] crypto ca reference-identity *reference-identity-name*

If a reference identity with this *reference-identity-name* is not found, a new reference identity is created. If the **no** form of the command is issued for a reference identity that is still in use, a warning is displayed and the reference identity is not deleted.

Step 2 Enter reference-ids while in ca-reference-identity mode. Multiple reference-ids of any type may be added to the reference identity.

- **[no] cn-id** *value*
- **[no] dns-id** *value*
- **[no] srv-id** *value*
- **[no] uri-id** *value*

To remove a reference identity, use the no form of the command.

Example

Configure a reference identity for RFC 6125 server certificate validation for a syslog server:

```
ciscoasa(config)# crypto ca reference-identity syslogServer
ciscoasa(config-ca-ref-identity)# dns-id syslog1-bxb.cisco.com
ciscoasa(config-ca-ref-identity)# cn-id syslog1-bxb.cisco.com
```

What to do next

Use the reference identity when configuring the Syslog and the Smart Call Home server connections.

Obtain Certificates Manually

To obtain certificates manually, perform the following steps:

Before you begin

You must have already obtained a base-64 encoded CA certificate from the CA represented by the trustpoint.

Procedure

Step 1 Import the CA certificate for the configured trustpoint.

crypto ca authenticate *trustpoint*

Example:

```
ciscoasa(config)# crypto ca authenticate Main
Enter the base 64 encoded CA certificate.
End with a blank line or the word "quit" on a line by itself
MIIDRTCCAu+gAwIBAgIQKVcqP/KW74VP0NZzL+JbRTANBgkqhkiG9w0BAQUFADCB
[ certificate data omitted ]
/7QEM8izy0EOTSErKu7Nd76jwf5e4qttkQ==
quit

INFO: Certificate has the following attributes:
Fingerprint:     24b81433 409b3fd5 e5431699 8d490d34
Do you accept this certificate? [yes/no]: y
Trustpoint CA certificate accepted.

% Certificate successfully imported
```

Whether a trustpoint requires that you manually obtain certificates is determined by the use of the **enrollment terminal** command when you configure the trustpoint.

Step 2 Enroll the ASA with the trustpoint.

crypto ca enroll *trustpoint*

Example:

```
ciscoasa(config)# crypto ca enroll Main
% Start certificate enrollment ..

% The fully-qualified domain name in the certificate will be: securityappliance.example.com

% Include the device serial number in the subject name? [yes/no]: n

Display Certificate Request to terminal? [yes/no]: y
Certificate Request follows:

MIIBoDCCAQkCAQAwIzEhMB8GCSqGSIb3DQEJAhYSRmVyYWxQaXguY2lzY28uY29t
[ certificate request data omitted ]
jF4waw68eOxQxVmdgMWeQ+RbIOYmvt8g6hnBTrd0GdqjjVLt

---End - This line not part of the certificate request---

Redisplay enrollment request? [yes/no]: n
```

This command generates a certificate for signing data and depending on the type of keys that you have configured, for encrypting data. If you use separate RSA keys for signing and encryption, the **crypto ca enroll** command displays two certificate requests, one for each key. If you use general-purpose RSA keys for both signing and encryption, the **crypto ca enroll** command displays one certificate request.

To complete enrollment, obtain a certificate for all certificate requests generated by the **crypto ca enroll** command from the CA represented by the applicable trustpoint. Make sure that the certificate is in base-64 format.

Step 3 When a trustpoint is configured for CMP, either a shared secret value (ir) or the name of the trustpoint that contains the certificate that will sign the request (cr) can be specified, but not both. Provide either an out-of-band value by the CA that is used to confirm the authenticity and integrity of messages exchanged with ASA or provide the name of the trustpoint with a previously-issued device certificate used for signing the CMP

enrollment request. The shared-secret or signing-certificate keywords are only available when the trustpoint enrollment protocol is set to CMP.

```
crypto ca enroll trustpoint [regenerate] [shared-secret <value> | signing-certificate <value>
```

Step 4 Determine whether or not a new keypair should be generated prior to building the enrollment request.

```
crypto ca enroll trustpoint [regenerate] [shared-secret <value> | signing-certificate <value>
```

Step 5 Import each certificate you receive from the CA and make sure that you paste the certificate to the terminal in base-64 format.

crypto ca import *trustpoint* **certificate**

Example:

```
ciscoasa (config)# crypto ca import Main certificate
% The fully-qualified domain name in the certificate will be: securityappliance.example.com

Enter the base 64 encoded certificate.
End with a blank line or the word "quit" on a line by itself
[ certificate data omitted ]
quit
INFO: Certificate successfully imported
```

Step 6 Verify that the enrollment process was successful by displaying certificate details issued for the ASA and the CA certificate for the trustpoint.

show crypto ca certificate

Example:

```
ciscoasa(config)# show crypto ca  certificate Main
```

Step 7 Save the running configuration.

write memory

Example:

```
ciscoasa(config)# write memory
```

Step 8 Repeat these steps for each trustpoint that you configure for manual enrollment.

Obtain Certificates Automatically with SCEP

This section describes how to obtain certificates automatically using SCEP.

Before you begin

You must have already obtained a base-64 encoded CA certificate from the CA represented by the trustpoint.

Procedure

Step 1 Obtain the CA certificate for the configured trustpoint.

crypto ca authenticate *trustpoint*

Example:

```
ciscoasa/contexta(config)# crypto ca authenticate Main
```

When you configure the trustpoint, use of the **enrollment url** command determines whether or not you must obtain certificates automatically via SCEP.

Step 2 Enroll the ASA with the trustpoint. This command retrieves a certificate for signing data and depending on the type of keys that you have configured, for encrypting data. Before entering this command, contact the CA administrator, who may need to authenticate the enrollment request manually before the CA grants certificates.

crypto ca enroll *trustpoint*

Example:

```
ciscoasa/contexta(config)# crypto ca enroll Main
```

If the ASA does not receive a certificate from the CA within one minute (the default) of sending a certificate request, it resends the certificate request. The ASA continues sending a certificate request each minute until a certificate is received.

If the fully qualified domain name configured for the trustpoint is not identical to the fully qualified domain name of the ASA, including the case of the characters, a warning appears. To resolve this issue, exit the enrollment process, make any necessary corrections, and reenter the **crypto ca enroll** command.

Note
If the ASA reboots after you have issued the **crypto ca enroll** command but before you have received the certificate, reenter the **crypto ca enroll** command and notify the CA administrator.

Step 3 Verify that the enrollment process was successful by displaying certificate details issued for the ASA and the CA certificate for the trustpoint.

show crypto ca certificate

Example:

```
ciscoasa/contexta(config)# show crypto ca certificate Main
```

Step 4 Save the running configuration.

write memory

Example:

```
ciscoasa/contexta(config)# write memory
```

Configure Proxy Support for SCEP Requests

To configure the ASA to authenticate remote access endpoints using third-party CAs, perform the following steps:

Procedure

Step 1 Enter tunnel-group ipsec-attributes configuration mode.

tunnel-group *name* **ipsec-attributes**

Example:

```
ciscoasa(config)# tunnel-group remotegrp ipsec-attributes
```

Step 2 Enable client services.

crypto ikev2 enable outside client-services port *portnumber*

Example:

```
ciscoasa(config-tunnel-ipsec)# crypto ikev2 enable outside client-services
```

The default port number is 443.

Note
This command is needed only if you support IKEv2.

Step 3 Enter tunnel-group general-attributes configuration mode.

tunnel-group *name* **general-attributes**

Example:

```
ciscoasa(config)# tunnel-group 209.165.200.225 general-attributes
```

Step 4 Enable SCEP enrollment for the tunnel group.

scep-enrollment enable

Example:

```
ciscoasa(config-tunnel-general)# scep-enrollment enable
INFO: 'authentication aaa certificate' must be configured to complete setup of this option.
```

Step 5 Enter group-policy attributes configuration mode.

group-policy *name* **attributes**

Example:

```
ciscoasa(config)# group-policy FirstGroup attributes
```

Step 6 Enroll the SCEP CA for the group policy. Enter this command once per group policy to support a third-party digital certificate.

scep-forwarding-url value *URL*

Example:

```
ciscoasa(config-group-policy)# scep-forwarding-url value http://ca.example.com:80/
```

URL is the SCEP URL on the CA.

Step 7 Supply a common, secondary password when a certificate is unavailable for WebLaunch support of the SCEP proxy.

secondary-pre-fill-username clientless hide use-common-password *password*

Example:

```
ciscoasa(config)# tunnel-group remotegrp webvpn-attributes
ciscoasa(config-tunnel-webvpn)# secondary-pre-fill-username clientless hide
use-common-password secret
```

You must use the **hide** keyword to support the SCEP proxy.

For example, a certificate is not available to an endpoint requesting one. Once the endpoint has the certificate, AnyConnect Client disconnects, then reconnects to the ASA to qualify for a DAP policy that provides access to internal network resources.

Step 8 Hide the secondary prefill username for AnyConnect Client VPN sessions.

secondary-pre-fill-username ssl-client hide use-common-password *password*

Example:

```
ciscoasa(config-tunnel-webvpn)# secondary-pre-fill-username ssl-client hide
use-common-password secret
```

Despite the **ssl-client** keyword inherited from earlier releases, use this command to support AnyConnect Client sessions that use either IKEv2 or SSL.

You must use the **hide** keyword to support the SCEP proxy.

Step 9 Supply the username when a certificate is unavailable.

secondary-username-from-certificate {**use-entire-name** | **use-script** | {*primary_attr* [*secondary-attr*]}} [**no-certificate-fallback cisco-secure-desktop machine-unique-id**]

Example:

```
ciscoasa(config-tunnel-webvpn)# secondary-username-from-certificate CN no-certificate-fallback
 cisco-secure-desktop machine-unique-id
```

Configure CAs

Configure Auto-Import of Trustpool Certificates

Smart licensing uses the Smart Call Home infrastructure. When the ASA configures Smart Call Home anonymous reporting in the background, the ASA automatically creates a trustpoint containing the certificate of the CA that issued the Call Home server certificate. The ASA now supports validation of the certificate if the issuing hierarchy of the server certificate changes, without the need for customer involvement to adjust certificate hierarchy changes. You can automate the update of the trustpool bundle at periodic intervals so that Smart Call Home can remain active if the self-signed certificate of the CA server changes. This feature is not supported under multi-context deployments.

Procedure

	Command or Action	Purpose
Step 1	Automatic import of trustpool certificate bundles requires you to specify the URL that ASA uses to download and import the bundle. Use the following command so the import happens daily at a regular interval with the default Cisco URL and default time of 22 hours:	`ciscoasa(config-ca-trustpool)# auto-import-url Default`
Step 2	You can also enable auto import with a custom URL with the following command:	`ciscoasa(config-ca-trustpool)# auto-import url http://www.thawte.com`
Step 3	To give you more flexibility to set downloads during off peak hours or other convenient times, enter the following command which enables the import with a custom time:	`ciscoasa(config-ca-trustpool)# auto-import time 23:23:23`
Step 4	Setting the automatic import with both a custom URL and custom time requires the following command:	`ciscoasa(config-ca-trustpool)# auto-import time 23:23:23 url http://www.thawte.com`

Show the State of the Trustpool Policy

Use the following command to see the current state of the trustpool policy:

```
show crypto ca trustpool policy
```

This command returns information like the following:

```
0 trustpool certificates installed
Trustpool auto renewal statistics:
 State: Not in progress
 Last import result: Not attempted N/A
```

```
 Current Jitter: 0

Trustpool auto import statistics:
 Last import result: N/A
 Next schedule import at 22:00:00 Tues Jul 21 2015

Trustpool Policy

 Trustpool revocation checking is disabled.
 CRL cache time: 60 seconds
 CRL next update field: required and enforced
 Auto import of trustpool is enabled
 Automatic import URL: http://www.cisco.com/security/pki/trs/ios_core.p7b
 Download time: 22:00:00

Policy Overrides:
 None configured
```

Clear CA Trustpool

To reset the trustpool policy to its default state, use the following command:

```
clear configure crypto ca trustpool
```

Since the automatic import of trustpoint certificates is turned off by default, using this command disables the feature.

Set a Certificate Expiration Alert (for Identity or CA Certificates)

ASA checks all the CA and ID certificates in the trust points for expiration once every 24 hours. If a certificate is nearing expiration, a syslog will be issued as an alert.

A CLI is provided to configure the reminder and recurrence intervals. By default, reminders start at 60 days prior to expiration and recur every 7 days. You can configure the interval at which reminders are sent and the number of days before the expiration at which the first alert is sent by using the following command:

```
[no] crypto ca alerts expiration [begin <days before expiration>] [repeat <days>]
```

Irrespective of the alerts configuration, a reminder is sent every day during the last week of expiration. The following **show** and **clear** commands have also been added:

```
clear conf crypto ca alerts
show run crypto ca alerts
```

In addition to the renewal reminder, if an already expired certificate is found in the configuration, a syslog is generated once every day to rectify the configuration by either renewing the certificate or removing the expired certificate.

For example, assume that the expiration alerts are configured to begin at 60 days and repeat every 6 days after that. If the ASA is rebooted at 40 days, an alert is sent on that day, and the next alert is sent on the 36th day.

Note Expiration checking is not done on trust pool certificates. The Local CA trust point is treated as a regular trustpoint for expiration checking too.

Monitoring Digital Certificates

See the following commands for monitoring digital certificate status:

- **show crypto ca server**

  This command shows local CA configuration and status.

- **show crypto ca server cert-db**

  This command shows user certificates issued by the local CA.

- **show crypto ca server certificate**

  This command shows local CA certificates on the console in base 64 format and the rollover certificate when available, including the rollover certificate thumb print for verification of the new certificate during import onto other devices.

- **show crypto ca server crl**

  This command shows CRLs.

- **show crypto ca server user-db**

  This command shows users and their status, which can be used with the following qualifiers to reduce the number of displayed records:

  - allowed. Shows only users currently allowed to enroll.
  - enrolled. Shows only users that are enrolled and hold a valid certificate
  - expired. Shows only users holding expired certificates.
  - on-hold. Lists only users without a certificate and not currently allowed to enroll.

- **show crypto ca server user-db allowed**

  This command shows users who are eligible to enroll.

- **show crypto ca server user-db enrolled**

  This command shows enrolled users with valid certificates.

- **show crypto ca server user-db expired**

  This command shows users with expired certificates.

- **show crypto ca server user-db on-hold**

  This command shows users without certificates who are not allowed to enroll.

- **show crypto key** *name of key*

  This command shows key pairs that you have generated.

- **show running-config**

  This command shows local CA certificate map rules.

Examples

The following example shows an RSA general-purpose key:

```
ciscoasa/contexta(config)# show crypto key mypubkey rsa
Key pair was generated at: 16:39:47 central Feb 10 2010
Key name: <Default-RSA-Key>
 Usage: General Purpose Key
 Modulus Size (bits): 2048
 Storage: config
 Key Data:

  30820122 300d0609 2a864886 f70d0101 01050003 82010f00 3082010a 02820101
  00ea2c38 df9c606e ddb7b08a e8b0a1a8 65592d85 0711cac5 fceddee1 fa494297
  525fffc0 90da8a4c e696e44e 0646c661 48b3602a 960d7a3a 52dae14a 5f983603
  e1f33e40 a6ce04f5 9a812894 b0fe0403 f8d7e05e aea79603 2dcd56cc 01261b3e
  93bff98f df422fb1 2066bfa4 2ff5d2a4 36b3b1db edaebf16 973b2bd7 248e4dd2
  071a978c 6e81f073 0c4cd57b db6d9f40 69dc2149 e755fb0f 590f2da8 b620efe6
  da6e8fa5 411a841f e72bb8ea cf4bdb79 f4e57ff3 a940ce3b 4a2c7052 56c1d17b
  af8fe2e2 e58718c6 ed1da0f0 1c6f36eb 79eb1aeb f098b5c4 79e07658 a52d8c7a
  51ceabfb f8ade096 7217cf2d 3728077e 89441d89 9bf5f875 c8d2db39 c858bb7a
  7d020301 0001
```

The following example shows the local CA CRL:

```
ciscoasa(config)# show crypto ca server crl
Certificate Revocation List:
    Issuer: cn=xx5520-1-3-2007-1
    This Update: 13:32:53 UTC Jan 4 2010
    Next Update: 13:32:53 UTC Feb 3 2010
    Number of CRL entries: 2
    CRL size: 270 bytes
Revoked Certificates:
    Serial Number: 0x6f
    Revocation Date: 12:30:01 UTC Jan 4 2010
    Serial Number: 0x47
    Revocation Date: 13:32:48 UTC Jan 4 2010
```

The following example shows one user on-hold:

```
ciscoasa(config)# show crypto ca server user-db on-hold
username: wilma101
email:    <None>
dn:       <None>
allowed:  <not allowed>
notified: 0
ciscoasa(config)#
```

The following example shows output of the **show running-config** command, in which local CA certificate map rules appear:

```
crypto ca certificate map 1
 issuer-name co asc
 subject-name attr ou eq Engineering
```

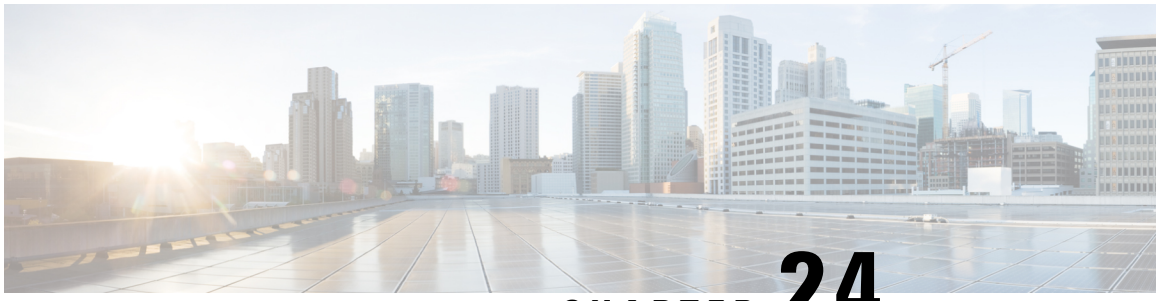
History for Certificate Management

Table 30: History for Certificate Management

Feature Name	Platform Releases	Description
Certificate management	7.0(1)	Digital certificates (including CA certificates, identity certificates, and code signer certificates) provide digital identification for authentication. A digital certificate includes information that identifies a device or user, such as the name, serial number, company, department, or IP address. CAs are trusted authorities that "sign" certificates to verify their authenticity, thereby guaranteeing the identity of the device or user. CAs issue digital certificates in the context of a PKI, which uses public-key or private-key encryption to ensure security.
Certificate management	7.2(1)	We introduced the following commands: **issuer-name** *DN-string*, **revocation-check crl none**, **revocation-check crl**, **revocation-check none**. We deprecated the following commands: **crl** {**required** | **optional** | **nocheck**}.
Certificate management	8.0(2)	We introduced the following commands: **cdp-url**, **crypto ca server**, **crypto ca server crl issue**, **crypto ca server revoke** *cert-serial-no*, **crypto ca server unrevoke** *cert-serial-no*, **crypto ca server user-db add** *user* [**dn** *dn*] [**email** *e-mail-address*], **crypto ca server user-db allow** {*username* | **all-unenrolled** | **all-certholders**} [**display-otp**] [**email-otp**] [**replace-otp**], **crypto ca server user-db email-otp** {*username* | **all-unenrolled** | **all-certholders**}, **crypto ca server user-db remove** *username*, **crypto ca server user-db show-otp** {*username* | **all-certholders** | **all-unenrolled**}, **crypto ca server user-db write**, [**no**] **database path** *mount-name directory-path*, **debug crypto ca server** [*level*], **lifetime** {**ca-certificate** | **certificate** | **crl**} *time*, **no shutdown**, **otp expiration** *timeout*, **renewal-reminder** *time*, **show crypto ca server**, **show crypto ca server cert-db** [**user** *username* | **allowed** | **enrolled** | **expired** | **on-hold**] [**serial** *certificate-serial-number*], **show crypto ca server certificate**, **show crypto ca server crl**, **show crypto ca server user-db** [**expired** | **allowed** | **on-hold** | **enrolled**], **show crypto key** *name of key*, **show running-config**, **shutdown**.

Feature Name	Platform Releases	Description
SCEP proxy	8.4(1)	We introduced this feature, which provides secure deployment of device certificates from third-party CAs. We introduced the following commands: **crypto ikev2 enable outside client-services port** *portnumber*, **scep-enrollment enable**, **scep-forwarding-url value** *URL*, **secondary-pre-fill-username clientless hide use-common-password** *password*, **secondary-pre-fill-username ssl-client hide use-common-password** *password*, **secondary-username-from-certificate** {**use-entire-name** | **use-script** | {*primary_attr* [*secondary-attr*]}} [**no-certificate-fallback cisco-secure-desktop machine-unique-id**].
Reference Identities	9.6(2)	TLS client processing now supports rules for verification of a server identity defined in RFC 6125, Section 6. Identity verification will be done during PKI validation for TLS connections to the Syslog Server and the Smart Licensing server only. If the presented identity cannot be matched against the configured reference identity, the connection is not established. We added or modified the following commands: **crypto ca reference-identity**, **logging host**, **call home profile destination address**
Local CA Server	9.12(1)	To make the FQDN of the enrollment URL configurable instead of using the ASA's configured FQDN, a new CLI option is introduced. This new option is added to the smpt mode of crypto ca server. We deprecated Local CA Server and will be removing in a later release—When ASA is configured as local CA server, it is enabled to issue digital certificates, publish Certificate Revocation Lists (CRLs), and securely revoke issued certificates. This feature has become obsolete and hence the crypto ca server command is deprecated.
Local CA Server	9.13(1)	We removed the local ca server support. Thus, the **crypto ca server** command and its subcommands are removed. We removed the following commands: **crypto ca server** and all of its subcommands.
Modifications to the CRL Distribution Point commands	9.13(1)	The static CDP URL configuration commands are removed and moved to the match certificate command. New/Modified commands: **crypto-ca-trustpoint crl** and **crl url** were removed with other related logic. **match-certificate override-cdp** was introduced.

Feature Name	Platform Releases	Description
CRL cache size increased	9.13(1)	To prevent failure of large CRL downloads, the cache size was increased, and the limit on the number of entries in an individual CRL was removed. • Increased the total CRL cache size to 16 MB per context for multi-context mode. • Increased the total CRL cache size to 128 MB for single-context mode.
Restoration of bypass certificate validity checks option	9.15(1)	The option to bypass revocation checking due to connectivity problems with the CRL or OCSP server that was removed in 9.13(1) was restored. New/Modified commands: **revocation-check crl none**, **revocation-check ocsp none**, **revocation-check crl ocsp none**, and **revocation-check ocsp crl none** were restored.
Modifications to Match Certificate commands to support static CRL Distribution Point URL	9.15(1)	The static CDP URL configuration command allowed static CDPs to be mapped uniquely to each certificate in a chain that is being validated. However, only one such mapping was supported for each certificate. This modification allows statically configured CDPs to be mapped to a chain of certificates for authentication. New/Modified commands: **match certificate** *map* **override cdp** *seq* **url** *url* and **no match certificate** *map* **override cdp** *seq* **url** *url*
Modifications to the trustpoint keypair and crypto key generate commands	9.16(1)	Support for certificates with key sizes smaller than 2048 was removed. Any configuration using 512, 768 or 1024-bit options are transitioned to 2048 with due notification. Support to use SHA1 hashing algorithm for certification was removed. **Note** **crypto ca permit-weak-crypto** command was introduced to override these restrictions. The new key option - EDDSA was added to the existing RSA and ECDSA options.



CHAPTER 24

ARP Inspection and the MAC Address Table

This chapter describes how to customize the MAC address table and configure ARP Inspection for bridge groups.

- About ARP Inspection and the MAC Address Table, on page 825
- Default Settings, on page 826
- Guidelines for ARP Inspection and the MAC Address Table, on page 826
- Configure ARP Inspection and Other ARP Parameters, on page 827
- Customize the MAC Address Table for Bridge Groups, on page 829
- Monitoring ARP Inspection and the MAC Address Table, on page 831
- History for ARP Inspection and the MAC Address Table, on page 831

About ARP Inspection and the MAC Address Table

For interfaces in a bridge group, ARP inspection prevents a “man-in-the-middle” attack. You can also customize other ARP settings. You can customize the MAC address table for bridge groups, including adding a static ARP entry to guard against MAC spoofing.

ARP Inspection for Bridge Group Traffic

By default, all ARP packets are allowed between bridge group members. You can control the flow of ARP packets by enabling ARP inspection.

ARP inspection prevents malicious users from impersonating other hosts or routers (known as ARP spoofing). ARP spoofing can enable a “man-in-the-middle” attack. For example, a host sends an ARP request to the gateway router; the gateway router responds with the gateway router MAC address. The attacker, however, sends another ARP response to the host with the attacker MAC address instead of the router MAC address. The attacker can now intercept all the host traffic before forwarding it on to the router.

ARP inspection ensures that an attacker cannot send an ARP response with the attacker MAC address, so long as the correct MAC address and the associated IP address are in the static ARP table.

When you enable ARP inspection, the ASA compares the MAC address, IP address, and source interface in all ARP packets to static entries in the ARP table, and takes the following actions:

- If the IP address, MAC address, and source interface match an ARP entry, the packet is passed through.
- If there is a mismatch between the MAC address, the IP address, or the interface, then the ASA drops the packet.

- If the ARP packet does not match any entries in the static ARP table, then you can set the ASA to either forward the packet out all interfaces (flood), or to drop the packet.

Note The dedicated Management interface never floods packets even if this parameter is set to flood.

MAC Address Table

When you use bridge groups, the ASA learns and builds a MAC address table in a similar way as a normal bridge or switch: when a device sends a packet through the bridge group, the ASA adds the MAC address to its table. The table associates the MAC address with the source interface so that the ASA knows to send any packets addressed to the device out the correct interface. Because traffic between bridge group members is subject to the ASA security policy, if the destination MAC address of a packet is not in the table, the ASA does not flood the original packet on all interfaces as a normal bridge does. Instead, it generates the following packets for directly-connected devices or for remote devices:

- Packets for directly-connected devices—The ASA generates an ARP request for the destination IP address, so that it can learn which interface receives the ARP response.
- Packets for remote devices—The ASA generates a ping to the destination IP address so that it can learn which interface receives the ping reply.

The original packet is dropped.

For routed mode, you can optionally enable flooding of non-IP packets on all interfaces.

Default Settings

- If you enable ARP inspection, the default setting is to flood non-matching packets.
- The default timeout value for dynamic MAC address table entries is 5 minutes.
- By default, each interface automatically learns the MAC addresses of entering traffic, and the ASA adds corresponding entries to the MAC address table.

Note ASA generates a reset packet to reset a connection that is denied by a stateful inspection engine. Here, the destination MAC address of the packet is not determined based on the ARP table lookup but instead it is taken directly from the packets (connections) that are being denied.

Guidelines for ARP Inspection and the MAC Address Table

- ARP inspection is only supported for bridge groups.

- MAC address table configuration is only supported for bridge groups.

Configure ARP Inspection and Other ARP Parameters

For bridge groups, you can enable ARP inspection. You can also configure other ARP parameters for both bridge groups and for routed mode interfaces.

Procedure

Step 1 Add static ARP entries according to Add a Static ARP Entry and Customize Other ARP Parameters, on page 827. ARP inspection compares ARP packets with static ARP entries in the ARP table, so static ARP entries are required for this feature. You can also configure other ARP parameters.

Step 2 Enable ARP inspection according to Enable ARP Inspection, on page 828.

Add a Static ARP Entry and Customize Other ARP Parameters

By default for bridge groups, all ARP packets are allowed between bridge group member interfaces. You can control the flow of ARP packets by enabling ARP inspection. ARP inspection compares ARP packets with *static* ARP entries in the ARP table.

For routed interfaces, you can enter static ARP entries, but normally dynamic entries are sufficient. For routed interfaces, the ARP table is used to deliver packets to directly-connected hosts. Although senders identify a packet destination by an IP address, the actual delivery of the packet on Ethernet relies on the Ethernet MAC address. When a router or host wants to deliver a packet on a directly connected network, it sends an ARP request asking for the MAC address associated with the IP address, and then delivers the packet to the MAC address according to the ARP response. The host or router keeps an ARP table so it does not have to send ARP requests for every packet it needs to deliver. The ARP table is dynamically updated whenever ARP responses are sent on the network, and if an entry is not used for a period of time, it times out. If an entry is incorrect (for example, the MAC address changes for a given IP address), the entry needs to time out before it can be updated with the new information.

For transparent mode, the ASA only uses dynamic ARP entries in the ARP table for traffic to and from the ASA, such as management traffic.

You can also set the ARP timeout and other ARP behavior.

Procedure

Step 1 Add a static ARP entry:

arp *interface_name ip_address mac_address* [**alias**]

Example:

```
ciscoasa(config)# arp outside 10.1.1.1 0009.7cbe.2100
```

This example allows ARP responses from the router at 10.1.1.1 with the MAC address 0009.7cbe.2100 on the outside interface.

Specify **alias** in routed mode to enable proxy ARP for this mapping. If the ASA receives an ARP request for the specified IP address, then it responds with the ASA MAC address. This keyword is useful if you have devices that do not perform ARP, for example. In transparent firewall mode, this keyword is ignored; the ASA does not perform proxy ARP.

Step 2 Set the ARP timeout for dynamic ARP entries:

arp timeout *seconds*

Example:

```
ciscoasa(config)# arp timeout 5000
```

This field sets the amount of time before the ASA rebuilds the ARP table, between 60 to 4294967 seconds. The default is 14400 seconds. Rebuilding the ARP table automatically updates new host information and removes old host information. You might want to reduce the timeout because the host information changes frequently.

Step 3 Allow non-connected subnets:

arp permit-nonconnected

The ASA ARP cache only contains entries from directly-connected subnets by default. You can enable the ARP cache to also include non-directly-connected subnets. We do not recommend enabling this feature unless you know the security risks. This feature could facilitate denial of service (DoS) attack against the ASA; a user on any interface could send out many ARP replies and overload the ASA ARP table with false entries.

You may want to use this feature if you use:

- Secondary subnets.
- Proxy ARP on adjacent routes for traffic forwarding.

Step 4 Set the ARP rate limit to control the number of ARP packets per second:

arp rate-limit *seconds*

Example:

```
ciscoasa(config)# arp rate-limit 1000
```

Enter a value between 10 and 32768. The default value depends on your ASA model. You can customize this value to prevent an ARP storm attack.

Enable ARP Inspection

This section describes how to enable ARP inspection for bridge groups.

Procedure

Enable ARP inspection:

arp-inspection *interface_name* **enable** [**flood** | **no-flood**]

Example:

```
ciscoasa(config)# arp-inspection outside enable no-flood
```

The **flood** keyword forwards non-matching ARP packets out all interfaces, and **no-flood** drops non-matching packets.

The default setting is to flood non-matching packets. To restrict ARP through the ASA to only static entries, then set this command to **no-flood**.

Customize the MAC Address Table for Bridge Groups

This section describes how you can customize the MAC address table for bridge groups.

Add a Static MAC Address for Bridge Groups

Normally, MAC addresses are added to the MAC address table dynamically as traffic from a particular MAC address enters an interface. You can add static MAC addresses to the MAC address table. One benefit to adding static entries is to guard against MAC spoofing. If a client with the same MAC address as a static entry attempts to send traffic to an interface that does not match the static entry, then the ASA drops the traffic and generates a system message. When you add a static ARP entry (see Add a Static ARP Entry and Customize Other ARP Parameters, on page 827), a static MAC address entry is automatically added to the MAC address table.

To add a static MAC address to the MAC address table, perform the following steps.

Procedure

Add a static MAC address entry:

mac-address-table static *interface_name mac_address*

Example:

```
ciscoasa(config)# mac-address-table static inside 0009.7cbe.2100
```

The *interface_name* is the source interface.

Set the MAC Address Timeout

The default timeout value for dynamic MAC address table entries is 5 minutes, but you can change the timeout. To change the timeout, perform the following steps.

Procedure

Set the MAC address entry timeout:

mac-address-table aging-time *timeout_value*

Example:

```
ciscoasa(config)# mac-address-table aging-time 10
```

The *timeout_value* (in minutes) is between 5 and 720 (12 hours). 5 minutes is the default.

Configure MAC Address Learning

By default, each interface automatically learns the MAC addresses of entering traffic, and the ASA adds corresponding entries to the MAC address table. You can disable MAC address learning if desired, however, unless you statically add MAC addresses to the table, no traffic can pass through the ASA. In routed mode, you can enable flooding of non-IP packets on all interfaces.

To configure MAC address learning, perform the following steps:

Procedure

Step 1 Disable MAC address learning:

mac-learn *interface_name* **disable**

Example:

```
ciscoasa(config)# mac-learn inside disable
```

The **no** form of this command reenables MAC address learning.

The **clear configure mac-learn** command reenables MAC address learning on all interfaces.

Step 2 (Routed mode only) Enable flooding of non-IP packets.

mac-learn flood

Example:

```
ciscoasa(config)# mac-learn flood
```

Monitoring ARP Inspection and the MAC Address Table

- **show arp-inspection**

  Monitors ARP Inspection. Shows the current settings for ARP inspection on all interfaces.

- **show mac-address-table [*interface_name*]**

  Monitors the MAC address table. You can view the entire MAC address table (including static and dynamic entries for both interfaces), or you can view the MAC address table for an interface.

  The following is sample output from the **show mac-address-table** command that shows the entire table:

```
ciscoasa# show mac-address-table
interface        mac address           type         Time Left
-----------------------------------------------------------------------
outside     0009.7cbe.2100       static    -
inside     0010.7cbe.6101       static    -
inside     0009.7cbe.5101       dynamic    10
```

  The following is sample output from the **show mac-address-table** command that shows the table for the inside interface:

```
ciscoasa# show mac-address-table inside
interface        mac address       type         Time Left
-----------------------------------------------------------------------
inside     0010.7cbe.6101       static    -
inside     0009.7cbe.5101       dynamic    10
```

  The following is sample output from the **show mac-address-table** command that shows the total number of static and dynamic bridge group entries:

```
ciscoasa# show mac-address-table count
Static     mac-address bridges (curr/max): 0/16384
Dynamic    mac-address bridges (curr/max): 0/16384
```

History for ARP Inspection and the MAC Address Table

Feature Name	Platform Releases	Feature Information
ARP inspection	7.0(1)	ARP inspection compares the MAC address, IP address, and source interface in all ARP packets to static entries in the ARP table. This feature is available for Transparent Firewall Mode, and for interfaces in a bridge group in both Transparent and Routed modes starting in 9.7(1). We introduced the following commands: **arp**, **arp-inspection**, and **show arp-inspection**.

Feature Name	Platform Releases	Feature Information
MAC address table	7.0(1)	You might want to customize the MAC address table for transparent mode, and for interfaces in a bridge group in both Transparent and Routed modes starting in 9.7(1). We introduced the following commands: **mac-address-table static**, **mac-address-table aging-time**, **mac-learn disable**, and **show mac-address-table**.
ARP cache additions for non-connected subnets	8.4(5)/9.1(2)	The ASA ARP cache only contains entries from directly-connected subnets by default. You can now enable the ARP cache to also include non-directly-connected subnets. We do not recommend enabling this feature unless you know the security risks. This feature could facilitate denial of service (DoS) attack against the ASA; a user on any interface could send out many ARP replies and overload the ASA ARP table with false entries. You may want to use this feature if you use: • Secondary subnets. • Proxy ARP on adjacent routes for traffic forwarding. We introduced the following command: **arp permit-nonconnected**.
Customizable ARP rate limiting	9.6(2)	You can set the maximum number of ARP packets allowed per second. The default value depends on your ASA model. You can customize this value to prevent an ARP storm attack. We added the following commands: **arp rate-limit, show arp rate-limit**

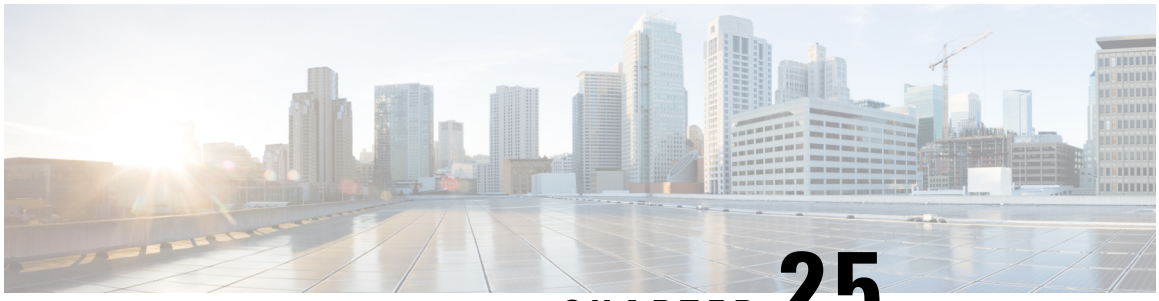
Feature Name	Platform Releases	Feature Information
Integrated Routing and Bridging	9.7(1)	Integrated Routing and Bridging provides the ability to route between a bridge group and a routed interface. A bridge group is a group of interfaces that the ASA bridges instead of routes. The ASA is not a true bridge in that the ASA continues to act as a firewall: access control between interfaces is controlled, and all of the usual firewall checks are in place. Previously, you could only configure bridge groups in transparent firewall mode, where you cannot route between bridge groups. This feature lets you configure bridge groups in routed firewall mode, and to route between bridge groups and between a bridge group and a routed interface. The bridge group participates in routing by using a Bridge Virtual Interface (BVI) to act as a gateway for the bridge group. Integrated Routing and Bridging provides an alternative to using an external Layer 2 switch if you have extra interfaces on the ASA to assign to the bridge group. In routed mode, the BVI can be a named interface and can participate separately from member interfaces in some features, such as access rules and DHCP server. The following features that are supported in transparent mode are not supported in routed mode: multiple context mode, ASA clustering. The following features are also not supported on BVIs: dynamic routing and multicast routing. We modified the following commands: **access-group, access-list ethertype, arp-inspection, dhcpd, mac-address-table static, mac-address-table aging-time, mac-learn, route, show arp-inspection, show bridge-group, show mac-address-table, show mac-learn**



PART V

IP Routing

- Routing Overview, on page 837
- Static and Default Routes, on page 851
- Policy Based Routing, on page 861
- Route Maps, on page 879
- Bidirectional Forwarding Detection Routing, on page 885
- BGP, on page 895
- OSPF, on page 937
- IS-IS, on page 999
- EIGRP, on page 1049
- Multicast Routing, on page 1071



CHAPTER 25

Routing Overview

This chapter describes how routing behaves within the ASA.

- Path Determination, on page 837
- Supported Route Types, on page 838
- Supported Internet Protocols for Routing, on page 839
- Routing Table, on page 840
- Routing Table for Management Traffic, on page 846
- Equal-Cost Multi-Path (ECMP) Routing, on page 848
- Disable Proxy ARP Requests, on page 848
- Display the Routing Table, on page 849
- History for Route Overview, on page 849

Path Determination

Routing protocols use metrics to evaluate what path will be the best for a packet to travel. A metric is a standard of measurement, such as path bandwidth, that is used by routing algorithms to determine the optimal path to a destination. To aid the process of path determination, routing algorithms initialize and maintain routing tables, which include route information. Route information varies depending on the routing algorithm used.

Routing algorithms fill routing tables with a variety of information. Destination or next hop associations tell a router that a particular destination can be reached optimally by sending the packet to a particular router representing the next hop on the way to the final destination. When a router receives an incoming packet, it checks the destination address and attempts to associate this address with a next hop.

Routing tables also can include other information, such as data about the desirability of a path. Routers compare metrics to determine optimal routes, and these metrics differ depending on the design of the routing algorithm used.

Routers communicate with one another and maintain their routing tables through the transmission of a variety of messages. The routing update message is one such message that generally consists of all or a portion of a routing table. By analyzing routing updates from all other routers, a router can build a detailed picture of network topology. A link-state advertisement, another example of a message sent between routers, informs other routers of the state of the sender links. Link information also can be used to build a complete picture of network topology to enable routers to determine optimal routes to network destinations.



Note Asymmetric routing is only supported for Active/Active failover in multiple context mode.

Supported Route Types

There are several route types that a router can use. The ASA uses the following route types:

- Static Versus Dynamic
- Single-Path Versus Multipath
- Flat Versus Hierarchical
- Link-State Versus Distance Vector

Static Versus Dynamic

Static routing algorithms are actually table mappings established by the network administrator. These mappings do not change unless the network administrator alters them. Algorithms that use static routes are simple to design and work well in environments where network traffic is relatively predictable and where network design is relatively simple.

Because static routing systems cannot react to network changes, they generally are considered unsuitable for large, constantly changing networks. Most of the dominant routing algorithms are dynamic routing algorithms, which adjust to changing network circumstances by analyzing incoming routing update messages. If the message indicates that a network change has occurred, the routing software recalculates routes and sends out new routing update messages. These messages permeate the network, stimulating routers to rerun their algorithms and change their routing tables accordingly.

Dynamic routing algorithms can be supplemented with static routes where appropriate. A router of last resort (a default route for a router to which all unrouteable packets are sent), for example, can be designated to act as a repository for all unrouteable packets, ensuring that all messages are at least handled in some way.

Single-Path Versus Multipath

Some sophisticated routing protocols support multiple paths to the same destination. Unlike single-path algorithms, these multipath algorithms permit traffic multiplexing over multiple lines. The advantages of multipath algorithms are substantially better throughput and reliability, which is generally called load sharing.

Flat Versus Hierarchical

Some routing algorithms operate in a flat space, while others use routing hierarchies. In a flat routing system, the routers are peers of all others. In a hierarchical routing system, some routers form what amounts to a routing backbone. Packets from non-backbone routers travel to the backbone routers, where they are sent through the backbone until they reach the general area of the destination. At this point, they travel from the last backbone router through one or more non-backbone routers to the final destination.

Routing systems often designate logical groups of nodes, called domains, autonomous systems, or areas. In hierarchical systems, some routers in a domain can communicate with routers in other domains, while others

can communicate only with routers within their domain. In very large networks, additional hierarchical levels may exist, with routers at the highest hierarchical level forming the routing backbone.

The primary advantage of hierarchical routing is that it mimics the organization of most companies and therefore supports their traffic patterns well. Most network communication occurs within small company groups (domains). Because intradomain routers need to know only about other routers within their domain, their routing algorithms can be simplified, and, depending on the routing algorithm being used, routing update traffic can be reduced accordingly.

Link-State Versus Distance Vector

Link-state algorithms (also known as shortest path first algorithms) flood routing information to all nodes in the internetwork. Each router, however, sends only the portion of the routing table that describes the state of its own links. In link-state algorithms, each router builds a picture of the entire network in its routing tables. Distance vector algorithms (also known as Bellman-Ford algorithms) call for each router to send all or some portion of its routing table, but only to its neighbors. In essence, link-state algorithms send small updates everywhere, while distance vector algorithms send larger updates only to neighboring routers. Distance vector algorithms know only about their neighbors. Typically, link-state algorithms are used in conjunction with OSPF routing protocols.

Supported Internet Protocols for Routing

The ASA supports several Internet protocols for routing. Each protocol is briefly described in this section.

- Enhanced Interior Gateway Routing Protocol (EIGRP)

  EIGRP is a Cisco proprietary protocol that provides compatibility and seamless interoperation with IGRP routers. An automatic-redistribution mechanism allows IGRP routes to be imported into Enhanced IGRP, and vice versa, so it is possible to add Enhanced IGRP gradually into an existing IGRP network.

- Open Shortest Path First (OSPF)

  OSPF is a routing protocol developed for Internet Protocol (IP) networks by the interior gateway protocol (IGP) working group of the Internet Engineering Task Force (IETF). OSPF uses a link-state algorithm to build and calculate the shortest path to all known destinations. Each router in an OSPF area includes an identical link-state database, which is a list of each of the router usable interfaces and reachable neighbors.

- Routing Information Protocol (RIP)

  RIP is a distance-vector protocol that uses hop count as its metric. RIP is widely used for routing traffic in the global Internet and is an interior gateway protocol (IGP), which means that it performs routing within a single autonomous system.

- Border Gateway Protocol (BGP)

  BGP is an interautonomous system routing protocol. BGP is used to exchange routing information for the Internet and is the protocol used between Internet service providers (ISP). Customers connect to ISPs, and ISPs use BGP to exchange customer and ISP routes. When BGP is used between autonomous systems (AS), the protocol is referred to as External BGP (EBGP). If a service provider is using BGP to exchange routes within an AS, then the protocol is referred to as Interior BGP (IBGP).

- Intermediate System to Intermediate System (IS-IS)

IS-IS is a link state Interior Gateway Protocol (IGP). Link-state protocols are characterized by the propagation of the information required to build a complete network connectivity map on each participating router. That map is then used to calculate the shortest path to destinations.

Routing GRE Traffic

In ASA, Generic Routing Encapsulation (GRE) traffic is managed using bi-directional flows. This means that each GRE session is represented by a single flow entry that handles both inbound and outbound traffic between endpoints. With bi-directional flows, only one route lookup is performed and both directions are mapped to the same interface. As a result, the system cannot make independent routing decisions for each direction, which may lead to sub-optimal routing in some scenarios.

Routing Table

The ASA uses separate routing tables for data traffic (through-the-device) and for management traffic (from-the-device). This section describes how the routing tables work. For information about the management routing table, see also Routing Table for Management Traffic, on page 846.

How the Routing Table Is Populated

The ASA routing table can be populated by statically defined routes, directly connected routes, and routes discovered by the dynamic routing protocols. Because the ASA device can run multiple routing protocols in addition to having static and connected routes in the routing table, it is possible that the same route is discovered or entered in more than one manner. When two routes to the same destination are put into the routing table, the one that remains in the routing table is determined as follows:

- If the two routes have different network prefix lengths (network masks), then both routes are considered unique and are entered into the routing table. The packet forwarding logic then determines which of the two to use.

  For example, if the RIP and OSPF processes discovered the following routes:

  - RIP: 192.168.32.0/24
  - OSPF: 192.168.32.0/19

  Even though OSPF routes have the better administrative distance, both routes are installed in the routing table because each of these routes has a different prefix length (subnet mask). They are considered different destinations and the packet forwarding logic determines which route to use.

- If the ASA device learns about multiple paths to the same destination from a single routing protocol, such as RIP, the route with the better metric (as determined by the routing protocol) is entered into the routing table.

  Metrics are values associated with specific routes, ranking them from most preferred to least preferred. The parameters used to determine the metrics differ for different routing protocols. The path with the lowest metric is selected as the optimal path and installed in the routing table. If there are multiple paths to the same destination with equal metrics, load balancing is done on these equal cost paths.

- If the ASA device learns about a destination from more than one routing protocol, the administrative distances of the routes are compared, and the routes with lower administrative distance are entered into the routing table.

Administrative Distances for Routes

You can change the administrative distances for routes discovered by or redistributed into a routing protocol. If two routes from two different routing protocols have the same administrative distance, then the route with the lower *default* administrative distance is entered into the routing table. In the case of EIGRP and OSPF routes, if the EIGRP route and the OSPF route have the same administrative distance, then the EIGRP route is chosen by default.

Administrative distance is a route parameter that the ASA uses to select the best path when there are two or more different routes to the same destination from two different routing protocols. Because the routing protocols have metrics based on algorithms that are different from the other protocols, it is not always possible to determine the best path for two routes to the same destination that were generated by different routing protocols.

Each routing protocol is prioritized using an administrative distance value. The following table shows the default administrative distance values for the routing protocols supported by the ASA.

Table 31: Default Administrative Distance for Supported Routing Protocols

Route Source	Default Administrative Distance
Connected interface	0
VPN route	1
Static route	1
EIGRP Summary Route	5
External BGP	20
Internal EIGRP	90
OSPF	110
IS-IS	115
RIP	120
EIGRP external route	170
Internal and local BGP	200
Unknown	255

The smaller the administrative distance value, the more preference is given to the protocol. For example, if the ASA receives a route to a certain network from both an OSPF routing process (default administrative distance - 110) and a RIP routing process (default administrative distance - 120), the ASA chooses the OSPF route because OSPF has a higher preference. In this case, the router adds the OSPF version of the route to the routing table.

A VPN advertised route (V-Route/RRI)) is equivalent to a static route with the default administrative distance 1. But it has a higher preference as with the network mask 255.255.255.255.

In this example, if the source of the OSPF-derived route was lost (for example, due to a power shutdown), the ASA would then use the RIP-derived route until the OSPF-derived route reappears.

The administrative distance is a local setting. For example, if you change the administrative distance of routes obtained through OSPF, that change would only affect the routing table for the ASA on which the command was entered. The administrative distance is not advertised in routing updates.

Administrative distance does not affect the routing process. The routing processes only advertise the routes that have been discovered by the routing process or redistributed into the routing process. For example, the RIP routing process advertises RIP routes, even if routes discovered by the OSPF routing process are used in the routing table.

Backup Dynamic and Floating Static Routes

A backup route is registered when the initial attempt to install the route in the routing table fails because another route was installed instead. If the route that was installed in the routing table fails, the routing table maintenance process calls each routing protocol process that has registered a backup route and requests them to reinstall the route in the routing table. If there are multiple protocols with registered backup routes for the failed route, the preferred route is chosen based on administrative distance.

Because of this process, you can create floating static routes that are installed in the routing table when the route discovered by a dynamic routing protocol fails. A floating static route is simply a static route configured with a greater administrative distance than the dynamic routing protocols running on the ASA. When the corresponding route discovered by a dynamic routing process fails, the static route is installed in the routing table.

How Forwarding Decisions Are Made

Forwarding decisions are made as follows:

- If the destination does not match an entry in the routing table, the packet is forwarded through the interface specified for the default route. If a default route has not been configured, the packet is discarded.
- If the destination matches a single entry in the routing table, the packet is forwarded through the interface associated with that route.
- If the destination matches more than one entry in the routing table, then the packet is forwarded out of the interface associated with the route that has the longer network prefix length.

For example, a packet destined for 192.168.32.1 arrives on an interface with the following routes in the routing table:

- 192.168.32.0/24 gateway 10.1.1.2
- 192.168.32.0/19 gateway 10.1.1.3

In this case, a packet destined to 192.168.32.1 is directed toward 10.1.1.2, because 192.168.32.1 falls within the 192.168.32.0/24 network. It also falls within the other route in the routing table, but 192.168.32.0/24 has the longest prefix within the routing table (24 bits verses 19 bits). Longer prefixes are always preferred over shorter ones when forwarding a packet.

Note Existing connections continue to use their established interfaces even if a new similar connection would result in different behavior due to a change in routes.

Dynamic Routing and Failover

Dynamic routes are synchronized on the standby unit when the routing table changes on the active unit. This means that all additions, deletions, or changes on the active unit are immediately propagated to the standby unit. If the standby unit becomes active in an active/standby ready Failover pair, it will already have an identical routing table as that of the former active unit because routes are synchronized as a part of the Failover bulk synchronization and continuous replication processes.

Dynamic Routing and Clustering

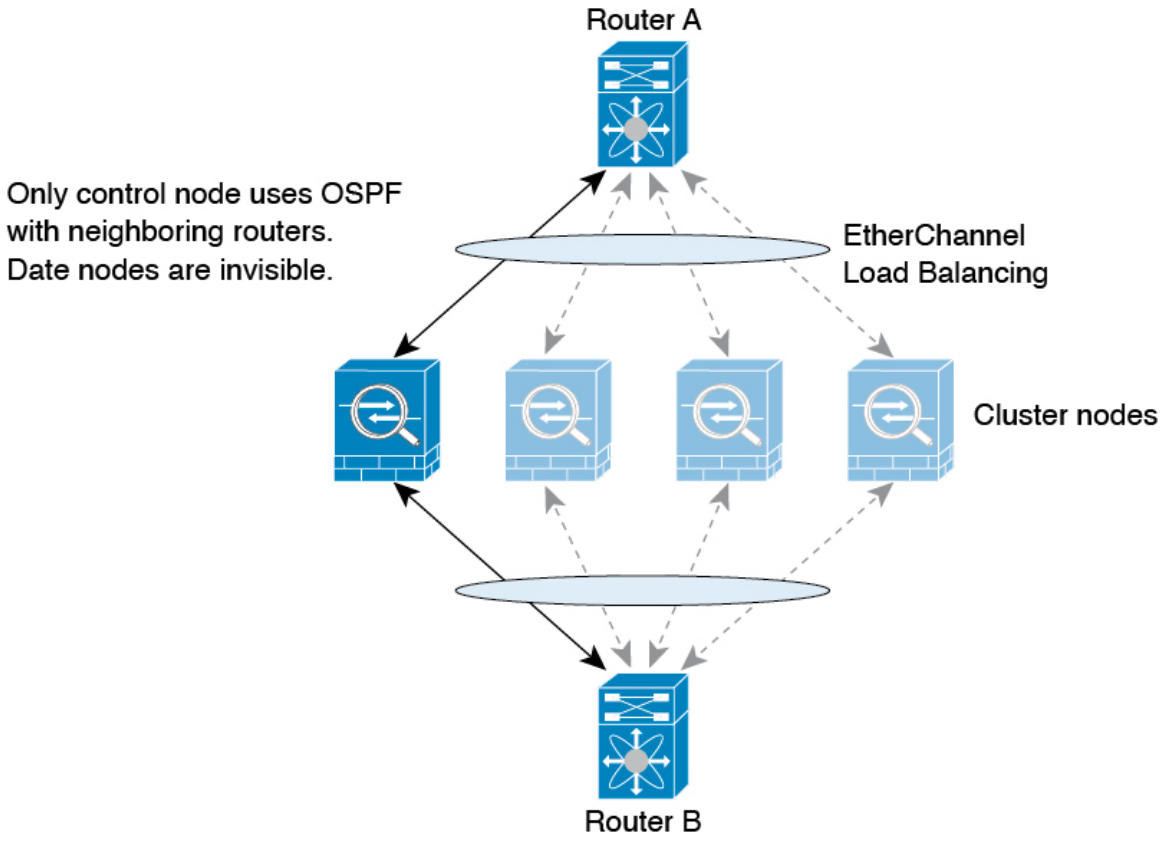
This section describes how to use dynamic routing with clustering.

Dynamic Routing in Spanned EtherChannel Mode

Note IS-IS is not supported in Spanned EtherChannel mode.

The routing process only runs on the control node, and routes are learned through the control node and replicated to data nodes. If a routing packet arrives at a data node, it is redirected to the control node.

Figure 56: Dynamic Routing in Spanned EtherChannel Mode



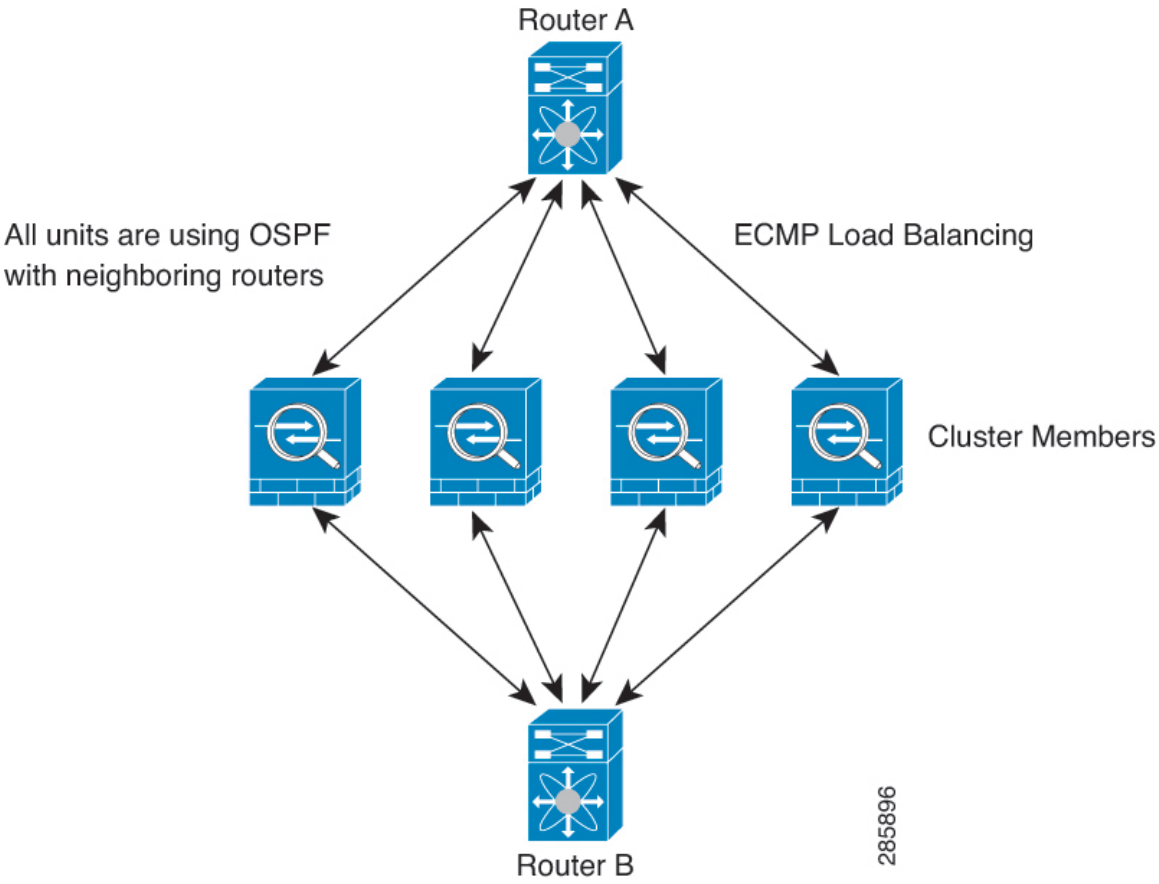
After the data node learn the routes from the control node, each node makes forwarding decisions independently.

The OSPF LSA database is not synchronized from the control node to data nodes. If there is a control node switchover, the neighboring router will detect a restart; the switchover is not transparent. The OSPF process picks an IP address as its router ID. Although not required, you can assign a static router ID to ensure a consistent router ID is used across the cluster. See the OSPF Non-Stop Forwarding feature to address the interruption.

Dynamic Routing in Individual Interface Mode

In Individual interface mode, each node runs the routing protocol as a standalone router, and routes are learned by each node independently.

Figure 57: Dynamic Routing in Individual Interface Mode



In the above diagram, Router A learns that there are 4 equal-cost paths to Router B, each through a node. ECMP is used to load balance traffic between the 4 paths. Each node picks a different router ID when talking to external routers.

You must configure a cluster pool for the router ID so that each node has a separate router ID.

EIGRP does not form neighbor relationships with cluster peers in individual interface mode.

Note If the cluster has multiple adjacencies to the same router for redundancy purposes, asymmetric routing can lead to unacceptable traffic loss. To avoid asymmetric routing, group all of these node interfaces into the same traffic zone. See Configure a Traffic Zone, on page 726.

Dynamic Routing in Multiple Context Mode

In multiple context mode, each context maintains a separate routing table and routing protocol databases. This enables you to configure OSPFv2 and EIGRP independently in each context. You can configure EIGRP in some contexts and OSPFv2 in the same or different contexts. In mixed context mode, you can enable any of the dynamic routing protocols in contexts that are in routed mode. RIP and OSPFv3 are not supported in multiple context mode.

The following table lists the attributes for EIGRP, OSPFv2, route maps used for distributing routes into OSPFv2 and EIGRP processes, and prefix lists used in OSPFv2 to filter the routing updates entering or leaving an area when they are used in multiple context mode:

EIGRP	OSPFv2	Route Maps and Prefix Lists
One instance is supported per context.	Two instances are supported per context.	N/A
It is disabled in the system context.		N/A
Two contexts may use the same or different autonomous system numbers.	Two contexts may use the same or different area IDs.	N/A
Shared interfaces in two contexts may have multiple EIGRP instances running on them.	Shared interfaces in two contexts may have multiple OSPF instances running on them.	N/A
The interaction of EIGRP instances across shared interfaces is supported.	The interaction of OSPFv2 instances across shared interfaces is supported.	N/A
All CLIs that are available in single mode are also available in multiple context mode.		
Each CLI has an effect only in the context in which it is used.		

Route Resource Management

A resource class called *routes* specifies the maximum number of routing table entries that can exist in a context. This resolves the problem of one context affecting the available routing table entries in another context and also allows you greater control over the maximum route entries per context.

Because there is no definitive system limit, you can only specify an absolute value for this resource limit; you may not use a percentage limit. Also, there are no minimum and maximum limits per context, so the default class does not change. If you add a new route for any of the static or dynamic routing protocols (connected, static, OSPF, EIGRP, and RIP) in a context and the resource limit for that context is exhausted, then the route addition fails and a syslog message is generated.

Routing Table for Management Traffic

As a standard security practice, it is often necessary to segregate and isolate management (from-the-device) traffic from data traffic. To achieve this isolation, the ASA device uses a separate routing table for management-only traffic vs. data traffic. Separate routing tables means that you can create separate default routes for data and management as well.

Types of Traffic for Each Routing Table

Through-the-device traffic always uses the data routing table.

From-the-device traffic, depending on the type, uses either the management-only routing table or the data routing table by default. If a match is not found in the default routing table, it checks the other routing table.

- Management-only table from-the-device traffic includes features that open a remote file using HTTP, SCP, TFTP, the **copy** command, Smart Licensing, Smart Call Home, **trustpoint**, **trustpool**, and so on.
- Data table from-the-device traffic includes all other features like ping, DNS, DHCP, and so on.

Interfaces Included in the Management-Only Routing Table

Management-only interfaces include any the Management x/x interfaces as well as any interfaces that you have configured to be management-only.

Fallback to the Other Routing Table

If a match is not found in the default routing table, it checks the other routing table.

Using the Non-Default Routing Table

If you need from-the-box traffic to go out an interface that isn't in its default routing table, then you might need to specify that interface when you configure it, rather than relying on the fall back to the other table. The ASA will only check routes for the specified interface. For example, if you need a ping to go out a management-only interface, then specify the interface in the ping function. Otherwise, if there is a default route in the data routing table, then it will match the default route and never fall back to the management routing table.

Dynamic Routing

The management-only routing table supports dynamic routing separate from the data interface routing table. A given dynamic routing process must run on either the management-only interface or the data interface; you cannot mix both types. When upgrading from an earlier release without the separate management routing table, if you have a mix of data and management interfaces using the same dynamic routing process, management interfaces will be dropped.

Management-Access Feature for VPN Requirements

If you configure the management-access feature that allows management access to an interface other than the one from which you entered the ASA when using VPN, then due to routing considerations with the separate management and data routing tables, the VPN termination interface and the management access interface need to be the same type: both need to be management-only interfaces or regular data interfaces.

Management Interface Identification

An interface configured with management-only is considered a management interface.

In the following configuration, both the interfaces GigabitEthernet0/0 and Management0/0 are considered as management interfaces.

```
a/admin(config-if)# show running-config int g0/0
!
interface GigabitEthernet0/0
 management-only
 nameif inside
 security-level 100
 ip address 10.10.10.123 255.255.255.0
 ipv6 address 123::123/64
a/admin(config-if)# show running-config int m0/0
!
interface Management0/0
 management-only
 nameif mgmt
 security-level 0
```

```
 ip address 10.106.167.118 255.255.255.0
a/admin(config-if)#
```

Equal-Cost Multi-Path (ECMP) Routing

The ASA supports Equal-Cost Multi-Path (ECMP) routing.

You can have up to 8 equal cost static or dynamic routes per interface. For example, you can configure multiple default routes on the outside interface that specify different gateways.

```
route outside 0 0 10.1.1.2
route outside 0 0 10.1.1.3
route outside 0 0 10.1.1.4
```

In this case, traffic is load-balanced on the outside interface between 10.1.1.2, 10.1.1.3, and 10.1.1.4. Traffic is distributed among the specified gateways based on an algorithm that hashes the source and destination IP addresses, incoming interface, protocol, source and destination ports.

ECMP Across Multiple Interfaces Using Traffic Zones

If you configure traffic zones to contain a group of interfaces, you can have up to 8 equal cost static or dynamic routes across up to 8 interfaces within each zone. For example, you can configure multiple default routes across three interfaces in the zone:

```
route outside1 0 0 10.1.1.2
route outside2 0 0 10.2.1.2
route outside3 0 0 10.3.1.2
```

Similarly, your dynamic routing protocol can automatically configure equal cost routes. The ASA load-balances traffic across the interfaces with a more robust load balancing mechanism.

When a route is lost, the device seamlessly moves the flow to a different route.

Disable Proxy ARP Requests

When a host sends IP traffic to another device on the same Ethernet network, the host needs to know the MAC address of the device. ARP is a Layer 2 protocol that resolves an IP address to a MAC address. A host sends an ARP request asking “Who is this IP address?” The device owning the IP address replies, “I own that IP address; here is my MAC address.”

Proxy ARP is used when a device responds to an ARP request with its own MAC address, even though the device does not own the IP address. The ASA uses proxy ARP when you configure NAT and specify a mapped address that is on the same network as the ASA interface. The only way traffic can reach the hosts is if the ASA uses proxy ARP to claim that the MAC address is assigned to destination mapped addresses.

Under rare circumstances, you might want to disable proxy ARP for NAT addresses.

If you have a VPN client address pool that overlaps with an existing network, the ASA by default sends proxy ARP requests on all interfaces. If you have another interface that is on the same Layer 2 domain, it will see the ARP requests and will answer with the MAC address of its interface. The result of this is that the return traffic of the VPN clients towards the internal hosts will go to the wrong interface and will get dropped. In this case, you need to disable proxy ARP requests for the interface on which you do not want them.

Procedure

Disable proxy ARP requests:

sysopt noproxyarp *interface*

Example:

```
ciscoasa(config)# sysopt noproxyarp exampleinterface
```

Display the Routing Table

Use the **show route** command to view the entries in the routing table.

```
ciscoasa# show route

Codes: C - connected, S - static, I - IGRP, 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, E - EGP
       i - IS-IS, L1 - IS-IS level-1, L2 - IS-IS level-2, ia - IS-IS inter area
       * - candidate default, U - per-user static route, o - ODR
       P - periodic downloaded static route

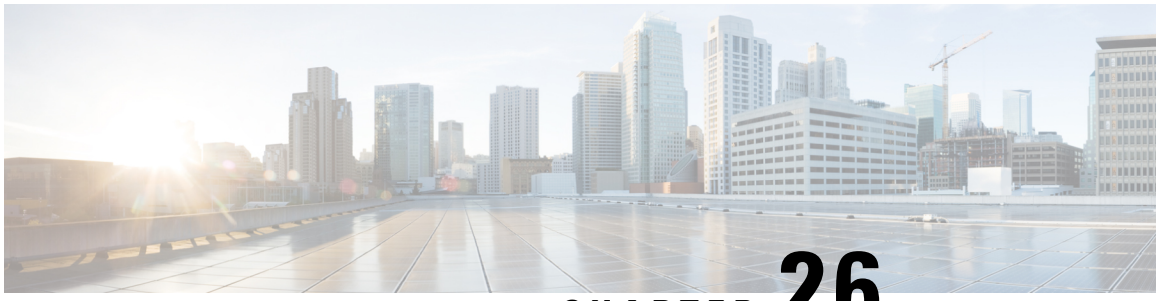
Gateway of last resort is 10.86.194.1 to network 0.0.0.0

S    10.1.1.0 255.255.255.0 [3/0] via 10.86.194.1, outside
C    10.86.194.0 255.255.254.0 is directly connected, outside
S*   0.0.0.0 0.0.0.0 [1/0] via 10.86.194.1, outside
```

History for Route Overview

Table 32: History for Route Overview

Feature Name	Platform Releases	Feature Information
Routing Table for Management Interface	9.5(1)	To segregate and isolate, management traffic from data traffic, a sepa table is added for management traffic. Separate routing tables, for ma data respectively, are created for both IPv4 and IPv6, for each contex Further, for each context of the ASA, two extra routing tables are add RIB and FIB. We introduced the following commands: show route management-on route management-only, show asp table route-management- only, cle management-only, clear ipv6 route management-only, copy interface tftp/ftp



CHAPTER 26

Static and Default Routes

This chapter describes how to configure static and default routes on the ASA.

- About Static and Default Routes, on page 851
- Guidelines for Static and Default Routes, on page 853
- Configure Default and Static Routes, on page 854
- Monitoring a Static or Default Route, on page 858
- Examples for Static or Default Routes, on page 858
- History for Static and Default Routes, on page 858

About Static and Default Routes

To route traffic to a non-connected host or network, you must define a route to the host or network, either using static or dynamic routing. Generally, you must configure at least one static route: a default route for all traffic that is not routed by other means to a default network gateway, typically the next hop router.

Default Route

The simplest option is to configure a default static route to send all traffic to an upstream router, relying on the router to route the traffic for you. A default route identifies the gateway IP address to which the ASA sends all IP packets for which it does not have a learned or static route. A default static route is simply a static route with 0.0.0.0/0 (IPv4) or ::/0 (IPv6) as the destination IP address.

You should always define a default route.

Because the ASA device uses separate routing tables for data traffic and for management traffic, you can optionally configure a default route for data traffic and another default route for management traffic. Note that from-the-device traffic uses either the management-only or data routing table by default depending on the type, but will fall back to the other routing table if a route is not found. Default routes will always match traffic, and will prevent a fall back to the other routing table. In this case, you must specify the interface you want to use for egress traffic if that interface is not in the default routing table.

Static Routes

You might want to use static routes in the following cases:

- Your networks use an unsupported router discovery protocol.

- Your network is small and you can easily manage static routes.
- You do not want the traffic or CPU overhead associated with routing protocols.
- In some cases, a default route is not enough. The default gateway might not be able to reach the destination network, so you must also configure more specific static routes. For example, if the default gateway is outside, then the default route cannot direct traffic to any inside networks that are not directly connected to the ASA.
- You are using a feature that does not support dynamic routing protocols.

Route to null0 Interface to Drop Unwanted Traffic

Access rules let you filter packets based on the information contained in their headers. A static route to the null0 interface is a complementary solution to access rules. You can use a null0 route to forward unwanted or undesirable traffic so the traffic is dropped.

Static null0 routes have a favorable performance profile. You can also use static null0 routes to prevent routing loops. BGP can leverage the static null0 route for Remotely Triggered Black Hole routing.

Route Priorities

- Routes that identify a specific destination take precedence over the default route.
- When multiple routes exist to the same destination (either static or dynamic), then the administrative distance for the route determines priority. Static routes are set to 1, so they typically are the highest priority routes.
- When you have multiple static routes to the same destination with the same administrative distance, see Equal-Cost Multi-Path (ECMP) Routing, on page 848.
- For traffic emerging from a tunnel with the Tunneled option, this route overrides any other configured or learned default routes.

Transparent Firewall Mode and Bridge Group Routes

For traffic that originates on the ASA and is destined through a bridge group member interface for a non-directly connected network, you need to configure either a default route or static routes so the ASA knows out of which bridge group member interface to send traffic. Traffic that originates on the ASA might include communications to a syslog server or SNMP server. If you have servers that cannot all be reached through a single default route, then you must configure static routes. For transparent mode, you cannot specify the BVI as the gateway interface; only member interfaces can be used. For bridge groups in routed mode, you must specify the BVI in a static route; you cannot specify a member interface. See MAC address and route lookups, on page 190 for more information.

Static Route Tracking

One of the problems with static routes is that there is no inherent mechanism for determining if the route is up or down. They remain in the routing table even if the next hop gateway becomes unavailable. Static routes are only removed from the routing table if the associated interface on the ASA goes down.

The static route tracking feature provides a method for tracking the availability of a static route and installing a backup route if the primary route should fail. For example, you can define a default route to an ISP gateway and a backup default route to a secondary ISP in case the primary ISP becomes unavailable.

The ASA implements static route tracking by associating a static route with a monitoring target host on the destination network that the ASA monitors using ICMP echo requests. If an echo reply is not received within a specified time period, the host is considered down, and the associated route is removed from the routing table. An untracked backup route with a higher metric is used in place of the removed route.

When selecting a monitoring target, you need to make sure that it can respond to ICMP echo requests. The target can be any network object that you choose, but you should consider using the following:

- The ISP gateway (for dual ISP support) address
- The next hop gateway address (if you are concerned about the availability of the gateway)
- A server on the target network, such as a syslog server, that the ASA needs to communicate with
- A persistent network object on the destination network



Note A PC that may be shut down at night is not a good choice.

You can configure static route tracking for statically defined routes or default routes obtained through DHCP or PPPoE. You can only enable PPPoE clients on multiple interfaces with route tracking configured.

Guidelines for Static and Default Routes

Firewall Mode and Bridge Groups

- In transparent mode, static routes must use the bridge group member interface as the gateway; you cannot specify the BVI.
- In routed mode, you must specify the BVI as the gateway; you cannot specify the member interface.
- Static route tracking is not supported for bridge group member interfaces or on the BVI.

Supported Network Address

- Static route tracking is not supported for IPv6.
- The ASA does not support Class E routing, so a Class E network is not routable in static routes.

Clustering and Multiple Context Mode

- In clustering, static route tracking is only supported on the control node.
- Static route tracking is not supported in multiple context mode.

ASP and RIB Route Entries

All routes and its distance installed on the device are captured in the ASP routing table. This is common for all static and dynamic routing protocols. Only the best distance route is captured in the RIB table.

Configure Default and Static Routes

At a minimum, you should configure a default route. You may need to configure static routes as well. In this section we will configure a default route, configure a static route and track a static route.

Configure a Default Route

A default route is simply a static route with 0.0.0.0/0 as the destination IP address. You should always have a default route, either configured manually with this procedure, or derived from a DHCP server or other routing protocol.

Before you begin

See the following guidelines for the Tunneled option:

- Do not enable unicast RPF (**ip verify reverse-path** command) on the egress interface of a tunneled route, because this setting causes the session to fail.
- Do not enable TCP intercept on the egress interface of the tunneled route, because this setting causes the session to fail.
- Do not use the VoIP inspection engines (CTIQBE, H.323, GTP, MGCP, RTSP, SIP, SKINNY), the DNS inspection engine, or the DCE RPC inspection engine with tunneled routes, because these inspection engines ignore the tunneled route.
- You cannot define more than one default route with the tunneled option.
- ECMP for tunneled traffic is not supported.
- Tunneled routes are not supported for bridge groups, which do not support VPN termination for through traffic.

Procedure

Add a default route:

IPv4:

route *if_name* **0.0.0.0 0.0.0.0** *gateway_ip* [**distance**] [**tunneled**]

IPv6:

ipv6 route *if_name* **::/0** *gateway_ip* [**distance**] [**tunneled**]

Example:

```
ciscoasa(config)# route outside 0.0.0.0 0.0.0.0 192.168.2.4
ciscoasa(config)# route inside 0.0.0.0 0.0.0.0 10.1.2.3 tunneled
```

```
ciscoasa(config)# ipv6 route inside ::/0 3FFE:1100:0:CC00::1
```

The *if_name* is the interface through which you want to send the specific traffic. For transparent mode, specify a bridge group member interface name. For routed mode with bridge groups, specify the BVI name.

The *distance* argument is the administrative distance for the route, between 1 and 254. The default is **1** if you do not specify a value. Administrative distance is a parameter used to compare routes among different routing protocols. The default administrative distance for static routes is 1, giving it precedence over routes discovered by dynamic routing protocols but not directly connected routes. The default administrative distance for routes discovered by OSPF is 110. If a static route has the same administrative distance as a dynamic route, the static routes take precedence. Connected routes always take precedence over static or dynamically discovered routes.

Note
For through-the-box traffic, if you have two default routes configured on different interfaces that have different metrics, the connection to the ASA that is made from the higher metric interface fails, but connections to the ASA from the lower metric interface succeed as expected. For from-the-box traffic, if you have two default routes configured on different interfaces that have different metrics, both interfaces might be used for from-the-box traffic depending on which interface was used for the incoming connection.

You can define a separate default route for VPN traffic if you want your VPN traffic to use a different default route than your non VPN traffic using the **tunneled** keyword. For example, traffic incoming from VPN connections can be easily directed towards internal networks, while traffic from internal networks can be directed towards the outside. When you create a default route with the tunneled option, all traffic from a tunnel terminating on the ASA that cannot be routed using learned or static routes, is sent to this route. This option is not supported for bridge groups.

Tip
You can enter **0 0** instead of **0.0.0.0 0.0.0.0** for the destination network address and mask, as shown in the following example: **route outside 0 0 192.168.2.4**

Configure a Static Route

A static route defines where to send traffic for specific destination networks.

Procedure

Add a static route:

IPv4:

route *if_name dest_ip mask gateway_ip* [**distance**]

IPv6:

ipv6 route *if_name dest_ipv6_prefix/prefix_length gateway_ip* [**distance**]

Example:

```
ciscoasa(config)# route outside 10.10.10.0 255.255.255.0 192.168.1.1
ciscoasa(config)# ipv6 route outside 2001:DB8:1::0/32 2001:DB8:0:CC00::1
```

The *if_name* is the interface through which you want to send the specific traffic. To drop unwanted traffic, enter the **null0** interface. For transparent mode, specify a bridge group member interface name. For routed mode with bridge groups, specify the BVI name.

The *dest_ip* and *mask* or *dest_ipv6_prefix*/*prefix_length* arguments indicate the IP address for the destination network and the *gateway_ip* argument is the address of the next-hop router. The addresses you specify for the static route are the addresses that are in the packet before entering the ASA and performing NAT.

The *distance* argument is the administrative distance for the route. The default is **1** if you do not specify a value. Administrative distance is a parameter used to compare routes among different routing protocols. The default administrative distance for static routes is 1, giving it precedence over routes discovered by dynamic routing protocols but not directly connected routes. The default administrative distance for routes discovered by OSPF is 110. If a static route has the same administrative distance as a dynamic route, the static route takes precedence. Connected routes always take precedence over static or dynamically discovered routes.

Example

The following example shows static routes for 3 networks that go to the same gateway, and another network that goes to a separate gateway:

```
route outside 10.10.10.0 255.255.255.0 192.168.1.1
route outside 10.10.20.0 255.255.255.0 192.168.1.1
route outside 10.10.30.0 255.255.255.0 192.168.1.1
route inside 10.10.40.0 255.255.255.0 10.1.1.1
```

Configure Static Route Tracking

To configure static route tracking, complete the following steps.

Procedure

Step 1 Configure static route for the interface (outside 1) used to reach the network:

```
ciscoasa(config)# route outside1 172.29.139.134 255.255.255.255 10.0.0.1
```

Step 2 Define the monitoring process:

sla monitor sla_id

Example:

```
ciscoasa(config)# sla monitor 5
ciscoasa(config-sla-monitor)#
```

Step 3 Specify the monitoring protocol, the target host on the tracked network, and the interface through which you reach the network:

type echo protocol ipicmpecho target_ip **interface** if_name

Example:

```
ciscoasa(config-sla-monitor)# type echo protocol ipicmpecho 172.29.139.134 interface outside1
ciscoasa(config-sla-monitor-echo)#
```

The *target_ip* argument is the IP address of the network object whose availability the tracking process monitors. While this object is available, the tracking process route is installed in the routing table. When this object becomes unavailable, the tracking process removes the route and the backup route is used in its place.

Step 4 (Optional) Configure monitoring options. See the command reference for the following commands: **frequency, num-packets, request-data-size, threshold, timeout,** and **tos**.

Step 5 Schedule the monitoring process:

sla monitor schedule sla_id [**life** {**forever** | seconds}] [**start-time** {hh:mm [:ss] [month day | day month] | **pending** | **now** | **after** hh:mm:ss}] [**ageout** seconds] [**recurring**]

Example:

```
ciscoasa(config)# sla monitor schedule 5 life forever start-time now
```

Typically, you will use the **sla monitor schedule** *sla_id* **life forever start-time now** command for the monitoring schedule, and allow the monitoring configuration to determine how often the testing occurs.

However, you can schedule this monitoring process to begin in the future and to only occur at specified times.

Step 6 Associate a tracked static route with the SLA monitoring process:

track track_id **rtr** sla_id **reachability**

Example:

```
ciscoasa(config)# track 6 rtr 5 reachability
```

The *track_id* argument is a tracking number you assign with this command. The *sla_id* argument is the ID number of the SLA process.

Step 7 Track one of the following route types:

- Static route:

  route if_name dest_ip mask gateway_ip [distance] **track** track_id

  Example:

  ```
  ciscoasa(config)# route outside 10.10.10.0 255.255.255.0 192.168.1.1 track 6
  ```

  You cannot use the **tunneled** option.

- Default route obtained through DHCP:

  ```
  interface interface_id
    dhcp client route track track_id
    ip address dhcp setroute
  ```

- Default route obtained through PPPoE:

```
interface interface_id
  pppoe client route track track_id
  ip address pppoe setroute
```

Step 8 Create an untracked backup route.

The backup route is a static route to the same destination as the tracked route, but through a different interface or gateway. You must assign this route a higher administrative distance (metric) than your tracked route.

Monitoring a Static or Default Route

- **show route**

  Displays the routing table.

Examples for Static or Default Routes

The following example shows how to create a static route that sends all traffic destined for 10.1.1.0/24 to the router 10.1.2.45, which is connected to the inside interface, defines three equal cost static routes that direct traffic to three different gateways on the dmz interface, and adds a default route for tunneled traffic and one for regular traffic.

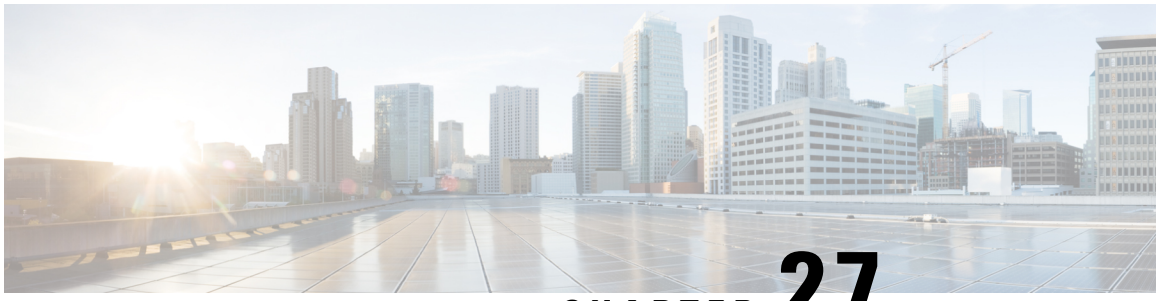
```
route inside 10.1.1.0 255.255.255.0 10.1.2.45
route dmz 10.10.10.0 255.255.255.0 192.168.2.1
route dmz 10.10.10.0 255.255.255.0 192.168.2.2
route dmz 10.10.10.0 255.255.255.0 192.168.2.3
route outside 0 0 209.165.201.1
route inside 0 0 10.1.2.45 tunneled
```

History for Static and Default Routes

Table 33: Feature History for Static and Default Routes

Feature Name	Platform Releases	Feature Information
Static Route Tracking	7.2(1)	The static route tracking feature provides a method for tracking the availability of a static route and installing a backup route if the primary route should fail. We introduced the following commands: **clear configure sla, frequency, num-packets, request-data-size, show sla monitor, show running-config sla, sla monitor, sla monitor schedule, threshold, timeout, tos, track rtr**

Feature Name	Platform Releases	Feature Information
Static null0 route to drop traffic	9.2(1)	Sending traffic to a null0 interface results in dropping the packets destined to the specified network. This feature is useful in configuring Remotely Triggered Black Hole (RTBH) for BGP. We modified the following command: **route**.



CHAPTER 27

Policy Based Routing

This chapter describes how to configure the ASA to support policy based routing (PBR). The following sections describe policy based routing, guidelines for PBR, and configuration for PBR.

- About Policy Based Routing, on page 861
- Guidelines for Policy Based Routing, on page 863
- Configure Policy Based Routing, on page 864
- Examples for Policy Based Routing, on page 868
- History for Policy Based Routing, on page 877

About Policy Based Routing

Traditional routing is destination-based, meaning packets are routed based on destination IP address. However, it is difficult to change the routing of specific traffic in a destination-based routing system. With Policy Based Routing (PBR), you can define routing based on criteria other than destination network—PBR lets you route traffic based on source address, source port, destination address, destination port, protocol, or a combination of these.

Policy Based Routing:

- Lets you provide Quality of Service (QoS) to differentiated traffic.
- Lets you distribute interactive and batch traffic across low-bandwidth, low-cost permanent paths and high-bandwidth, high-cost switched paths.
- Allows Internet service providers and other organizations to route traffic originating from various sets of users through well-defined Internet connections.

Policy Based Routing can implement QoS by classifying and marking traffic at the network edge, and then using PBR throughout the network to route marked traffic along a specific path. This permits routing of packets originating from different sources to different networks, even when the destinations are the same, and it can be useful when interconnecting several private networks.

Why Use Policy Based Routing?

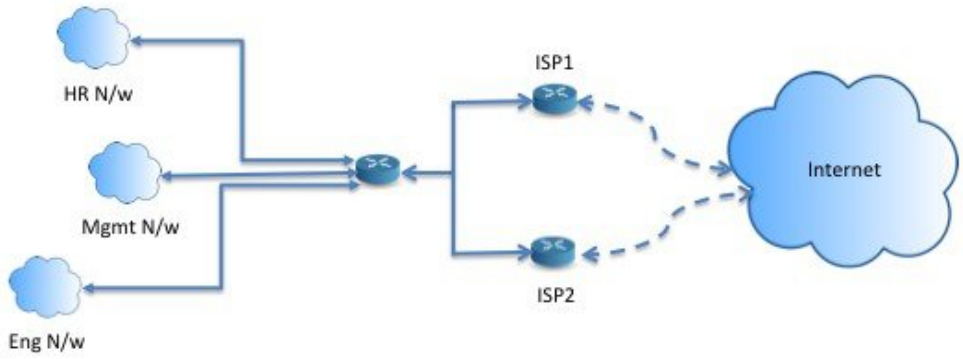
Consider a company that has two links between locations: one a high-bandwidth, low-delay expensive link, and the other a low-bandwidth, higher-delay, less-expensive link. While using traditional routing protocols, the higher-bandwidth link would get most, if not all, of the traffic sent across it based on the metric savings

obtained by the bandwidth and/or delay (using EIGRP or OSPF) characteristics of the link. PBR allows you to route higher priority traffic over the high-bandwidth/low-delay link, while sending all other traffic over the low-bandwidth/high-delay link.

Some applications of policy based routing are:

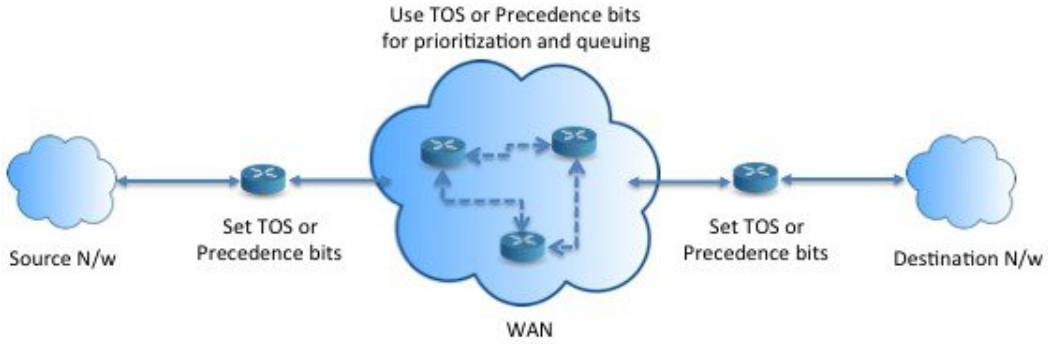
Equal-Access and Source-Sensitive Routing

In this topology, traffic from HR network & Mgmt network can be configured to go through ISP1 and traffic from Eng network can be configured to go through ISP2. Thus, policy based routing enables the network administrators to provide equal-access and source-sensitive routing, as shown here.



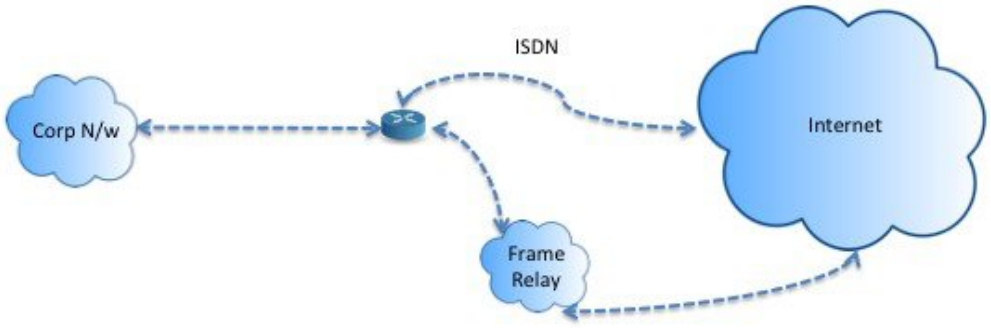
Quality of Service

By tagging packets with policy based routing, network administrators can classify the network traffic at the perimeter of the network for various classes of service and then implementing those classes of service in the core of the network using priority, custom or weighted fair queuing (as shown in the figure below). This setup improves network performance by eliminating the need to classify the traffic explicitly at each WAN interface in the core of backbone network.



Cost Saving

An organization can direct the bulk traffic associated with a specific activity to use a higher-bandwidth high-cost link for a short time and continues basic connectivity over a lower-bandwidth low-cost link for interactive traffic by defining the topology, as show here.



Load Sharing

In addition to the dynamic load-sharing capabilities offered by ECMP load balancing, network administrators can now implement policies to distribute traffic among multiple paths based on the traffic characteristics.

As an example, in the topology depicted in the Equal-Access Source Sensitive Routing scenario, an administrator can configure policy based routing to load share the traffic from HR network through ISP1 and traffic from Eng network through ISP2.

Implementation of PBR

The ASA uses ACLs to match traffic and then perform routing actions on the traffic. Specifically, you configure a route map that specifies an ACL for matching, and then you specify one or more actions for that traffic. Finally, you associate the route map with an interface where you want to apply PBR on all incoming traffic.

Note Before proceeding with configuration, ensure that the ingress and egress traffic of each session flows through the same ISP-facing interface to avoid unexpected behavior caused by asymetirc routing, specifically when NAT and VPN are in use.

Guidelines for Policy Based Routing

Firewall Mode

Supported only in routed firewall mode. Transparent firewall mode is not supported.

Per-flow Routing

Since the ASA performs routing on a per-flow basis, policy routing is applied on the first packet and the resulting routing decision is stored in the flow created for the packet. All subsequent packets belonging to the same connection simply match this flow and are routed appropriately.

PBR Policies Not Applied for Output Route Look-up

Policy Based Routing is an ingress-only feature; that is, it is applied only to the first packet of a new incoming connection, at which time the egress interface for the forward leg of the connection is selected. Note that PBR

will not be triggered if the incoming packet belongs to an existing connection, or if NAT is applied and NAT chooses the egress interface.

PBR Policies Not Applied for Embryonic Traffic

Note An embryonic connection is where the necessary handshake between source and destination has not been made.

When a new internal interface is added and a new VPN policy is created using a unique address pool, PBR is applied to the outside interface matching the source of the new client pool. Thus, PBR sends traffic from the client to the next hop on the new interface. However, PBR is not involved in the return traffic from a host that has not yet established a connection with the new internal interface routes to the client. Thus, the return traffic from the host to the VPN client, specifically, the VPN client response is dropped as there is no valid route. You must configure a weighted static route with a higher metric on the internal interface.

Clustering

- Clustering is supported.
- In a cluster scenario, without static or dynamic routes, with ip-verify-reverse path enabled, asymmetric traffic may get dropped. So disabling ip-verify-reverse path is recommended.

IPv6 Support

IPv6 is supported

Additional Guidelines

- All existing route map related configuration restrictions and limitations will be carried forward.
- Do not use route maps containing match policy lists for policy based routing. The match policy-list is only used for BGP.
- Unicast Reverse Path Forwarding (uRPF) validates the source IP address of packets received on an interface against the routing table and not against the PBR route map. When uRPF is enabled, packets received on an interface through PBR are dropped as they are without the specific route entry. Hence, when using PBR, ensure to disable uRPF.

Configure Policy Based Routing

A route map is comprised of one or more route-map statements. Each statement has a sequence number, as well as a permit or deny clause. Each route-map statement contains match and set commands. The match command denotes the match criteria to be applied on the packet. The set command denotes the action to be taken on the packet.

- When a route map is configured with both IPv4 and IPv6 match/set clauses or when a unified ACL matching IPv4 and IPv6 traffic is used, the set actions will be applied based on destination IP version.

- When multiple next-hops or interfaces are configured as a set action, all options are evaluated one after the other until a valid usable option is found. No load balancing will be done among the configured multiple options.
- The verify-availability option is not supported in multiple context mode.

Procedure

Step 1 Define a standard or extended access-list:

access-list *name* **standard** {**permit** | **deny**} {**any4** | **host** *ip_address* | *ip_address mask*}

access-list *name* **extended** {**permit** | **deny**} *protocol source_and_destination_arguments*

Example:

```
ciscoasa(config)# access-list testacl extended permit ip
10.1.1.0 255.255.255.0 10.2.2.0 255.255.255.0
```

If you use a standard ACL, matching is done on the destination address only. If you use an extended ACL, you can match on source, destination, or both.

For the extended ACL, you can specify IPv4, IPv6, Identity Firewall, or Cisco TrustSec parameters. You can also include network-service objects. For complete syntax, see the ASA command reference.

Step 2 Create a route map entry:

route-map *name* {**permit** | **deny**} [*sequence_number*]

Example:

```
ciscoasa(config)# route-map testmap permit 12
```

Route map entries are read in order. You can identify the order using the *sequence_number* argument, or the ASA uses the order in which you add route map entries.

The ACL also includes its own permit and deny statements. For Permit/Permit matches between the route map and the ACL, the Policy Based Routing processing continues. For Permit/Deny matches, processing ends for this route map, and other route maps are checked. If the result is still Permit/Deny, then the regular routing table is used. For Deny/Deny matches, the Policy Based Routing processing continues.

Note
When a route-map is configured without a permit or deny action and without a sequence-number, it by default will assume the action as permit and sequence-number as 10.

Step 3 Define the match criteria to be applied using an access-list:

match ip address *access-list_name* [*access-list_name...*]

Example:

```
ciscoasa(config-route-map)# match ip address testacl
```

Note

Ensure that the access list does not contain any inactive rules. You cannot set match ACL with inactive rules to a PBR.

Step 4 Configure one or more set actions:

- Set the next hop address:

  set {ip | ipv6} next-hop *ipv4_or_ipv6_address*

  You can configure multiple next-hop IP addresses in which case they are evaluated in the specified order until a valid routable next-hop IP address is found. The configured next-hops should be directly connected; otherwise the set action will not be applied.

- Set the default next hop address:

  set {ip | ipv6} default next-hop *ipv4_or_ipv6_address*

  If the normal route lookup fails for matching traffic, then the ASA forwards the traffic using this specified next-hop IP address.

- Set a recursive next hop IPv4 address:

  set ip next-hop recursive *ip_address*

  Both **set ip next-hop** and **set ip default next-hop** require that the next-hop be found on a directly connected subnet. With **set ip next-hop recursive**, the next-hop address does not need to be directly connected. Instead a recursive lookup is performed on the next-hop address, and matching traffic is forwarded to the next-hop used by that route entry according to the routing path in use on the router.

- Verify if the next IPv4 hops of a route map are available:

  set ip next-hop verify-availability *next-hop-address sequence_number* **track** *object*

  You can configure an SLA monitor tracking object to verify the reachability of the next-hop. To verify the availability of multiple next-hops, multiple **set ip next-hop verify-availability** commands can be configured with different sequence numbers and different tracking objects.

- Set the output interface for the packet:

  set interface *interface_name*

  or

  set interface null0

  This command configures the interface through which the matching traffic is forwarded. You can configure multiple interfaces, in which case they are evaluated in the specified order until a valid interface is found. When you specify **null0**, all traffic matching the route-map will be dropped. There must be a route for the destination that can be routed through the specified interface (either static or dynamic).

- Set the output interface based on the interface's cost:

  set adaptive-interface cost *interface_list*

  The egress interface is selected from the space-separated list of interfaces. If the costs of the interfaces are the same, it is an active-active configuration and packets are load-balanced (round-robin) on the egress interfaces. If the costs are different, the interface with the lowest cost is selected. Interfaces are considered only if they are up. For example:

  ```
  set adaptive-interface cost output1 output2
  ```

- Set the default interface to null0:

  set default interface null0

  If a normal route lookup fails, the ASA forwards the traffic null0, and the traffic will be dropped.

- Set the Don't Fragment (DF) bit value in the IP header:

  set ip df {0|1}

- Classify IP traffic by setting a Differentiated Services Code Point (DSCP) or an IP-precedence value in the packet:

  set {ip | ipv6} dscp *new_dscp*

Note
When multiple set actions are configured, the ASA evaluates them in the following order: **set ip next-hop verify-availability; set ip next-hop**; **set ip next-hop recursive**; **set interface; set adaptive-interface cost**; **set ip default next-hop**; **set default interface**.

Step 5 Configure an interface and enter interface configuration mode:

interface *interface_id*

Example:

```
ciscoasa(config)# interface GigabitEthernet0/0
```

Step 6 If you use **set adaptive-interface cost** as a criteria in the route map, set the cost on the interface:

policy-route cost *value*

The value can be 1-65535. The default is 0, which you can reset by using the **no** version of the command. The lower the number, the higher the priority. For example, 1 has priority over 2.

When you set policy-route cost, and use the **set adaptive-interface cost** command in the route map, the egress traffic is round-robin load balanced across any selected interfaces (assuming they are up) that have the same interface cost. If costs are different, higher cost interfaces are used as backups to the lowest cost interface.

For example, by setting the same cost on 2 WAN links, you can load balance the traffic across those links to perhaps improve performance. However, if one WAN link has higher bandwidth than the other, you can set the higher bandwidth link's cost to 1, and the lower bandwidth link to 2, so that the lower bandwidth link is used only if the higher bandwidth link is down.

Step 7 Configure policy based routing for through-the-box traffic:

`policy-route route-map` *route_map_name*

Example:

```
ciscoasa(config-if)# policy-route route-map testmap
```

To remove an existing Policy Based Routing map, simply enter the **no** form of this command.

Example:

```
ciscoasa(config-if)# no policy-route route-map testmap
```

Examples for Policy Based Routing

The following sections show examples for route map configuration, policy based routing, and a specific example of PBR in action.

Examples for Route Map Configuration

In the following example, since no action and sequence is specified, an implicit action of permit and a sequence number of 10 is assumed:

```
ciscoasa(config)# route-map testmap
```

In the following example, since no match criteria is specified, an implicit match 'any' is assumed:

```
ciscoasa(config)# route-map testmap permit 10
ciscoasa(config-route-map)# set ip next-hop 1.1.1.10
```

In this example, all traffic matching <acl> will be policy routed and forwarded through outside interface.

```
ciscoasa(config)# route-map testmap permit 10
ciscoasa(config-route-map)# match ip address <acl>
ciscoasa(config-route-map)# set interface outside
```

In this example, since there are no interface or next-hop actions are configured, all traffic matching <acl> will have df bit and dscp fields modified as per configuration and are forwarding using normal routing.

```
ciscoasa(config)# route-map testmap permit 10
ciscoasa(config-route-map)# match ip address <acl>
set ip df 1
set ip precedence af11
```

In the following example, all traffic matching <acl_1> is forwarded using next-hop 1.1.1.10, all traffic matching <acl_2> is forwarded using next-hop 2.1.1.10 and rest of the traffic is dropped. No "match" criteria implies an implicit match "any".

```
ciscoasa(config)# route-map testmap permit 10
ciscoasa(config-route-map)# match ip address <acl_1>
ciscoasa(config-route-map)# set ip next-hop 1.1.1.10

ciscoasa(config)# route-map testmap permit 20
ciscoasa(config-route-map)# match ip address <acl_2>

ciscoasa(config-route-map)# set ip next-hop 2.1.1.10
ciscoasa(config)# route-map testmap permit 30
```

```
ciscoasa(config-route-map)# set interface Null0
```

In the following example, the route-map evaluation will be such that (i) a route-map action permit and acl action permit will apply the set actions (ii) a route-map action deny and acl action permit will skip to normal route lookup (iii) a route-map action of permit/deny and acl action deny will continue with next route-map entry. When no next route-map entry available, we will fallback to normal route lookup.

```
ciscoasa(config)# route-map testmap permit 10
ciscoasa(config-route-map)# match ip address permit_acl_1 deny_acl_2
ciscoasa(config-route-map)# set ip next-hop 1.1.1.10

ciscoasa(config)# route-map testmap deny 20
ciscoasa(config-route-map)# match ip address permit_acl_3 deny_acl_4
ciscoasa(config-route-map)# set ip next-hop 2.1.1.10

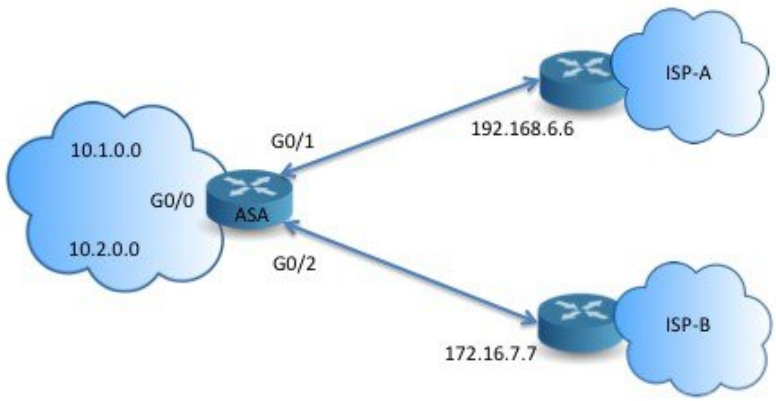
ciscoasa(config)# route-map testmap permit 30
ciscoasa(config-route-map)# match ip address deny_acl_5
ciscoasa(config-route-map)# set interface outside
```

In the following example, when multiple set actions are configured, they are evaluated in the order mentioned above. Only when all options of a set action are evaluated and cannot be applied, the next set actions will be considered. This ordering will ensure that the most available and least distant next-hop will be tried first followed by next most available and least distant next-hop and so on.

```
ciscoasa(config)# route-map testmap permit 10
ciscoasa(config-route-map)# match ip address acl_1
ciscoasa(config-route-map)# set ip next-hop verify-availability 1.1.1.10 1 track 1
ciscoasa(config-route-map)# set ip next-hop verify-availability 1.1.1.11 2 track 2
ciscoasa(config-route-map)# set ip next-hop verify-availability 1.1.1.12 3 track 3
ciscoasa(config-route-map)# set ip next-hop 2.1.1.10 2.1.1.11 2.1.1.12
ciscoasa(config-route-map)# set ip next-hop recursive 3.1.1.10
ciscoasa(config-route-map)# set interface outside-1 outside-2
ciscoasa(config-route-map)# set ip default next-hop 4.1.1.10 4.1.1.11
ciscoasa(config-route-map)# set default interface Null0
```

Example Configuration for PBR

This section describes the complete set of configuration required to configure PBR for the following scenario:



First, we need to configure interfaces.

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# no shutdown
ciscoasa(config-if)# nameif inside
ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0

ciscoasa(config)# interface GigabitEthernet0/1
ciscoasa(config-if)# no shutdown
ciscoasa(config-if)# nameif outside-1
ciscoasa(config-if)# ip address 192.168.6.5 255.255.255.0

ciscoasa(config)# interface GigabitEthernet0/2
ciscoasa(config-if)# no shutdown
ciscoasa(config-if)# nameif outside-2
ciscoasa(config-if)# ip address 172.16.7.6 255.255.255.0
```

Then, we need to configure an access-list for matching the traffic.

```
ciscoasa(config)# access-list acl-1 permit ip 10.1.0.0 255.255.0.0
ciscoasa(config)# access-list acl-2 permit ip 10.2.0.0 255.255.0.0
```

We need to configure a route-map by specifying the above access-list as match criteria along with the required set actions.

```
ciscoasa(config)# route-map equal-access permit 10
ciscoasa(config-route-map)# match ip address acl-1
ciscoasa(config-route-map)# set ip next-hop 192.168.6.6

ciscoasa(config)# route-map equal-access permit 20
ciscoasa(config-route-map)# match ip address acl-2
ciscoasa(config-route-map)# set ip next-hop 172.16.7.7

ciscoasa(config)# route-map equal-access permit 30
ciscoasa(config-route-map)# set ip interface Null0
```

Now, this route-map has to be attached to an interface.

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# policy-route route-map equal-access
```

To display the policy routing configuration.

```
ciscoasa(config)# show policy-route
Interface                   Route map
GigabitEthernet0/0   equal-access
```

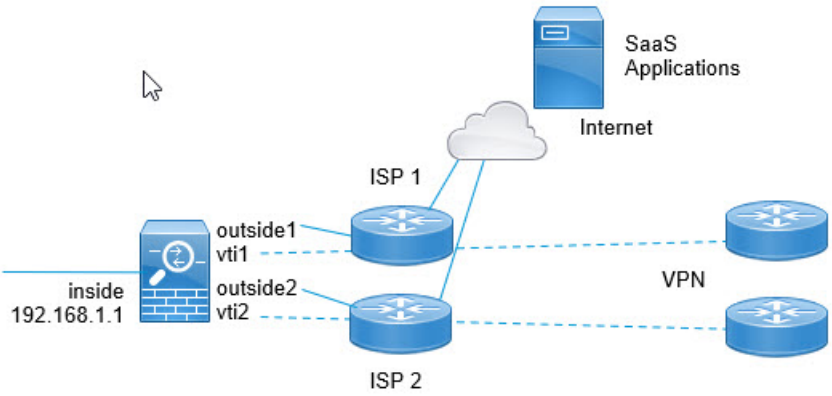
Direct Internet Access using Software-Defined WAN

A typical branch office network uses a site-to-site VPN to connect the branch to a corporate hub. All non-local traffic is then directed to the corporate network, where it is either directed to internal services or to the Internet, as appropriate.

This setup creates a bottleneck at the corporate hub. If some branch traffic is for Internet services, such as Google search or Gmail, there is no need to first go to the corporate network before going to the Internet.

Using policy-based routing, you can instead set up direct Internet access from the branch for traffic that does not need the services of the corporate network. Thus, traffic to the Internet is not sent to the corporate hub, and the hub need only handle traffic destined to the internal services of the corporate network. This configuration should improve overall network performance and throughput.

The following example shows how to set up direct Internet access for the following setup, where two outside interfaces connect to different Internet Service Providers, and virtual tunnel interfaces (VTI) host site-to-site VPN connections to the corporate network. The example shows how to direct traffic destined to select SaaS applications to the Internet and thus bypass the corporate network.



Before you begin

This example assumes that you already have defined a site-to-site VPN using virtual tunnel interfaces (VTI) defined on the outside (WAN-facing) interfaces to connect the branch to the corporate hub and that it is functioning correctly. Traffic routed to the VTI interfaces is thus directed to the corporate network, whereas traffic routed directly to the outside interfaces goes to the Internet.

It also assumes that you have configured DNS servers and enabled DNS resolution on the device interfaces. Use the **show dns trusted-source detail** command to see which servers will be snooped. If you want to limit which servers are used, use the **no dns trusted-source** command to turn off snooping on select servers.

Procedure

Step 1 Configure network-service objects and groups to define the desired traffic.

The following example creates objects to define Office365 and WebEx, then creates a SaaS_Applications object group to contain these. You must create an object group, you cannot use objects directly in an access control entry.

```
object network-service office365
    domain outlook.office365.com tcp eq 443
    domain onlineapps.live.com tcp eq 443
    domain skype.live.com tcp eq 443

object network-service webex
    domain webex.com tcp eq 443

object-group network-service SaaS_Applications
    network-service-member office365
    network-service-member webex
```

Step 2 Create an extended ACL to match the desired traffic.

The following example matches traffic from the inside network to the SaaS Applications object group.

```
access-list DIA_traffic extended permit ip 192.168.1.0 255.255.255.0
object-group-network-service SaaS_Applications
```

Step 3 (Optional.) Configure the cost on the egress interfaces.

Assuming the output1 and output2 interfaces are already configured and functioning, simply add the policy-route cost command. This step is optional if you want to configure the system to use round robin processing to load balance across the 2 egress WAN links. However, you must set the costs if you want to create an active/backup configuration, where one link is used unless it is down.

Following is an example of an equal cost active/active setup.

```
interface G0/0
  nameif outside1
  policy-route cost 1

interface G0/1
  nameif outside2
  policy-route cost 1
```

Following is an example where output1 is the preferred link, and output2 is used only if output1 is down.

```
interface G0/0
  nameif outside1
  policy-route cost 1

interface G0/1
  nameif outside2
  policy-route cost 2
```

Step 4 Create a route map to match the extended ACL and direct traffic accordingly.

The following example uses the ACL to match traffic, then uses adaptive interface cost to direct the traffic to the egress interface.

```
route-map mymap 10
  match ip address DIA_traffic
  set adaptive-interface cost outside1 outside2
```

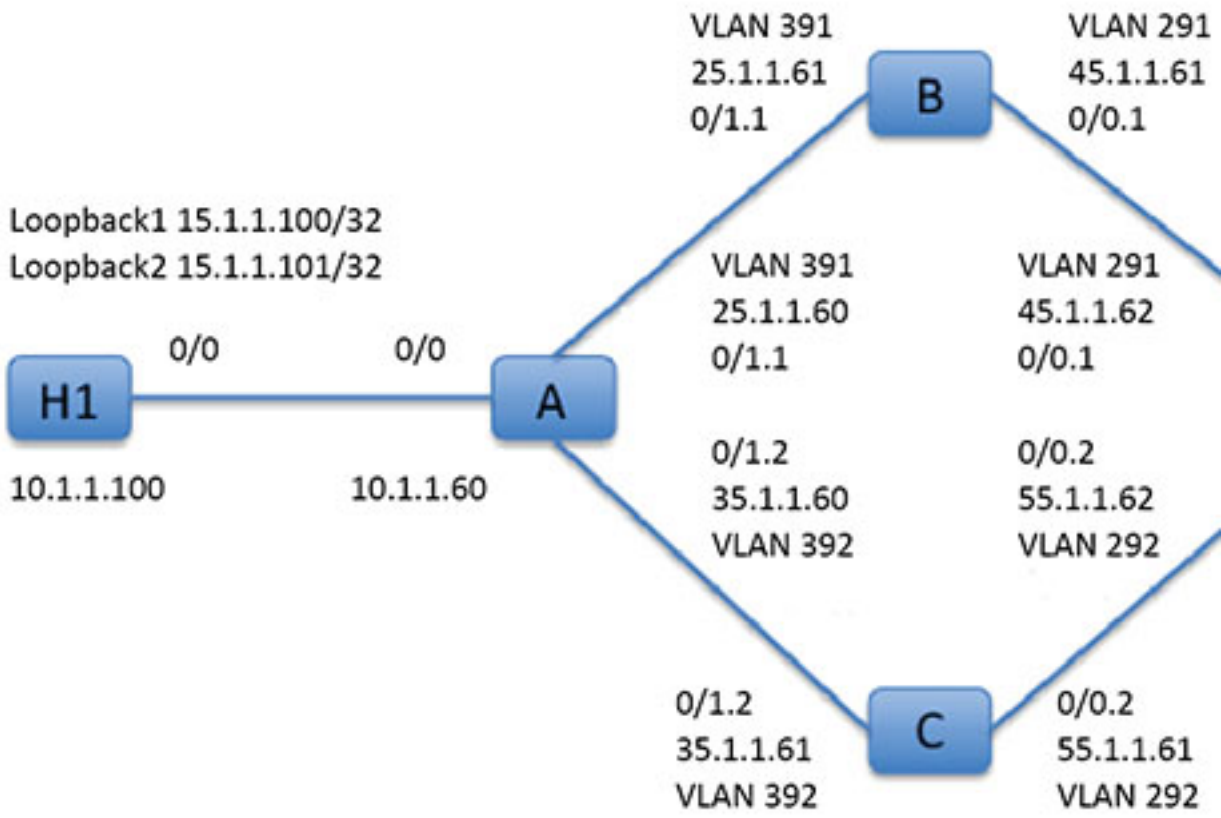
Step 5 Configure policy based routing on the ingress interface to send SaaS traffic to the outside interfaces.

The following example attaches the route map to the inside interface to enable policy-based routing for direct Internet access.

```
interface G1/0
  nameif inside
  policy-route route-map mymap
```

Policy Based Routing in Action

We will use this test setup to configure policy based routing with different match criteria and set actions to see how they are evaluated and applied.



First, we will start with the basic configuration for all the devices involved in the set-up. Here, A, B, C, and D represent ASA devices, and H1 and H2 represent IOS routers.

ASA-A:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# nameif inside
ciscoasa(config-if)# security-level 100
ciscoasa(config-if)# ip address 10.1.1.60 255.255.255.0
ciscoasa(config)# interface GigabitEthernet0/1
ciscoasa(config-if)# no shut

ciscoasa(config)# interface GigabitEthernet0/1.1
ciscoasa(config-if)# vlan 391
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# security-level 0
ciscoasa(config-if)# ip address 25.1.1.60 255.255.255.0
```

```
ciscoasa(config)# interface GigabitEthernet0/1.2
ciscoasa(config-if)# vlan 392
ciscoasa(config-if)# nameif dmz
ciscoasa(config-if)# security-level 50
ciscoasa(config-if)# ip address 35.1.1.60 255.255.255.0
```

ASA-B:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# no shut

ciscoasa(config)# interface GigabitEthernet0/0.1
ciscoasa(config-if)# vlan 291
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# security-level 0
ciscoasa(config-if)# ip address 45.1.1.61 255.255.255.0

ciscoasa(config)# interface GigabitEthernet0/1
ciscoasa(config-if)# no shut

ciscoasa(config)# interface GigabitEthernet0/1.1
ciscoasa(config-if)# vlan 391
ciscoasa(config-if)# nameif inside
ciscoasa(config-if)# security-level 100
ciscoasa(config-if)# ip address 25.1.1.61 255.255.255.0
```

ASA-C:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# no shut

ciscoasa(config)# interface GigabitEthernet0/0.2
ciscoasa(config-if)# vlan 292
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# security-level 0
ciscoasa(config-if)# ip address 55.1.1.61 255.255.255.0

ciscoasa(config)# interface GigabitEthernet0/1
ciscoasa(config-if)# no shut

ciscoasa(config)# interface GigabitEthernet0/1.2
ciscoasa(config-if)# vlan 392
ciscoasa(config-if)# nameif inside
ciscoasa(config-if)# security-level 0
ciscoasa(config-if)# ip address 35.1.1.61 255.255.255.0
```

ASA-D:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# no shut

ciscoasa(config) #interface GigabitEthernet0/0.1
ciscoasa(config-if)# vlan 291
ciscoasa(config-if)# nameif inside-1
ciscoasa(config-if)# security-level 100
ciscoasa(config-if)# ip address 45.1.1.62 255.255.255.0

ciscoasa(config)# interface GigabitEthernet0/0.2
ciscoasa(config-if)# vlan 292
```

```
ciscoasa(config-if)# nameif inside-2
ciscoasa(config-if)# security-level 100
ciscoasa(config-if)# ip address 55.1.1.62 255.255.255.0

ciscoasa(config)# interface GigabitEthernet0/1
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# security-level 0
ciscoasa(config-if)# ip address 65.1.1.60 255.255.255.0
```

H1:

```
ciscoasa(config)# interface Loopback1
ciscoasa(config-if)# ip address 15.1.1.100 255.255.255.255

ciscoasa(config-if)# interface Loopback2
ciscoasa(config-if)# ip address 15.1.1.101 255.255.255.255

ciscoasa(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.60
```

H2:

```
ciscoasa(config)# interface GigabitEthernet0/1
ciscoasa(config-if)# ip address 65.1.1.100 255.255.255.0

ciscoasa(config-if)# ip route 15.1.1.0 255.255.255.0 65.1.1.60
```

We will configure PBR on ASA-A to route traffic sourced from H1.

ASA-A:

```
ciscoasa(config-if)# access-list pbracl_1 extended permit ip host 15.1.1.100 any

ciscoasa(config-if)# route-map testmap permit 10
ciscoasa(config-if)# match ip address pbracl_1
ciscoasa(config-if)# set ip next-hop 25.1.1.61

ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# policy-route route-map testmap

ciscoasa(config-if)# debug policy-route
```

H1: ping 65.1.1.100 repeat 1 source loopback1

```
pbr: policy based route lookup called for 15.1.1.100/44397 to 65.1.1.100/0 proto 1 sub_proto
 8 received on interface inside
pbr: First matching rule from ACL(2)
pbr: route map testmap, sequence 10, permit; proceed with policy routing
pbr: evaluating next-hop 25.1.1.61
pbr: policy based routing applied; egress_ifc = outside : next_hop = 25.1.1.61
```

The packet is forwarded as expected using the next-hop address in the route-map.

When a next-hop is configured, we do a lookup in input route table to identify a connected route to the configured next-hop and use the corresponding interface. The input route table for this example is shown here (with the matching route entry highlighted).

```
in   255.255.255.255 255.255.255.255 identity
```

```
in   10.1.1.60       255.255.255.255 identity
in   25.1.1.60       255.255.255.255 identity
in   35.1.1.60       255.255.255.255 identity
in   10.127.46.17    255.255.255.255 identity
in   10.1.1.0        255.255.255.0   inside
in   25.1.1.0        255.255.255.0   outside
in   35.1.1.0        255.255.255.0   dmz
```

Next let's configure ASA-A to route packets from H1 loopback2 out of ASA-A dmz interface.

```
ciscoasa(config)# access-list pbracl_2 extended permit ip host 15.1.1.101 any

ciscoasa(config)# route-map testmap permit 20
ciscoasa(config-route-map)# match ip address pbracl
ciscoasa(config-route-map)# set ip next-hop 35.1.1.61

ciscoasa(config)# show run route-map
!
route-map testmap permit 10
  match ip address pbracl_1
  set ip next-hop 25.1.1.61
!
route-map testmap permit 20
  match ip address pbracl_2
  set ip next-hop 35.1.1.61
!
```

H1: ping 65.1.1.100 repeat 1 source loopback2

The debugs are shown here:

```
pbr: policy based route lookup called for 15.1.1.101/1234 to 65.1.1.100/1234 proto 6 sub_proto
 0 received on interface inside
pbr: First matching rule from ACL(3)
pbr: route map testmap, sequence 20, permit; proceed with policy routing
pbr: evaluating next-hop 35.1.1.61
pbr: policy based routing applied; egress_ifc = dmz : next_hop = 35.1.1.61
```

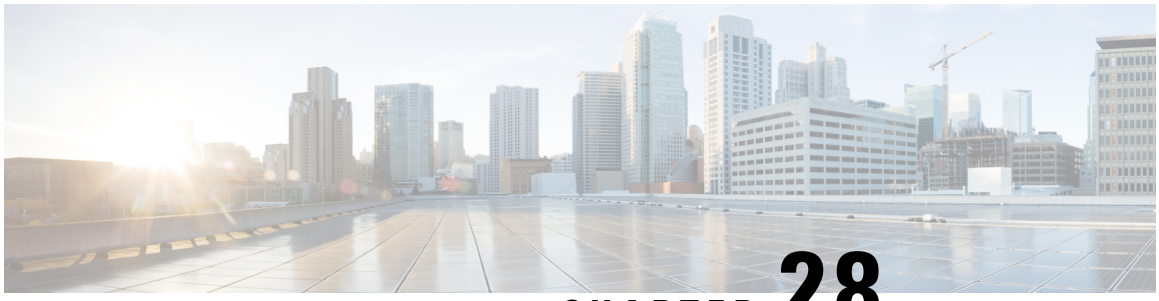
and the route entry chosen from input route table is shown here:

```
in   255.255.255.255 255.255.255.255 identity
 in   10.1.1.60       255.255.255.255 identity
 in   25.1.1.60       255.255.255.255 identity
 in   35.1.1.60       255.255.255.255 identity
 in   10.127.46.17    255.255.255.255 identity
 in   10.1.1.0        255.255.255.0   inside
 in   25.1.1.0        255.255.255.0   outside
 in   35.1.1.0        255.255.255.0   dmz
```

History for Policy Based Routing

Table 34: History for Route Maps

Feature Name	Platform Releases	Feature Information
Policy based routing	9.4(1)	Policy Based Routing (PBR) is a mechanism by which traffic is routed through specific paths with a specified QoS using ACLs. ACLs let traffic be classified based on the content of the packet's Layer 3 and Layer 4 headers. This solution lets administrators provide QoS to differentiated traffic, distribute interactive and batch traffic among low-bandwidth, low-cost permanent paths and high-bandwidth, high-cost switched paths, and allows Internet service providers and other organizations to route traffic originating from various sets of users through well-defined Internet connections. We introduced the following commands: **set ip next-hop verify-availability**, **set ip next-hop**, **set ip next-hop recursive**, **set interface**, **set ip default next-hop, set default interface**, **set ip df**, **set ip dscp**, **policy-route route-map**, **show policy-route**, **debug policy-route**
IPv6 support for Policy Based Routing	9.5(1)	IPv6 addresses are now supported for Policy Based Routing. We introduced the following commands: **set ipv6 next-hop**,**set default ipv6-next hop**, **set ipv6 dscp**
VXLAN support for Policy Based Routing	9.5(1)	You can now enable Policy Based Routing on a VNI interface. We did not modify any commands.
Policy Based Routing support for Identity Firewall and Cisco Trustsec	9.5(1)	You can configure Identity Firewall and Cisco TrustSec and then use Identity Firewall and Cisco TrustSec ACLs in Policy Based Routing route maps. We did not modify any commands.



CHAPTER 28

Route Maps

This chapter describes how to configure and customize route-maps, for ASA.

- About Route Maps, on page 879
- Guidelines for Route Maps, on page 881
- Define a Route Map, on page 881
- Customize a Route Map, on page 881
- Example for Route Maps, on page 883
- History for Route Maps, on page 884

About Route Maps

Route maps are used when redistributing routes into an OSPF, RIP, EIGRP or BGP routing process. They are also used when generating a default route into an OSPF routing process. A route map defines which of the routes from the specified routing protocol are allowed to be redistributed into the target routing process.

Route maps have many features in common with widely known ACLs. These are some of the traits common to both:

- They are an ordered sequence of individual statements, and each has a permit or deny result. Evaluation of an ACL or a route map consists of a list scan, in a predetermined order, and an evaluation of the criteria of each statement that matches. A list scan is aborted once the first statement match is found and an action associated with the statement match is performed.
- They are generic mechanisms. Criteria matches and match interpretation are dictated by the way that they are applied and the feature that uses them. The same route map applied to different features might be interpreted differently.

These are some of the differences between route maps and ACLs:

- Route maps are more flexible than ACLs and can verify routes based on criteria which ACLs can not verify. For example, a route map can verify if the type of route is internal.
- Each ACL ends with an implicit deny statement, by design convention. If the end of a route map is reached during matching attempts, the result depends on the specific application of the route map. Route maps that are applied to *redistribution* behave the same way as ACLs: if the route does not match any clause in a route map then the route redistribution is denied, as if the route map contained a deny statement at the end.

Permit and Deny Clauses

Route maps can have permit and deny clauses. The deny clause rejects route matches from redistribution. You can use an ACL as the matching criterion in the route map. Because ACLs also have permit and deny clauses, the following rules apply when a packet matches the ACL:

- ACL permit + route map permit: routes are redistributed.
- ACL permit + route map deny: routes are not redistributed.
- ACL deny + route map permit or deny: the route map clause is not matched, and the next route-map clause is evaluated.

Match and Set Clause Values

Each route map clause has two types of values:

- A match value selects routes to which this clause should be applied.
- A set value modifies information that will be redistributed into the target protocol.

For each route that is being redistributed, the router first evaluates the match criteria of a clause in the route map. If the match criteria succeeds, then the route is redistributed or rejected as dictated by the permit or deny clause, and some of its attributes might be modified by the values set from the set commands. If the match criteria fail, then this clause is not applicable to the route, and the software proceeds to evaluate the route against the next clause in the route map. Scanning of the route map continues until a clause is found that matches the route or until the end of the route map is reached.

A match or set value in each clause can be missed or repeated several times, if one of these conditions exists:

- If several match entries are present in a clause, all must succeed for a given route in order for that route to match the clause (in other words, the logical AND algorithm is applied for multiple match commands).
- If a match entry refers to several objects in one entry, either of them should match (the logical OR algorithm is applied).
- If a match entry is not present, all routes match the clause.
- If a set entry is not present in a route map permit clause, then the route is redistributed without modification of its current attributes.



Note Do not configure a set entry in a route map deny clause because the deny clause prohibits route redistribution—there is no information to modify.

A route map clause without a match or set entry does perform an action. An empty permit clause allows a redistribution of the remaining routes without modification. An empty deny clause does not allow a redistribution of other routes (this is the default action if a route map is completely scanned, but no explicit match is found).

Guidelines for Route Maps

Firewall Mode

Supported only in routed firewall mode. Transparent firewall mode is not supported.

Additional Guidelines

Route maps do not support ACLs that include a user, user group, or fully qualified domain name objects.

Define a Route Map

You must define a route map when specifying which of the routes from the specified routing protocol are allowed to be redistributed into the target routing process.

Procedure

Create the route map entry:

route-map *name* {**permit** | **deny**} [*sequence_number*]

Example:

```
ciscoasa(config)# route-map name {permit} [12]
```

Route map entries are read in order. You can identify the order using the *sequence_number* argument, or the ASA uses the order in which you add route map entries.

Customize a Route Map

This section describes how to customize the route map.

Define a Route to Match a Specific Destination Address

Procedure

Step 1 Create the route map entry:

route-map *name* {**permit** | **deny**} [*sequence_number*]

Example:

```
ciscoasa(config)# route-map name {permit} [12]
```

Route map entries are read in order. You can identify the order using the *sequence_number* option, or the ASA uses the order in which you add route map entries.

Step 2 Match any routes that have a destination network that matches a standard ACL or prefix list:

match ip address {*acl_id* [*acl_id*] [...] | **prefix-list** *prefix_list_id* [*prefix_list_id*] [...]}

Example:

```
ciscoasa(config-route-map)# match ip address acl1 acl2 acl3
```

If you specify more than one ACL or prefix list, then the route can match any of the ACLs or prefix lists.

Step 3 Match any routes that have a specified metric:

match metric *metric_value*

Example:

```
ciscoasa(config-route-map)# match metric 200
```

The *metric_value* can range from 0 to 4294967295.

Step 4 Match any routes that have a next hop router address that matches a standard ACL:

match ip next-hop *acl_id* [*acl_id*] [...]

Example:

```
ciscoasa(config-route-map)# match ip next-hop acl2
```

If you specify more than one ACL, then the route can match any of the ACLs.

Step 5 Match any routes with the specified next hop interface:

match interface *if_name*

Example:

```
ciscoasa(config-route-map)# match interface if_name
```

If you specify more than one interface, then the route can match either interface.

Step 6 Match any routes that have been advertised by routers that match a standard ACL:

match ip route-source *acl_id* [*acl_id*] [...]

Example:

```
ciscoasa(config-route-map)# match ip route-source acl_id [acl_id] [...]
```

If you specify more than one ACL, then the route can match any of the ACLs.

Step 7 Match the route type:

match route-type {**internal** | **external** [**type-1** | **type-2**]}

Configure the Metric Values for a Route Action

If a route matches the **match** commands, then the following **set** commands determine the action to perform on the route before redistributing it.

To configure the metric value for a route action, perform the following steps:

Procedure

Step 1 Create the route map entry:

route-map *name* {**permit** | **deny**} [*sequence_number*]

Example:

```
ciscoasa(config)# route-map name {permit} [12]
```

Route map entries are read in order. You can identify the order using the *sequence_number* argument, or the ASA uses the order in which you add route map entries.

Step 2 Set the metric value for the route map:

set metric *metric_value*

Example:

```
ciscoasa(config-route-map)# set metric 200
```

The *metric_value* argument can range from 0 to 294967295.

Step 3 Set the metric type for the route map:

set metric-type {**type-1** | **type-2**}

Example:

```
ciscoasa(config-route-map)# set metric-type type-2
```

The *metric-type* argument can be type-1 or type-2.

Example for Route Maps

The following example shows how to redistribute routes with a hop count equal to 1 into OSPF.

The ASA redistributes these routes as external LSAs with a metric of 5 and a metric type of Type 1.

```
ciscoasa(config)# route-map 1-to-2 permit
ciscoasa(config-route-map)# match metric 1
ciscoasa(config-route-map)# set metric 5
ciscoasa(config-route-map)# set metric-type type-1
```

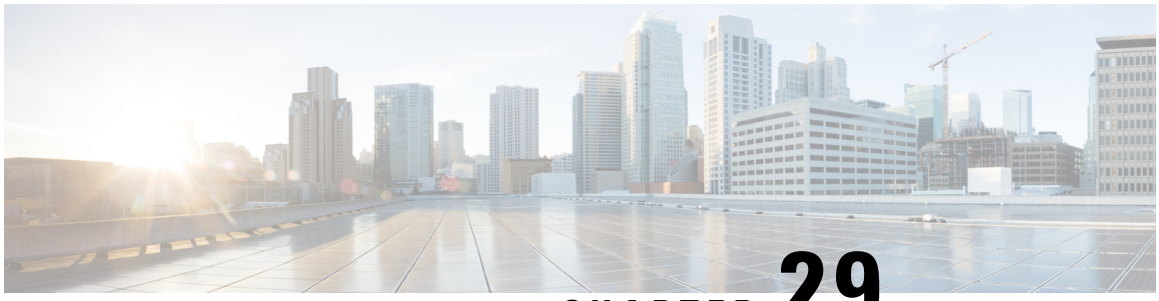
The following example shows how to redistribute the 10.1.1.0 static route into eigrp process 1 with the configured metric value:

```
ciscoasa(config)# route outside 10.1.1.0 255.255.255.0 192.168.1.1
ciscoasa(config-route-map)# access-list mymap2 line 1 permit 10.1.1.0 255.255.255.0
ciscoasa(config-route-map)# route-map mymap2 permit 10
ciscoasa(config-route-map)# match ip address mymap2
ciscoasa(config-route-map)# router eigrp 1
ciscoasa(config-router)# redistribute static metric 250 250 1 1 1 route-map mymap2
```

History for Route Maps

Table 35: Feature History for Route Maps

Feature Name	Platform Releases	Feature Information
Route maps	7.0(1)	We introduced this feature. We introduced the following command: **route-map**.
Enhanced support for static and dynamic route maps	8.0(2)	Enhanced support for dynamic and static route maps was added.
Support for Stateful Failover of dynamic routing protocols (EIGRP, OSPF, and RIP) and debugging of general routing-related operations	8.4(1)	We introduced the following commands: **debug route**, **show debug route**. We modified the following command: **show route**.
Dynamic Routing in Multiple Context Mode	9.0(1)	Route maps are supported in multiple context mode.
Support for BGP	9.2(1)	We introduced this feature. We introduced the following commands: **router bgp**
IPv6 support for Prefix Rule	9.3.2	We introduced this feature.



CHAPTER 29

Bidirectional Forwarding Detection Routing

This chapter describes how to configure the ASA to use the Bidirectional Forwarding Detection (BFD) routing protocol.

- About BFD Routing, on page 885
- Guidelines for BFD Routing, on page 888
- Configure BFD, on page 889
- Monitoring for BFD, on page 893
- History for BFD Routing, on page 894

About BFD Routing

BFD is a detection protocol designed to provide fast forwarding path failure detection times for all media types, encapsulations, topologies, and routing protocols. BFD operates in a unicast, point-to-point mode on top of any data protocol being forwarded between two systems. Packets are carried in the payload of the encapsulating protocol appropriate for the media and the network.

BFD provides a consistent failure detection method for network administrators in addition to fast forwarding path failure detection. Because the network administrator can use BFD to detect forwarding path failures at a uniform rate, rather than the variable rates for different routing protocol hello mechanisms, network profiling and planning are easier and reconvergence time is consistent and predictable.

BFD Asynchronous Mode and Echo Function

BFD can operate in asynchronous mode with or without the echo function enabled.

Asynchronous Mode

In asynchronous mode, the systems periodically send BFD control packets to one another, and if a number of those packets in a row are not received by the other system, the session is declared to be down. Pure asynchronous mode (without the Echo function) is useful because it requires half as many packets to achieve a particular detection time as the Echo function requires.

BFD Echo Function

The BFD echo function sends echo packets from the forwarding engine to the directly-connected single-hop BFD neighbor. The echo packets are sent by the forwarding engine and forwarded back along the same path to perform detection. The BFD session at the other end does not participate in the actual forwarding of the echo packets. Because the echo function and the forwarding engine are responsible for the detection

process, the number of BFD control packets that are sent out between BFD neighbors is reduced. And also because the forwarding engine is testing the forwarding path on the remote neighbor system without involving the remote system, the inter-packet delay variance is improved. This results in quicker failure detection times.

When the echo function is enabled, BFD can use the slow timer to slow down the asynchronous session and reduce the number of BFD control packets that are sent between BFD neighbors, which reduces processing overhead while at the same time delivering faster failure detection.

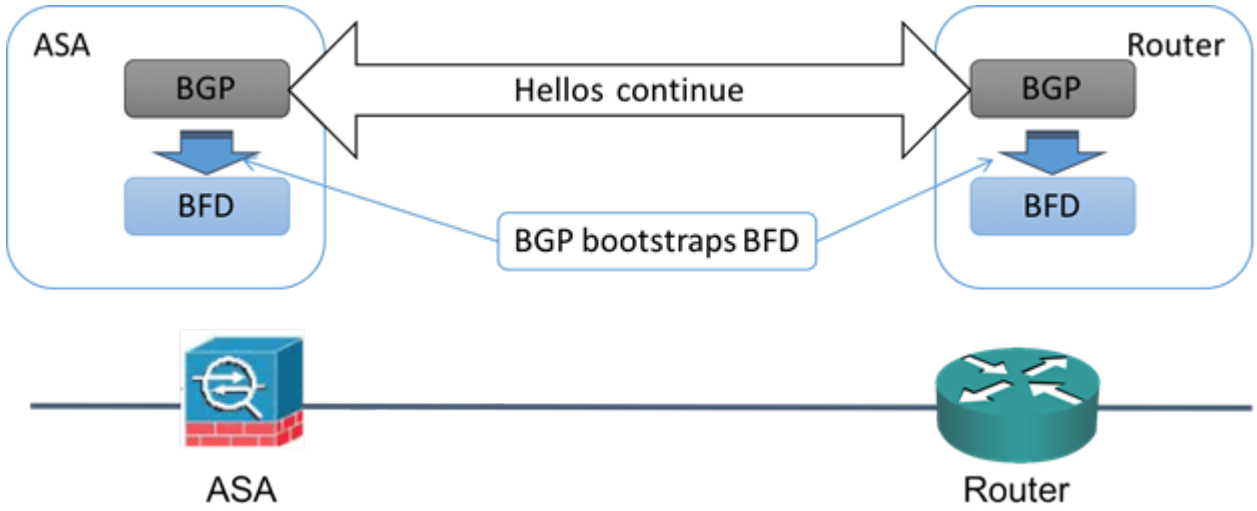
Note The echo function is not supported for IPv4 multi-hop or IPv6 single-hop BFD neighbors.

You can enable BFD at the interface and routing protocol levels. You must configure BFD on both systems (BFD peers). After you enable BFD on the interfaces and at the router level for the appropriate routing protocols, a BFD session is created, BFD timers are negotiated, and the BFD peers begin to send BFD control packets to each other at the negotiated level.

BFD Session Establishment

The following example shows the ASA and a neighboring router running Border Gateway Protocol (BGP). At the time when both devices come up, there is no BFD session established between them.

Figure 58: Established BFD Session



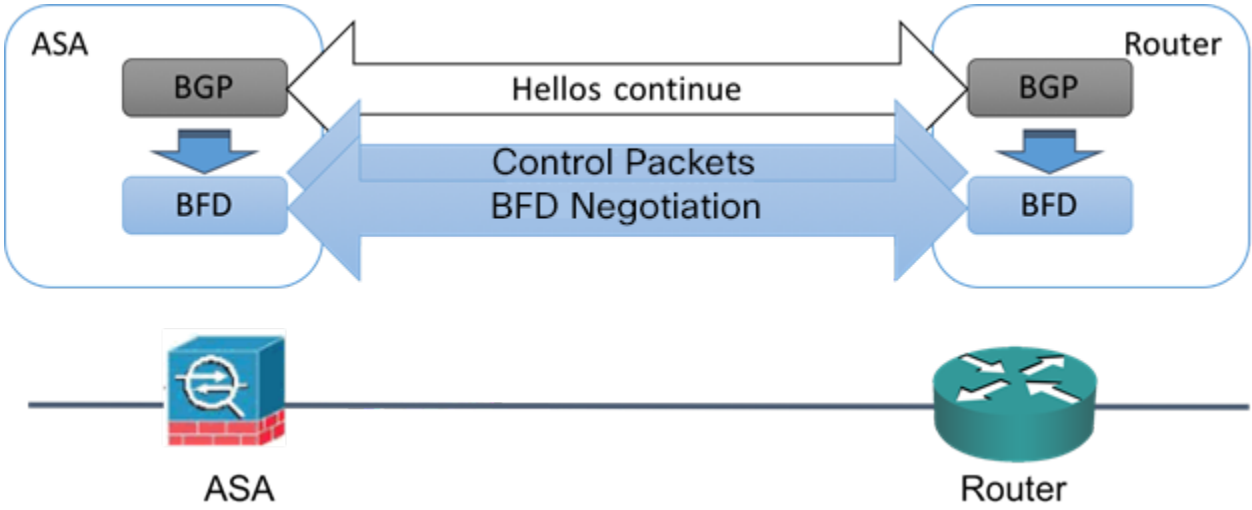
After BGP identifies its BGP neighbor, it bootstraps the BFD process with the IP address of the neighbor. BFD does not discover its peers dynamically. It relies on the configured routing protocols to tell it which IP addresses to use and which peer relationships to form.

The BFD on the router and the BFD on the ASA form a BFD control packet and start sending the packets to each other at a one-second interval until the BFD session is established. The initial control packets from either system are very similar, for example, the Vers, Diag, H, D, P, and F bits are all set to zero, and the State is set to Down. The My Discriminator field is set to a value that is unique on the transmitting device. The Your Discriminator field is set to zero because the BFD session has not yet been established. The TX and RX timers are set to the values found in the configuration of the device.

After the remote BFD device receives a BFD control packet during the session initiation phase, it copies the value of the My Discriminator field into its own Your Discriminator field and the transition from Down state to Init state and then eventually to Up state occurs. Once both systems see their own Discriminators in each other's control packets, the session is officially established.

The following illustration shows the established BFD connection.

Figure 59: BGP With No BFD Session Established



BFD Timer Negotiation

BFD devices must negotiate the BFD timers to control and synchronize the send rate of BFD control packets. A device needs to ensure the following before it can negotiate a BFD timer:

- That its peer device saw the packet containing the proposed timers of the local device
- That it never sends BFD control packets faster than the peer is configured to receive them
- That the peer never sends BFD control packets faster than the local system is configured to receive them

The setting of the Your Discriminator field and the H bit are sufficient to let the local device that the remote device has seen its packets during the initial timer exchange. After receiving a BFD control packet, each system takes the Required Min RX Interval and compares it to its own Desired Min TX Interval, and then takes the greater (slower) of the two values and uses it as the transmission rate for its BFD packets. The slower of the two systems determines the transmission rate.

When these timers have been negotiated, they can be renegotiated at any time during the session without causing a session reset. The device that changes its timers sets the P bit on all subsequent BFD control packets until it receives a BFD control packet with the F bit set from the remote system. This exchange of bits guards against packets that might otherwise be lost in transit.

Note The setting of the F bit by the remote system does not mean that it accepts the newly proposed timers. It indicates that the remote system has seen the packets in which the timers were changed.

BFD Failure Detection

When the BFD session and timers have been negotiated, the BFD peers send BFD control packets to each other at the negotiated interval. These control packets act as a heartbeat that is very similar to IGP Hello protocol except that the rate is more accelerated.

As long as each BFD peer receives a BFD control packet within the configured detection interval (Required Minimum RX Interval), the BFD session stays up and any routing protocol associated with BFD maintains its adjacencies. If a BFD peer does not receive a control packet within this interval, it informs any clients participating in that BFD session about the failure. The routing protocol determines the appropriate response to that information. The typical response is to terminate the routing protocol peering session and reconverge and thus bypass a failed peer.

Each time a BFD peer successfully receives a BFD control packet in a BFD session, the detection timer for that session is reset to zero. Thus the failure detection is dependent on received packets and NOT when the receiver last transmitted a packet.

BFD Deployment Scenarios

The following describes how BFD operates in these specific scenarios.

Failover

In a failover scenario, BFD sessions are established and maintained between the active unit and the neighbor unit. Standby units do not maintain any BFD sessions with the neighbors. When a failover happens, the new active unit must initiate session establishment with the neighbor because session information is not synched between active and standby units.

For a graceful restart/NSF scenario, the client (BGP IPv4/IPv6) is responsible for notifying its neighbor about the event. When the neighbor receives the information, it keeps the RIB table until failover is complete. During failover, the BFD and the BGP sessions go down on the device. When the failover is complete, a new BFD session between the neighbors is established when the BGP session comes up.

Spanned EtherChannel and L2 Cluster

In a Spanned EtherChannel cluster scenario, the BFD session is established and maintained between the primary unit and its neighbor. Subordinate units do not maintain any BFD sessions with the neighbors. If a BFD packet is routed to the subordinate unit because of load balancing on the switch, the subordinate unit must forward this packet to the primary unit through the cluster link. When a cluster switchover happens, the new primary unit must initiate session establishment with the neighbor because session information is not synched between primary and subordinate units.

Individual Interface Mode and L3 Cluster

In an individual interface mode cluster scenario, individual units maintain their BFD sessions with their neighbors.

Guidelines for BFD Routing

Context Mode Guidelines

Supported in single and multiple context modes.

Firewall Mode Guidelines

Supported in routed firewall mode; support for standalone, failover, and cluster modes. BFD is not supported on failover and cluster interfaces. In clustering this feature is only supported on the primary unit. BFD is not supported in transparent mode.

IPv6 Guidelines

Echo mode is not supported for IPv6.

Additional Guidelines

BGP IPv4 and BGP IPv6 protocol are supported.

OSPFv2, OSPFv3, IS-IS, and EiGRP protocols are not supported.

BFD for Static Routes is not supported.

BFD on Transfer and Tunnel is not supported.

Configure BFD

This section describes how to enable and configure the BFD routing process on your system.

Procedure

Step 1 Create the BFD Template, on page 889.
Step 2 Configure BFD Interfaces, on page 891.
Step 3 Configure BFD Maps, on page 892.

Create the BFD Template

This section describes the steps required to create a BFD template and enter BFD configuration mode.

The BFD template specifies a set of BFD interval values. BFD interval values as configured in the BFD template are not specific to a single interface. You can also configure authentication for single-hop and multi-hop sessions. You can enable Echo on single-hop only.

Procedure

Step 1 Enable BFD as a routing protocol on the ASA by creating the BFD template, either single-hop or multi-hop:

bfd-template [single-hop | multi-hop] *template_name*

Example:

```
ciscoasa(config)# bfd-template single-hop TEMPLATE1
ciscoasa(config-bfd)#
```

- **single-hop**—Specifies a single-hop BFD template.
- **multi-hop**—Specifies a multi-hop BFD template.
- *template-name*—Specifies the template name. The template name cannot contains spaces.

The **bfd-template** command lets you create the BFD template and enter BFD configuration mode.

Step 2 (Optional) Configure Echo on a single-hop BFD template:

bfd-template single-hop *template_name*

Example:

```
ciscoasa(config)# bfd-template single-hop TEMPLATE1
ciscoasa (config-bfd)# echo
```

You can only enable Echo mode on a single-hop template. BFD echo is not supported for IPv6 BFD sessions.

Step 3 Configure the intervals in the BFD template:

interval [**both** *milliseconds* | **microseconds {both | min-tx}** *microseconds* | **min-tx** *milliseconds*

Example:

```
ciscoasa(config)# bfd-template single-hop TEMPLATE1
ciscoasa(config-bfd)# interval both 50
```

- **both**—Minimum transmit and receive interval capability.
- *milliseconds*—The interval in milliseconds. The range is 50 to 999.
- **microseconds**—Specifies the BFD interval in microseconds for **both** and **min-tx**.
- *microseconds*—The range is 50,000 to 999,000.
- **min-tx**—The minimum transmit interval capability.

BFD interval values specified as part of the BFD template are not specific to a single interface. You can apply individual BFD templates per interface. See Configure BFD Interfaces, on page 891.

Step 4 Configure authentication in the BFD template:

authentication {md5 | meticulous-mds | meticulous-sha-1 | sha-1} [0|8] *word* **key-id** *id*

Example:

```
ciscoasa(config)# bfd-template single-hop TEMPLATE1
ciscoasa(config-bfd)# authentication sha-1 0 cisco key-id 10
```

- **authentication**—Specifies the authentication type.
- **md5**—Message Digest 5 (MD5) authentication.
- **meticulous-md5**—Meticulous keyed MD5 authentication.
- **meticulous-sha-1**—Meticulous keyed SHA-1 authentication.
- **sha-1**—Keyed SHA-1 authentication.

- **0|8**—0 specifies that an UNENCRYPTED password will follow. 8 specifies that an ENCRYPTED password will follow.
- *word*—The BFD password (key), which is a single-digit password/key of up to 29 characters. Passwords starting with a digit followed by a whitespace are not supported, for example, '0 pass' and '1' are not valid.
- **key-id**—The authentication Key ID.
- *id*—The shared key ID that matches the key string. The range is 0 to 255 characters.

You can configure authentication in single-hop and multi-hop templates. We recommend that you configure authentication to enhance security. You must configure authentication on each BFD source-destination pair and the authentication parameters must match on both devices.

Configure BFD Interfaces

You can bind a BFD template to an interface, configure the baseline BFD session parameters per interface, and enable echo mode per interface.

Procedure

Step 1 Enter interface configuration mode:

interface *interface_id*

Example:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)#
```

Step 2 Apply a BFD template to an interface:

bfd template *template-name*

Example:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# bfd template TEMPLATE1
```

Even if you have not created the template using the **bfd-template** command, you can configure the name of the template under an interface, but the template is considered invalid until you define the template. You do not have to reconfigure the template name again. It becomes valid automatically.

Step 3 Configure the BFD session parameters:

bfd interval *milliseconds* **min_rx** *milliseconds* **multiplier** *multiplier-value*

Example:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-router)# bfd interval 200 min_rx 200 multiplier 3
```

- **interval** *milliseconds*—Specifies the rate at which BFD control packets are sent to BFD peers. The range is 50 to 999 milliseconds.
- **min_rx** *milliseconds*—Specifies the rate at which BFD control packets are expected to be received from BFD peers. The range is 50 to 999 milliseconds.
- **multiplier** *multiplier-value*—Specifies the number of consecutive BFD control packets that must be missed from a BFD peer before BFD declares that the peer is unavailable and the Layer 3 BFD peer is informed of the failure. The range is 3 to 50.

Step 4 Enable BFD echo mode on an interface:

bfd echo

Example:

```
ciscoasa(config)# interface gigabitethernet 0/0
ciscoasa(if)# bfd echo
```

Echo mode is enabled by default but not supported in BFD IPv6 sessions. When echo mode is enabled, the minimum echo transmit level and required minimum transmit interval values are taken from the **bfd interval** *milliseconds* **min_rx** *milliseconds* configuration.

Note
Before using BFD echo mode, you must disable ICMP redirect messages using the **no ip redirects** command. This avoids high CPU use.

Configure BFD Maps

You can create a BFD map containing destinations that you can associate with a multi-hop template. You must have a multi-hop BFD template already configured.

Procedure

Step 1 Create a multi-hop BFD template. See Create the BFD Template, on page 889 for the procedure.

Step 2 Associate the BFD multi-hop template with a map of destinations:

bfd map {ipv4 | ipv6} *destination/cdir source/cdire template-name*

Example:

```
ciscoasa(config)# bfd map ipv4 10.11.11.0/24 10.36.42.5/32 MULTI-TEMPLATE1
ciscoasa(config-bfd)#
```

- **ipv4** —Configures an IPv4 address.
- **ipv6** —Configures an IPv6 address.
- *destination/cdir*—Specifies the destination prefix/length. The format is A.B.C.D/<0-32>.

- *source/cdir*—Specifies the destination prefix/length. The format is X:X:X;X::X/<0-128>.
- *template-name*—Specifies the name of the multi-hop template associated with this BFD map.

Step 3 (Optional) Configure the BFD slow timers value:

bfd slow-timers [*milliseconds*]

Example:

```
ciscoasa(config)# bfd slow-timers 14000
ciscoasa(config-bfd)#
```

milliseconds—(Optional) The BFD slow timers value. The range is 1000 to 30000. The default is 1000.

Monitoring for BFD

You can use the following commands to monitor the BFD routing process. For examples and descriptions of the command output, see the command reference.

To monitor or disable various BFD routing statistics, enter one of the following commands:

- **show bfd neighbors**

  Displays a line-by-line listing of existing BFD adjacencies.

- **show bfd summary**

  Displays summary information for BFD, BFD clients, or BFD sessions.

- **show bfd drops**

  Displays the number of dropped packets in BFD.

- **show bfd map**

  Displays the configured BFD maps.

- **show running-config bfd**

  Displays BFD map and other BFD related configurations.

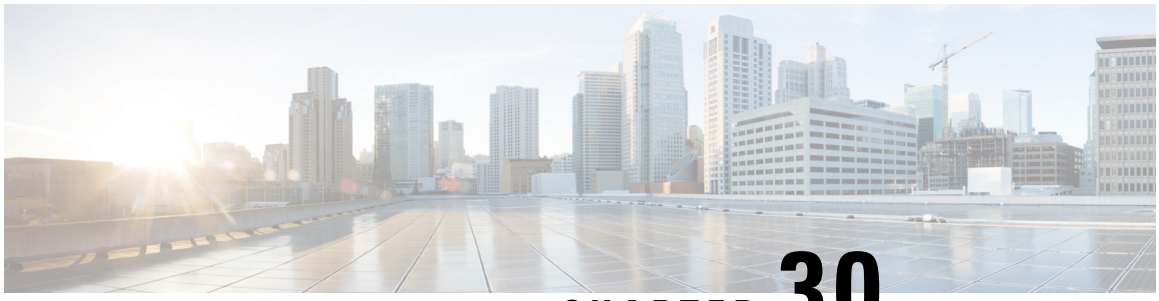
- **show running-config bfd-template**

  Displays BFD-template related configurations.

History for BFD Routing

Table 36: Feature History for BFD Routing

Feature Name	Platform Releases	Feature Information
BFD routing support	9.6(2)	The ASA now supports the BFD routing protocol. Support was added for configuring BFD templates, interfaces, and maps. Support for BGP routing protocol to use BFD was also added. We added the following commands: **bfd echo, bfd interval, bfd map, bfd slow-timers, bfd-template, clear bfd counters, clear conf bfd, neighbor fall-over bfd, show bfd drops, show bfd map, show bfd neighbors, show bfd summary, show running-config bfd**



CHAPTER 30

BGP

This chapter describes how to configure the ASA to route data, perform authentication, and redistribute routing information using the Border Gateway Protocol (BGP).

- BGP, on page 895
- Guidelines for BGP, on page 899
- Configure BGP, on page 900
- Monitoring BGP, on page 929
- Example for BGP, on page 931
- History for BGP, on page 934

BGP

Border Gateway Protocol (BGP) is an inter and intra-autonomous system routing protocol that

- operates within and between autonomous systems (networks or groups of networks under common administration with common routing policies),
- exchanges routing information for the Internet, and
- serves as the protocol used between Internet service providers (ISPs).

BGP usage scenarios

BGP usage scenarios are network configurations that

- enable ISPs to exchange customer and ISP routes between autonomous systems using External BGP (EBGP),
- allow service providers to exchange routes within an autonomous system using Interior BGP (IBGP), and
- support IPv6 prefix routing over IPv6 networks.

BGP implementation contexts

Customer networks, such AS universities and corporations, usually employ an Interior Gateway Protocol (IGP) such AS OSPF for the exchange of routing information within their networks. Customers connect to

ISPs, and ISPs use BGP to exchange customer and ISP routes. When BGP is used between autonomous systems (AS), the protocol is referred to AS External BGP (EBGP). If a service provider is using BGP to exchange routes within an AS, then the protocol is referred to AS Interior BGP (IBGP).

BGP can also be used for carrying routing information for IPv6 prefix over IPv6 networks.



Note When a BGPv6 device joins the cluster, it generates a soft traceback when logging level 7 is enabled.

Routing table changes

Routing table changes are network updates that

- occur when BGP neighbors exchange full routing information upon initial TCP connection establishment
- involve sending only changed routes to neighbors when routing table modifications are detected, and
- advertise only the optimal PATH to a destination network without periodic updates.

BGP attributes for route selection

RRoutes learned via BGP have properties used to determine the best route when multiple paths to a destination exist. These properties are called BGP attributes and are used in the route selection process:

- Weight—This is a Cisco-defined attribute that is local to a router. The weight attribute is not advertised to neighboring routers. If the router learns about more than one route to the same destination, the route with the highest weight is preferred.
- Local preference—The local preference attribute selects an exit point from the local AS. Unlike the weight attribute, it is shared throughout the local AS. If multiple exit points exist, the one with the highest local preference attribute is chosen for a specific route.
- Multi-exit discriminator—The multi-exit discriminator (MED) or metric attribute is used AS a suggestion to an external AS regarding the preferred route into the AS that is advertising the metric. It is referred to AS a suggestion because the external AS that is receiving the MEDs may also be using other BGP attributes for route selection. The route with the lower MED metric is preferred.
- Origin—The origin attribute indicates how BGP learned about a particular route. The origin attribute can have one of three possible values and is used in route selection.
  - IGP—The route is interior to the originating AS. This value is set when the network router configuration command is used to inject the route into BGP.
  - EGP—The route is learned via the Exterior Border Gateway Protocol (EBGP).
  - Incomplete—The origin of the route is unknown or learned in some other way. An origin of incomplete occurs when a route is redistributed into BGP.
- AS_path—When a route advertisement passes through an autonomous system, the AS number is added to an ordered list of AS numbers that the route advertisement has traversed. Only the route with the shortest AS_path list is installed in the IP routing table.
- Next hop—The EBGP next-hop attribute is the IP address that is used to reach the advertising router. For EBGP peers, the next-hop address is the IP address of the connection between the peers. For IBGP,

the EBGP next-hop address is carried into the local AS. However, when the next hop is in the same subnet AS the peering address of the EBGP peer, the next hop is not modified. This behavior is referred to AS the third party next hop.

Use the **next-hop-self** command when redistributing VPN-advertised routes to IBGP peers to ensure that the routes are redistributed with the correct next hop IP.

- Community—The community attribute provides a way of grouping destinations, called communities, to which routing decisions (such AS acceptance, preference, and redistribution) can be applied. Route maps are used to set the community attribute. The predefined community attributes are AS follows:
  - no-export—Do not advertise this route to EBGP peers.
  - no-advertise—Do not advertise this route to any peer.
  - internet—Advertise this route to the Internet community; all routers in the network belong to it.



Note AS loop detection is done by scanning the full AS PATH (AS specified in the AS_PATH attribute), and checking that the AS number of the local system does not appear in the AS PATH. By default, EBGP advertises the learned routes to the same peer to prevent additional CPU cycles on the device in performing loop checks and to avoid delays in the existing outgoing update tasks.

BGP path selection

BGP path selection is a routing process that selects only one path AS the best path when BGP receives multiple advertisements for the same route from different sources, puts the selected path in the IP routing table, and propagates the path to its neighbors using specific criteria in order of precedence.

BGP path selection criteria

BGP uses these criteria, in the order presented, to select a path for a destination:

1. If the path specifies a next hop that is inaccessible, drop the update. Otherwise, prefer the path with the largest weight.
2. If the weights are the same, prefer the path with the largest local preference. If the local preferences are the same, prefer the path that was originated by BGP running on this router.
3. If no route was originated, prefer the route that has the shortest AS_path. If all paths have the same AS_path length, prefer the path with the lowest origin type (where IGP is lower than EGP, and EGP is lower than incomplete).
4. If the origin codes are the same, prefer the path with the lowest MED attribute. If the paths have the same MED, prefer the external path over the internal path.
5. If the paths are still the same, prefer the path through the closest IGP neighbor. Then, determine if multiple paths require installation in the routing table for BGP multipath, on page 898.
6. Determine if multiple paths require installation in the routing table for BGP multipath, on page 898.
7. If both paths are external, prefer the path that was received first (the oldest one).

8. Prefer the path with the lowest IP address, AS specified by the BGP router ID. If the originator or router ID is the same for multiple paths, prefer the path with the minimum cluster list length.
9. Prefer the path that comes from the lowest neighbor address.

BGP multipath

BGP multipath is a BGP feature that

- allows installation into the IP routing table of multiple equal-cost BGP paths to the same destination prefix,
- enables traffic to the destination prefix to be shared across all installed paths, and
- does not affect best-PATH selection algorithm.

BGP multipath PATH requirements

These paths are installed in the table together with the best PATH for load-sharing. BGP multipath does not affect best-PATH selection. For example, a router still designates one of the paths AS the best PATH, according to the algorithm, and advertises this best PATH to its BGP peers.

In order to be candidates for multipath, paths to the same destination need to have these characteristics equal to the best-PATH characteristics:

- Weight
- Local preference
- AS-PATH length
- Origin code
- Multi Exit Discriminator (MED)
- One of these:
  - Neighboring AS or sub-AS (before the addition of the BGP Multipaths)
  - AS-PATH (after the addition of the BGP multipaths)

Some BGP multipath features put additional requirements on multipath candidates:

- The PATH should be learned from an external or confederation-external neighbor (eBGP).
- The IGP metric to the BGP next hop should be equal to the best-PATH IGP metric.

These are the additional requirements for internal BGP (iBGP) multipath candidates:

- The PATH should be learned from an internal neighbor (iBGP).
- The IGP metric to the BGP next hop should be equal to the best-PATH IGP metric, unless the router is configured for unequal-cost iBGP multipath.

BGP inserts up to *n* most recently received paths from multipath candidates into the IP routing table, where *n* is the number of routes to install to the routing table, AS specified when you configure BGP multipath. The default value, when multipath is disabled, is 1.

For unequal-cost load balancing, you can also use BGP Link Bandwidth.

Note The equivalent next-hop-self is performed on the best PATH that is selected among eBGP multipaths before it is forwarded to internal peers.

Guidelines for BGP

Context mode guidelines

Consider these context mode requirements when implementing BGP.

- Supported in single and multiple context mode.
- Only one Autonomous System (AS) number is supported for all contexts.

Firewall mode guidelines

Does not support transparent firewall mode. BGP is supported only in routed mode.

IPv6 guidelines

Supports IPv6. Graceful restart is not supported for IPv6 address family.

Additional guidelines

Consider these additional operational requirements and limitations.

- The system does not add route entry for the IP address received over PPPoE in the CP route table. BGP always looks into CP route table for initiating the TCP session, hence BGP does not form TCP session.

  Thus, BGP over PPPoE is not supported.
- BGP is not supported on management-only or BVI interfaces.
- To avoid adjacency flaps due to route updates being dropped if the route update is larger than the minimum MTU on the link, ensure that you configure the same MTU on the interfaces on both sides of the link.
- BGP with PATH MTU (PMTU) can cause adjacency flaps if MTU discovery fails, especially with ECMP routing. Hence, be cautious while using BGP, PMTU, and ECMP AS packet drops can occur if MTU discovery fails due to any reason.
- The BGP table of the member unit is not synchronized with the control unit table. Only its routing table is synchronized with the control unit routing table.
- Enable the graceful timeout feature when you use BGP on Firewall Threat Defense devices that are configured in high availability mode. Enabling this feature reduces packet drops that occur due to issues related to BGP convergence.

Configure BGP

This section describes how to enable and configure the BGP process on your system.

Procedure

Step 1 Enable BGP, on page 900.
Step 2 Define the Best Path for a BGP Routing Process, on page 902.
Step 3 Configure Policy Lists, on page 903.
Step 4 Configure AS Path Filters, on page 904.
Step 5 Configure Community Rules, on page 904.
Step 6 Configure IPv4 Address Family Settings, on page 905.
Step 7 Configure IPv6 Address Family Settings, on page 918.

Enable BGP

This section describes the steps required to enable BGP routing, establish a BGP routing process and configure general BGP parameters.

Procedure

Step 1 Enable a BGP routing process, which places the ASA in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Valid values for autonomous-num are from 1-4294967295 and 1.0-XX.YY.

Step 2 Discard routes that have as-path segments that exceed a specified value:

bgp maxas-limit number

Example:

```
ciscoasa(config-router)# bgp maxas-limit 15
```

The number argument specifies the maximum number of autonomous system segments, allowed. Valid values are from 1 to 254.

Step 3 Log BGP neighbor resets:

bgp log-neighbor-changes

Step 4 Enable BGP to automatically discover the best TCP path MTU for each BGP session:

bgp transport path-mtu-discovery

Step 5 Enable BGP to terminate external BGP sessions of any directly adjacent peer if the link used to reach the peer goes down; without waiting for the hold-down timer to expire:

bgp fast-external-fallover

Step 6 Allow a BGP routing process to discard updates received from an external BGP (eBGP) peers that do not list their autonomous system (AS) number as the first AS path segment in the AS_PATH attribute of the incoming route:

bgp enforce-first-as

Step 7 Change the default display and regular expression match format of BGP 4-byte autonomous system numbers from asplain (decimal values) to dot notation:

bgp asnotation dot

Step 8 Adjust BGP network timers:

timers bgp keepalive holdtime [min-holdtime]

Example:

```
ciscoasa(config-router)# timers bgp 80 120
```

- keepalive — frequency (in seconds) with which the ASA sends keepalive messages to its peer. The default value 60 seconds.
- holdtime — interval (in seconds) after not receiving a keepalive message that the ASA declares a peer dead. The default is 180 seconds.
- (Optional) min-holdtime — interval (in seconds) after not receiving a keepalive message from a neighbor, that the ASA declares a neighbor dead.

  Note
  A holdtime of less than 20 seconds increases the possibility of peer flapping.

Step 9 Enable BGP graceful restart capability:

bgp graceful-restart [restart-time seconds|stalepath-time seconds][all]

Example:

```
ciscoasa(config-router)# bgp graceful-restart restart-time 200
```

- restart-time — maximum time period (in seconds) that the ASA will wait for a graceful-restart-capable neighbor to return to normal operation after a restart event occurs. The default is 120 seconds. Valid values are from 1 to 3600 seconds.

- stalepath-time — maximum time period (in seconds) that the ASA will hold stale paths for a restarting peer. All stale paths are deleted after this timer expires. The default value is 360 seconds. Valid values are from 1 to 3600 seconds.

Define the Best Path for a BGP Routing Process

This section describes the steps required to configure the BGP best path. For more information on the best path, see BGP path selection, on page 897.

Procedure

Step 1 Enable a BGP routing process, which places the ASA in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Change the default local preference value:

bgp default local-preference number

Example:

```
ciscoasa(config-router)# bgp default local-preference 500
```

The number argument is any value between 0 and 4294967295. Higher values indicate higher preference.

The default value is 100.

Step 3 Enable Multi Exit Discriminator (MED) comparison among paths learned from neighbors in different autonomous systems:

bgp always-compare-med

Step 4 Compare between similar routes received from external BGP (eBGP) peers during the best path selection process and switch the best path to the route with the lowest router ID:

bgp bestpath compare-routerid

Step 5 Select the best MED path advertised from the neighboring AS:

bgp deterministic-med

Step 6 Set a path with a missing MED attribute as the least preferred path:

bgp bestpath med missing-as-worst

Configure Policy Lists

When a policy list is referenced within a route map, all of the match statements within the policy list are evaluated and processed. Two or more policy lists can be configured with a route map. A policy list can also coexist with any other preexisting match and set statements that are configured within the same route map but outside of the policy list. This section describes the steps required to configure policy lists.

Procedure

Step 1 Create a BGP policy list.

policy-list *policy_list_name* {**permit** | **deny**}

The **permit** keyword allows access for matching conditions.

The **deny** keyword denies access for matching conditions.

Example:

```
ciscoasa(config)# policy-list Example-policy-list1 permit
```

Step 2 Distribute routes that have their next hop out of one of the interfaces specified:

match interface [*interface_name* [*interface_name*] […]]

Example:

```
ciscoasa(config-policy-list)# match interface outside
```

Step 3 Redistribute routes by matching either or all of the following: the destination address, next hop router address, and router/access server source:

match ip {**address** | **next-hop** | **route-source**}

Step 4 Match a BGP autonomous system path:

match as-path

Step 5 Match a BGP community:

match community {*community-list_name* | **exact-match**}

- *community-list_name* — one or more community lists.
- **exact-match** — indicates that an exact match is required. All of the communities and only those communities specified must be present.

Example:

```
ciscoasa(config-policy-list)# match community ExampleCommunity1
```

Step 6 Redistribute routes with the metrics specified:

match metric *metric* [*metric* […]]

Step 7 Redistribute routes in the routing table that match the specified tags:

match tag *tag* [*tag* […]]

Configure AS Path Filters

An AS path filter allows you to filter the routing update message by using access lists and look at the individual prefixes within an update message. If a prefix within the update message matches the filter criteria then that individual prefix is filtered out or accepted depending on what action the filter entry has been configured to carry out. This section describes the steps required to configure AS path filters.



Note The as-path access-lists are not the same as the regular firewall ACLs.

Procedure

Configure an autonomous system path filter using a regular expression in the global configuration mode:

as-path access-list acl-number {permit|deny} regexp

Example:

```
ciscoasa(config)# as-path access-list 35 permit testaspath
```

- *acl-number* — AS-path access-list number. Valid values are from 1 to 500.
- regexp — regular expression that defines the AS-path filter. The autonomous system number is expressed in the range from 1 to 65535.

Configure Community Rules

A community is a group of destinations that share some common attribute. You can use community lists to create groups of communities to use in a match clause of a route map. Just like an access list, a series of community lists can be created. Statements are checked until a match is found. As soon as one statement is satisfied, the test is concluded. This section describes the steps required to configure community rules.

Procedure

Create or configure a BGP community list and control access to it:

community-list {standard| community list-name {deny|permit} [community-number] [AA:NN] [internet] [no-advertise][no-export]}| {expanded|expanded list-name {deny| permit}regexp}

Example:

```
ciscoasa(config)# community-list standard excomm1 permit 100 internet no-advertise no-export
```

- standard — configures a standard community list using a number from 1 to 99 to identify one or more permit or deny groups of communities.
- (Optional) community-number — community as a 32-bit number from 1 to 4294967200. A single community can be entered or multiple communities can be entered, each separated by a space.
- AA:NN — an autonomous system number and network number entered in the 4-byte new community format. This value is configured with two 2-byte numbers separated by a colon. A number from 1 to 65535 can be entered for each 2-byte number. A single community can be entered or multiple communities can be entered, each separated by a space.
- (Optional) internet — specifies the Internet community. Routes with this community are advertised to all peers (internal and external).
- (Optional) no-advertise — specifies the no-advertise community. Routes with this community are not advertised to any peer (internal or external).
- (Optional) no-export — specifies the no-export community. Routes with this community are advertised to only peers in the same autonomous system or to only other subautonomous systems within a confederation. These routes are not advertised to external peers.
- (Optional) expanded— configures an expanded community list number from 100 to 500 to identify one or more permit or deny groups of communities.
- regexp — regular expression that defines the AS-path filter. The autonomous system number is expressed in the range from 1 to 65535.

Note
Regular expressions can be used only with expanded community lists.

Configure IPv4 Address Family Settings

The IPv4 settings for BGP can be set up from the IPv4 family option within the BGP configuration setup. The IPv4 family section includes subsections for General settings, Aggregate address settings, Filtering settings and Neighbor settings. Each of these subsections enable you to customize parameters specific to the IPv4 family.

Configure IPv4 Family General Settings

This section describes the steps required to configure the general IPv4 settings.

Procedure

Step 1 Enable a BGP routing process, which places the router in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 4 (IPv4) address prefixes:

address-family ipv4 [unicast]

The keyword unicast specifies IPv4 unicast address prefixes. This is the default, even if not specified.

Step 3 (Optional) Configure a fixed router ID for the local BGP routing process:

bgp router-id A.B.C.D

Example:

```
ciscoasa(config-router-af)# bgp router-id 10.86.118.3
```

The argument A.B.C.D specifies a router identifier in the form of an IP address. If you do not specify a router ID, it is automatically assigned.

Step 4 (Optional) Configure a cluster pool of IP addresses in the Individual Interface (L3) mode:

bgp router-id cluster-pool

Example:

```
ciscoasa(config-router-af)# bgp router-id cp
```

Note
In an L3 cluster, you cannot define a BGP neighbor as one of the cluster pool IP addresses.

Step 5 Configure the administrative distance for BGP routes:

distance bgp external-distance internal-distance local-distance

Example:

```
ciscoasa(config-router-af)# distance bgp 80 180 180
```

- external-distance — administrative distance for external BGP routes. Routes are external when learned from an external autonomous system. The range of values for this argument are from 1 to 255.
- internal-distance — administrative distance for internal BGP routes. Routes are internal when learned from peer in the local autonomous system. The range of values for this argument are from 1 to 255.
- local-distance — administrative distance for local BGP routes. Local routes are those networks listed with a network router configuration command, often as back doors, for the router or for the networks that is being redistributed from another process. The range of values for this argument are from 1 to 255.

Step 6 Modify metric and tag values when the IP routing table is updated with BGP learned routes:

table-map {WORD|route-map_name}

Example:

```
ciscoasa(config-router-af)# table-map example1
```

The argument route-map_name specifies the route map name from the route-map command.

Step 7 Configure a BGP routing process to distribute a default route (network 0.0.0.0):

default-information originate

Step 8 Configure automatic summarization of subnet routes into network-level routes:

auto-summary

Step 9 Suppress the advertisement of routes that are not installed in the routing information base (RIB):

bgp suppress-inactive

Step 10 Synchronize between BGP and your Interior Gateway Protocol (IGP) system:

synchronization

Step 11 Configure iBGP redistribution into an IGP, such as OSPF:

bgp redistribute-internal

Step 12 Configure scanning intervals of BGP routers for next hop validation:

bgp scan-time scanner-interval

Example:

```
ciscoasa(config-router-af)# bgp scan-time 15
```

The argument scanner-interval specifies scanning interval of BGP routing information. Valid values are from 5 to 60 seconds. The default is 60 seconds.

Step 13 Configure BGP next-hop address tracking:

bgp nexthop trigger {delay seconds|enable}

Example:

```
ciscoasa(config-router-af)# bgp nexthop trigger delay 15
```

- trigger — specifies the use of BGP next-hop address tracking. Use this keyword with the delay keyword to change the next-hop tracking delay. Use this keyword with the enable keyword to enable next-hop address tracking.
- delay — changes the delay interval between checks on updated next-hop routes installed in the routing table.
- seconds — specifies the delay in seconds. Range is from 0 to 100. Default is 5.
- enable — enables BGP next-hop address tracking immediately.

Step 14 Control the maximum number of parallel iBGP routes that can be installed in a routing table:

maximum-paths {number_of_paths|ibgp number_of_paths}

Example:

```
ciscoasa(config-router-af)# maximum-paths ibgp 2
```

Note
If the ibgp keyword is not used, then the number_of_paths argument controls the maximum number of parallel EBGP routes.

The number_of_paths argument specifies the number of routes to install to the routing table. Valid values are between 1 and 8.

Configure IPv4 Family Aggregate Address Settings

This section describes the steps required to define the aggregation of specific routes into one route.

Procedure

Step 1 Enable a BGP routing process, which places the ASA in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 4 (IPv4) address prefixes:

address-family ipv4 [unicast]

The keyword unicast specifies IPv4 unicast address prefixes. This is the default, even if not specified.

Step 3 Create an aggregate entry in a BGP database:

aggregate-address address mask [as-set][summary-only][suppress-map map-name][advertise-map map-name][attribute-map map-name]

Example:

```
ciscoasa(config-router-af) aggregate-address 10.86.118.0 255.255.255.0 as-set summary-only
 suppress-map example1 advertise-map example1 attribute-map example1
```

- address — the aggregate address.
- mask — the aggregate mask.
- map-name — the route map.
- (Optional) as-set — generates autonomous system set path information.

- (Optional) summary-only — filters all more-specific routes from updates.
- (Optional) Suppress-map map-name —specifies the name of the route map used to select the routes to be suppressed.
- (Optional) Advertise-map map-name — specifies the name of the route map used to select the routes to create AS_SET origin communities.
- (Optional) Attribute-map map-name — specifies the name of the route map used to set the attribute of the aggregate route.

Configure IPv4 Family Filtering Settings

This section describes the steps required to filter routes or networks received in incoming BGP updates.

Procedure

Step 1 Enable a BGP routing process and enter router configuration mode:

router bgp *autonomous-num*

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 4 (IPv4) address prefixes:

address-family ipv4 [**unicast**]

The keyword unicast specifies IPv4 unicast address prefixes. This is the default, even if not specified.

Step 3 Filter routes or networks received in incoming or advertised in outgoing BGP updates:

distribute-list *acl-number* {**in** | **out**} [*protocol process-number* | **connected** | **static**]

The argument *acl-number* specifies IP access list number. The access list defines which networks are to be received and which are to be suppressed in routing updates.

The keyword **in** specifies that the filter must be applied to incoming BGP updates and **out** specifies that the filter must be applied to outgoing BGP updates.

For outbound filters, you can optionally specify a protocol (**bgp**, **eigrp**, **ospf**, or **rip**) with a process number (except for RIP) to apply to the distribution list. You can also filter on whether the peers and networks were learned through **connected** or **static** routes.

Example:

```
ciscoasa(config-router-af)# distribute-list ExampleAcl in bgp 2
```

Configure IPv4 Family BGP Neighbor Settings

This section describes the steps required to define BGP neighbors and neighbor settings.

Procedure

Step 1 Enable a BGP routing process, which places the router in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 4 (IPv4) address prefixes:

address-family ipv4 [unicast]

The keyword unicast specifies IPv4 unicast address prefixes. This is the default, even if not specified.

Step 3 Add an entry to the BGP neighbor table:

neighbor ip-address remote-as autonomous-number

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 remote-as 3
```

Step 4 (Optional) Disable a neighbor or peer group:

neighbor ip-address shutdown

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 shutdown 3
```

Step 5 Exchange information with a BGP neighbor:

neighbor ip-address activate

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 activate
```

Step 6 Enable or disable the Border Gateway Protocol (BGP) graceful restart capability for a BGP neighbor:

neighbor ip-address ha-mode graceful-restart [disable]

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 ha-mode graceful-restart
```

(Optional) The disable keyword disables BGP graceful restart capability for a neighbor.

Step 7 Distribute BGP neighbor information as specified in an access list:

neighbor {ip-address} distribute-list {access-list-name}{in|out}

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 distribute-list ExampleAcl in
```

- access-list-number — the number of a standard or extended access list. The range of a standard access list number is from 1 to 99. The range of an extended access list number is from 100 to 199.
- expanded-list-number — the number of an expanded access list number. The range of an expanded access list is from 1300 to 2699.
- access-list-name — the name of a standard or extended access list.
- prefix-list-name — the name of a BGP prefix list.
- in — the access list is applied to incoming advertisements to that neighbor.
- out — that the access list is applied to outgoing advertisements to that neighbor.

Step 8 Apply a route map to incoming or outgoing routes:

neighbor {ip-address} route-map map-name {in|out}

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 route-map example1 in
```

The keyword in applies a route map to incoming routes.

The keyword out applies a route map to outgoing routes.

Step 9 Distribute BGP neighbor information as specified in a prefix list:

neighbor {ip-address} prefix-list prefix-list-name {in|out}

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 prefix-list NewPrefixList in
```

The keyword in implies that the prefix list is applied to incoming advertisements from that neighbor.

The keyword out implies that the prefix list is applied to outgoing advertisements to that neighbor.

Step 10 Set up a filter list:

neighbor {ip-address} filter-list access-list-number {in|out}

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 filter-list 5 in
```

- access-list-name — specifies the number of an autonomous system path access list. You define this access list with the ip as-path access-list command.
- in — that the access list is applied to incoming advertisements from that neighbor.

- out — that the access list is applied to outgoing advertisements to that neighbor.

Step 11 Control the number of prefixes that can be received from a neighbor:

neighbor {ip-address} maximum-prefix maximum [threshold][restart restart interval][warning-only]

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 maximum-prefix 7 75 restart 12
```

- maximum — the maximum number of prefixes allowed from this neighbor.
- (Optional) threshold — integer specifying at what percentage of maximum the router starts to generate a warning message. The range is from 1 to 100; the default is 75 (percent).
- (Optional) restart interval — integer value (in minutes) that specifies the time interval after which the BGP neighbor restarts.
- (Optional) warning-only — allows the router to generate a log message when the maximum number of prefixes is exceeded, instead of terminating the peering.

Step 12 Allow a BGP speaker (the local router) to send the default route 0.0.0.0 to a neighbor for use as a default route:

neighbor {ip-address} default-originate [route-map map-name]

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 default-originate route-map example1
```

The argument map-name is the name of the route-map. The route map allows route 0.0.0.0 to be injected conditionally.

Step 13 Set the minimum interval between the sending of BGP routing updates:

neighbor {ip-address} advertisement-interval seconds

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 advertisement-interval 15
```

The argument seconds is the time (in seconds). Valid values are from 0 to 600.

Step 14 Advertise the routes in the BGP table that matches the configured route-map:

neighbor {ip-address} advertise-map map-name {exist-map map-name |non-exist-map map-name}[check-all-paths]

Example:

```
ciscoasa(config-router-af)# neighbor 10.2.1.1 advertise-map MAP1 exist-map MAP2
```

- advertise-map map name — the name of the route map that will be advertised if the conditions of the exist map or non-exist map are met.

- exist-map map name — the name of the exist-map that is compared with the routes in the BGP table to determine whether the advertise-map route is advertised or not.
- non-exist-map map name — the name of the non-exist-map that is compared with the routes in the BGP table to determine whether the advertise-map route is advertised or not.
- (Optional) check all paths — enables checking of all paths by the exist-map with a prefix in the BGP table.

Step 15 Remove private autonomous system numbers from outbound routing updates:

neighbor {ip-address} remove-private-as

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 remove-private-as
```

Step 16 Sets the timers for a specific BGP peer or peer group.

neighbor {ip-address} timers keepalive holdtime min holdtime

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 timers 15 20 12
```

- keepalive — the frequency (in seconds) with which the ASA sends keepalive messages to its peer. The default is 60 seconds. Valid values are from 0 to 65535.
- holdtime — the interval (in seconds) after not receiving a keepalive message that the ASA declares a peer dead. The default is 180 seconds.
- min holdtime — the minimum interval (in seconds) after not receiving a keepalive message that the ASA declares a peer dead.

  Note
  A holdtime of less than 20 seconds increases the possibility of peer flapping.

Step 17 Enable Message Digest 5 (MD5) authentication on a TCP connection between two BGP peers:

neighbor {ip-address} password string

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 password test
```

The argument string is a case-sensitive password of up to 25 characters when the service password-encryption command is enabled and up to 81 characters when the service password-encryption command is not enabled. The string can contain any alphanumeric characters, including spaces.

Note
When you set the first character of the password as a number, do not provide a space immediately after the number. That is, you cannot specify a password in the format number-space-anything. The space after the number can cause authentication to fail.

Step 18 Specify that communities attributes should be sent to a BGP neighbor:

neighbor {ip-address} send-community

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 send-community
```

Step 19 Configure the router as the next hop for a BGP-speaking neighbor or peer group:

neighbor {ip-address}next-hop-self

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 next-hop-self
```

Step 20 Accept and attempt BGP connections to external peers residing on networks that are not directly connected:

neighbor {ip-address} ebgp-multihop [ttl]

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 ebgp-multihop 5
```

The argument ttl specifies time-to-live in the range from 1 to 255 hops.

Step 21 Disable connection verification to establish an eBGP peering session with a single-hop peer that uses a loopback interface:

neighbor {ip-address} disable-connected-check

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 disable-connected-check
```

Step 22 Secure a BGP peering session and configures the maximum number of hops that separate two external BGP (eBGP) peers:

neighbor {ip-address} ttl-security hops hop-count

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 ttl-security hops 15
```

The argument hop-count is the number of hops that separate the eBGP peers. The TTL value is calculated by the router from the configured hop-count argument. Valid values are from 1 to 254.

Step 23 Assign a weight to a neighbor connection:

neighbor {ip-address} weight number

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 weight 30
```

The argument number is the weight to assign to a neighbor connection. Valid values are from 0 to 65535.

Step 24 Configure the ASA to accept only a particular BGP version:

neighbor {ip-address} version number

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 version 4
```

The argument number specifies the BGP version number. The version can be set to 2 to force the software to use only Version 2 with the specified neighbor. The default is to use Version 4 and dynamically negotiate down to Version 2 if requested.

Step 25 Enable a TCP transport session option for a BGP session:

neighbor {ip-address} transport {connection-mode{active|passive}| path-mtu-discovery[disable]}

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 transport path-mtu-discovery
```

- connection-mode — the type of connection (active or passive).
- path-mtu-discovery — enables TCP transport path maximum transmission unit (MTU) discovery. TCP path MTU discovery is enabled by default.
- (Optional) disable — disables TCP path MTU discovery.

Step 26 Customize the AS_PATH attribute for routes received from an external Border Gateway Protocol (eBGP) neighbor:

neighbor {ip-address} local-as [autonomous-system-number[no-prepend]]

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 local-as 5 no-prepend replace-as
```

- (Optional) autonomous-system-number — the number of an autonomous system to prepend to the AS_PATH attribute. The range of values for this argument is any valid autonomous system number from 1 to 4294967295 or 1.0 to XX.YY.
- (Optional) no-prepend — does not prepend the local autonomous system number to any routes received from the eBGP neighbor.

Configure IPv4 Network Settings

This section describes the steps required to define the networks to be advertised by the BGP routing process.

Procedure

Step 1 Enable a BGP routing process, which places the ASA in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 4 (IPv4) address prefixes:

address-family ipv4 [unicast]

The keyword unicast specifies IPv4 unicast address prefixes. This is the default, even if not specified.

Step 3 Specify the networks to be advertised by the BGP routing processes:

network {network-number [mask network-mask]}[route-map map-tag]

Example:

```
ciscoasa(config-router-af)# network 10.86.118.13 mask 255.255.255.255 route-map example1
```

Note
For a network prefix to be advertised, a route to the device must exist on the routing table.

- network-number — the network that BGP will advertise.
- (Optional) network-mask — the network or subnetwork mask with mask address.
- (Optional) map-tag — the identifier of a configured route map. The route map should be examined to filter the networks to be advertised. If not specified, all networks are advertised.

Configure IPv4 Redistribution Settings

This section describes the steps required to define the conditions for redistributing routes from another routing domain into BGP.

Procedure

Step 1 Enable a BGP routing process, which places the ASA in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 4 (IPv4) address prefixes:

address-family ipv4 [unicast]

Example:

```
ciscoasa(config-router)# address-family ipv4[unicast]
```

The keyword unicast specifies IPv4 unicast address prefixes. This is the default, even if not specified.

Step 3 Redistribute routes from another routing domain into a BGP autonomous system:

redistribute protocol [process-id] [metric] [route-map [map-tag]]

Example:

```
ciscoasa(config-router-af)# redistribute ospf 2 route-map example1 match external
```

- protocol — the source protocol from which routes are being redistributed. It can be one of the following: Connected, EIGRP, OSPF, RIP or Static.
- (Optional) process-id — a name for the specific routing process.
- (Optional) metric — the metric for the redistributed route.
- (Optional) map-tag — the identifier of a configured route map.

Note
The route map should be examined to filter the networks to be redistributed. If not specified, all networks are redistributed.

Configure IPv4 Route Injection Settings

This section describes the steps required to define the routes to be conditionally injected into the BGP routing table.

Procedure

Step 1 Enable a BGP routing process, which places the ASA in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 4 (IPv4) address prefixes:

address-family ipv4 [unicast]

Example:

```
ciscoasa(config-router)# address-family ipv4[unicast]
```

The keyword unicast specifies IPv4 unicast address prefixes. This is the default, even if not specified.

Step 3 Configure conditional route injection to inject more specific routes into a BGP routing table:

bgp inject-map inject-map exist-map exist-map [copy-attributes]

Example:

```
ciscoasa(config-router-af)# bgp inject-map example1 exist-map example2 copy-attributes
```

- inject-map — the name of the route map that specifies the prefixes to inject into the local BGP routing table.
- exist-map — the name of the route map containing the prefixes that the BGP speaker will track.
- (Optional) copy-attributes — configures the injected route to inherit attributes of the aggregate route.

Configure IPv6 Address Family Settings

The IPv6 settings for BGP can be set up from the IPv6 family option within the BGP configuration setup. The IPv6 family section includes subsections for General settings, Aggregate address settings and Neighbor settings. Each of these subsections enable you to customize parameters specific to the IPv6 family.

This section describes how to customize the BGP IPv6 family settings.

Configure IPv6 Family General Settings

This section describes the steps required to configure the general IPv6 settings.

Procedure

Step 1 Enable a BGP routing process, which places the router in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 6 (IPv6) address prefixes:

address-family ipv6 [unicast]

Step 3 Configure the administrative distance for BGP routes:

distance bgp external-distance internal-distance local-distance

Example:

```
ciscoasa(config-router-af)# distance bgp 80 180 180
```

- external-distance — administrative distance for external BGP routes. Routes are external when learned from an external autonomous system. The range of values for this argument are from 1 to 255.
- internal-distance — administrative distance for internal BGP routes. Routes are internal when learned from peer in the local autonomous system. The range of values for this argument are from 1 to 255.
- local-distance — administrative distance for local BGP routes. Local routes are those networks listed with a network router configuration command, often as back doors, for the router or for the networks that is being redistributed from another process. The range of values for this argument are from 1 to 255.

Step 4 (Optional) Configure a BGP routing process to distribute a default route (network 0.0.0.0):

default-information originate

Step 5 (Optional) Suppress the advertisement of routes that are not installed in the routing information base (RIB):

bgp suppress-inactive

Step 6 Synchronize between BGP and your Interior Gateway Protocol (IGP) system:

synchronization

Step 7 Configure iBGP redistribution into an IGP, such as OSPF:

bgp redistribute-internal

Step 8 Configure scanning intervals of BGP routers for next hop validation:

bgp scan-time scanner-interval

Example:

```
ciscoasa(config-router-af)# bgp scan-time 15
```

Valid values for the scanner-interval argument from 5 to 60 seconds. The default is 60 seconds.

Step 9 Control the maximum number of parallel iBGP routes that can be installed in a routing table:

maximum-paths {number_of_paths|ibgp number_of_paths}

Example:

```
ciscoasa(config-router-af)# maximum-paths ibgp 2
```

Valid values for the number_of_paths argument is between 1 and 8.

If the ibgp keyword is not used, then the number_of_paths argument controls the maximum number of parallel EBGP routes.

Configure IPv6 Family Aggregate Address Settings

This section describes the steps required to define the aggregation of specific routes into one route.

Procedure

Step 1 Enable a BGP routing process, which places the ASA in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 6 (IPv6) address prefixes:

address-family ipv6 unicast

Step 3 Create an aggregate entry in a BGP database:

aggregate-address ipv6-address/cidr [as-set][summary-only][suppress-map map-name][advertise-map ipv6-map-name][attribute-map map-name]

Example:

```
ciscoasa(config-router-af) aggregate-address 2000::1/8 summary-only
```

- address — the aggregate IPv6 address.
- (Optional) as-set — generates autonomous system set path information.
- (Optional) summary-only — filters all more-specific routes from updates.
- (Optional) suppress-map map-name — specifies the name of the route map used to select the routes to be suppressed.
- (Optional) advertise-map map-name — specifies the name of the route map used to select the routes to create AS_SET origin communities.
- (Optional) attribute-map map-name — specifies the name of the route map used to set the attribute of the aggregate route.

Step 4 Set the interval at which BGP routes will be aggregated:

bgp aggregate-timer seconds

Example:

```
ciscoasa(config-router-af)bgp aggregate-timer 20
```

Configure IPv6 Family BGP Neighbor Settings

This section describes the steps required to define BGP neighbors and neighbor settings.

Procedure

Step 1 Enable a BGP routing process, which places the router in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 6 (IPv6) address prefixes:

address-family ipv6 [unicast]

Step 3 Add an entry to the BGP neighbor table:

neighbor ipv6-address remote-as autonomous-number

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1/8 remote-as 3
```

The argument ipv6-address specifies the IPv6 address of the next hop that can be used to reach the specified network.TheIPv6 address of the next hop need not be directly connected; recursion is done to find the IPv6 address of the directly connected next hop.When an interface type and interface number are specified, you can optionally specify the IPv6 address of the next hop to which packets are output.You must specify an interface type and an interface number when using a link-local address as the next hop (the link-local next hop must also be an adjacent device).

Note
This argument must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons.

Step 4 (Optional) Disable a neighbor or peer group:

neighbor ipv6-address shutdown

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1/8 shutdown 3
```

Step 5 Exchange information with a BGP neighbor:

neighbor ipv6-address activate

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1/8 activate
```

Step 6 Apply a route map to incoming or outgoing routes:

neighbor {ipv6-address} route-map map-name {in|out}

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 route-map example1 in
```

The keyword in applies a route map to incoming routes.

The keyword out applies a route map to outgoing routes.

Step 7 Distribute BGP neighbor information as specified in a prefix list:

neighbor {ipv6-address} prefix-list prefix-list-name {in|out}

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 prefix-list NewPrefixList in
```

The keyword in implies that the prefix list is applied to incoming advertisements from that neighbor.

The keyword out implies that the prefix list is applied to outgoing advertisements to that neighbor.

Step 8 Set up a filter list:

neighbor {ipv6-address} filter-list access-list-name {in|out}

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 filter-list 5 in
```

- access-list-name — specifies the number of an autonomous system path access list. You define this access list with the ip as-path access-list command.
- in — that the access list is applied to incoming advertisements from that neighbor.
- out — that the access list is applied to outgoing advertisements to that neighbor.

Step 9 Control the number of prefixes that can be received from a neighbor:

neighbor {ipv6-address} maximum-prefix maximum [threshold][restart restart interval][warning-only]

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 maximum-prefix 7 75 restart 12
```

- maximum — the maximum number of prefixes allowed from this neighbor.
- (Optional) threshold — integer specifying at what percentage of maximum the router starts to generate a warning message. The range is from 1 to 100; the default is 75 (percent).
- (Optional) restart interval — integer value (in minutes) that specifies the time interval after which the BGP neighbor restarts.
- (Optional) warning-only — allows the router to generate a log message when the maximum number of prefixes is exceeded, instead of terminating the peering.

Step 10 Allow a BGP speaker (the local router) to send the default route 0.0.0.0 to a neighbor for use as a default route:

neighbor {ipv6-address} default-originate [route-map map-name]

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 default-originate route-map example1
```

The argument map-name is the name of the route-map.The route map allows route 0.0.0.0 to be injected conditionally.

Step 11 Set the minimum interval between the sending of BGP routing updates:

neighbor {ipv6-address} advertisement-interval seconds

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 advertisement-interval 15
```

The argument seconds is the time (in seconds). Valid values are from 0 to 600.

Step 12 Remove private autonomous system numbers from outbound routing updates:

neighbor {ipv6-address} remove-private-as

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 remove-private-as
```

Step 13 Advertise the routes in the BGP table that matches the configured route-map:

neighbor {ipv6-address} advertise-map map-name {exist-map map-name |non-exist-map map-name}[check-all-paths]

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 advertise-map MAP1 exist-map MAP2
```

- advertise-map map name — the name of the route map that will be advertised if the conditions of the exist map or non-exist map are met.
- exist-map map name — the name of the exist-map that is compared with the routes in the BGP table to determine whether the advertise-map route is advertised or not.
- non-exist-map map name — the name of the non-exist-map that is compared with the routes in the BGP table to determine whether the advertise-map route is advertised or not.
- (Optional) check all paths — enables checking of all paths by the exist-map with a prefix in the BGP table.

Step 14 Sets the timers for a specific BGP peer or peer group.

neighbor {ipv6-address} timers keepalive holdtime min holdtime

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 timers 15 20 12
```

- keepalive — the frequency (in seconds) with which the ASA sends keepalive messages to its peer. The default is 60 seconds. Valid values are from 0 to 65535.
- holdtime — the interval (in seconds) after not receiving a keepalive message that the ASA declares a peer dead. The default is 180 seconds.
- min holdtime — the minimum interval (in seconds) after not receiving a keepalive message that the ASA declares a peer dead.

Note
A holdtime of less than 20 seconds increases the possibility of peer flapping.

Step 15 Enable Message Digest 5 (MD5) authentication on a TCP connection between two BGP peers:

neighbor {ipv6-address} password string

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 password test
```

The argument string is a case-sensitive password of up to 25 characters when the service password-encryption command is enabled and up to 81 characters when the service password-encryption command is not enabled. The string can contain any alphanumeric characters, including spaces.

Note
When you set the first character of the password as a number, do not provide a space immediately after the number. That is, you cannot specify a password in the format number-space-anything. The space after the number can cause authentication to fail.

Step 16 Specify that communities attributes should be sent to a BGP neighbor:

neighbor {ipv6-address} send-community [standard]

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 send-community
```

(Optional) standard keyword — only standard communities will be sent.

Step 17 Configure the router as the next hop for a BGP-speaking neighbor or peer group:

neighbor {ipv6-address}next-hop-self

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 next-hop-self
```

Step 18 Accept and attempt BGP connections to external peers residing on networks that are not directly connected:

neighbor {ipv6-address} ebgp-multihop [ttl]

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 ebgp-multihop 5
```

The argument ttl specifies time-to-live in the range from 1 to 255 hops.

Step 19 Disable connection verification to establish an eBGP peering session with a single-hop peer that uses a loopback interface:

neighbor {ipv6-address} disable-connected-check

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 disable-connected-check
```

Step 20 Secure a BGP peering session and configures the maximum number of hops that separate two external BGP (eBGP) peers:

neighbor {ipv6-address} ttl-security hops hop-count

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 ttl-security hops 15
```

The argument hop-count is the number of hops that separate the eBGP peers. The TTL value is calculated by the router from the configured hop-count argument. Valid values are from 1 to 254.

Step 21 Assign a weight to a neighbor connection:

neighbor {ipv6-address} weight number

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 weight 30
```

The argument number is the weight to assign to a neighbor connection. Valid values are from 0 to 65535.

Step 22 Configure the ASA to accept only a particular BGP version:

neighbor {ipv6-address} version number

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 version 4
```

The argument number specifies the BGP version number. The default is Version 4. Currently only BGP version 4 is supported.

Step 23 Enable a TCP transport session option for a BGP session:

neighbor {ipv6-address} transport {connection-mode{active|passive}| path-mtu-discovery[disable]}

Example:

```
ciscoasa(config-router-af)# neighbor 2000::1 transport connection-mode active
```

- connection-mode — the type of connection (active or passive).
- path-mtu-discovery — enables TCP transport path maximum transmission unit (MTU) discovery. TCP path MTU discovery is enabled by default.

- (Optional) disable — disables TCP path MTU discovery.

Step 24 Customize the AS_PATH attribute for routes received from an external Border Gateway Protocol (eBGP) neighbor:

neighbor {ipv6-address} local-as [autonomous-system-number[no-prepend]]

Example:

```
ciscoasa(config-router-af)# neighbor 10.86.118.12 local-as 5 no-prepend replace-as
```

- (Optional) autonomous-system-number — the number of an autonomous system to prepend to the AS_PATH attribute. The range of values for this argument is any valid autonomous system number from 1 to 4294967295 or 1.0 to XX.YY.
- (Optional) no-prepend — does not prepend the local autonomous system number to any routes received from the eBGP neighbor.

Caution
BGP prepends the autonomous system number from each BGP network that a route traverses to maintain network reachability information and to prevent routing loops. This command should be configured only for autonomous system migration, and should be removed after the transition has been completed. This procedure should be attempted only by an experienced network operator. Routing loops can be created through improper configuration.

Configure IPv6 Network Settings

This section describes the steps required to define the networks to be advertised by the BGP routing process.

Procedure

Step 1 Enable a BGP routing process, which places the ASA in router configuration mode:

router bgp *autonomous-num*

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 6 (IPv6) address prefixes:

address-family ipv6 [unicast]

Step 3 Specify the networks to be advertised by the BGP routing processes:

network {*prefix_delegation_name* [*subnet_prefix*/*prefix_length*] | *ipv6_prefix*/*prefix_length*} [**route-map** *route_map_name*]

Example:

```
ciscoasa(config-router-af)# network 2001:1/64 route-map test_route_map
ciscoasa(config-router-af)# network outside-prefix 1::/64
ciscoasa(config-router-af)# network outside-prefix 2::/64
```

- *prefix_delegation_name*—If you enable the DHCPv6 Prefix Delegation client (**ipv6 dhcp client pd**), then you can advertise the prefix(es). To subnet the prefix, specify the *subnet_prefix*/*prefix_length*.
- *ipv6 network*/*prefix_length*— The network that BGP will advertise.
- (Optional) **route-map** *name*— The identifier of a configured route map. The route map should be examined to filter the networks to be advertised. If not specified, all networks are advertised.

Configure IPv6 Redistribution Settings

This section describes the steps required to define the conditions for redistributing routes from another routing domain into BGP.

Procedure

Step 1 Enable a BGP routing process, which places the ASA in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 6 (IPv6) address prefixes:

address-family ipv6 [unicast]

Example:

```
ciscoasa(config-router)# address-family ipv6[unicast]
```

Step 3 Redistribute routes from another routing domain into a BGP autonomous system:

```
redistribute protocol [process-id][autonomous-num][metric metric value][match{internal|
external1|external2|NSSA external 1|NSSA external 2}][route-map [map-tag]][subnets]
```

Example:

```
ciscoasa(config-router-af)# redistribute ospf 2 route-map example1 match external
```

- protocol — the source protocol from which routes are being redistributed. It can be one of the following: Connected, EIGRP, OSPF, RIP or Static.

- (Optional) process-id — For the ospf protocol, this is an appropriate OSPF process ID from which routes are to be redistributed. This identifies the routing process. This value takes the form of a nonzero decimal number.

  Note
  This value is auto-populated for the other protocols.

- (Optional) metric metric value — When redistributing from one OSPF process to another OSPF process on the same router, the metric will be carried through from one process to the other if no metric value is specified. When redistributing other processes to an OSPF process, the default metric is 20 when no metric value is specified. The default value is 0.
- (Optional) match internal | external1 | external2 | NSSA external 1 | NSSA external 2 — For the criteria by which OSPF routes are redistributed into other routing domains. It can be one of the following:
  - internal — Routes that are internal to a specific autonomous system.
  - external 1 — Routes that are external to the autonomous system, but are imported into BGP as OSPF Type 1 external route.
  - external 2 — Routes that are external to the autonomous system, but are imported into BGP as OSPF Type 2 external route.
  - NSSA external 1 — Routes that are external to the autonomous system, but are imported into BGP as OSPF NSSA Type 1 external route.
  - NSSA external 2 — Routes that are external to the autonomous system, but are imported into BGP as OSPF NSSA Type 2 external route.
- (Optional) map-tag — the identifier of a configured route map.

Note
The route map should be examined to filter the networks to be redistributed. If not specified, all networks are redistributed

Configure IPv6 Route Injection Settings

This section describes the steps required to define the routes to be conditionally injected into the BGP routing table.

Procedure

Step 1 Enable a BGP routing process, which places the ASA in router configuration mode:

router bgp autonomous-num

Example:

```
ciscoasa(config)# router bgp 2
```

Step 2 Enter address family configuration mode to configure a routing session using standard IP Version 6 (IPv6) address prefixes:

address-family ipv6 [unicast]

Example:

```
ciscoasa(config-router)# address-family ipv6 [unicast]
```

Step 3 Configure conditional route injection to inject more specific routes into a BGP routing table:

bgp inject-map inject-map exist-map exist-map [copy-attributes]

Example:

```
ciscoasa(config-router-af)# bgp inject-map example1 exist-map example2 copy-attributes
```

- inject-map — the name of the route map that specifies the prefixes to inject into the local BGP routing table.
- exist-map — the name of the route map containing the prefixes that the BGP speaker will track.
- (Optional) copy-attributes — configures the injected route to inherit attributes of the aggregate route.

Monitoring BGP

You can use the following commands to monitor the BGP routing process. For examples and descriptions of the command output, see the command reference. Additionally, you can disable the logging of neighbor change messages and neighbor warning messages.

To monitor various BGP routing statistics, enter one of the following commands:

- **show bgp** [ip-address [mask [longer-prefixes [injected] | shorter-prefixes [length]]]| prefix-list name | route-map name]

  Displays the entries in the BGP routing table.

- **show bgp** cidr-only

  Displays routes with non-natural network masks (that is, classless interdomain routing, or CIDR).

- **show bgp community community-number [exact-match][no-advertise][no-export]**

  Display routes that belong to specified BGP communities.

- **show bgp community-list community-list-name [exact-match]**

  Displays routes that are permitted by the BGP community list.

- **show bgp filter-list** *access-list-number*

  Displays routes that conform to a specified filter list.

- **show bgp injected-paths**

  Displays all the injected paths in the BGP routing table.

- **show bgp ipv4 unicast**

  Displays entries in the IP version 4 (IPv4) BGP routing table for unicast sessions.

- show bgp ipv6 unicast

  Displays entries in the IPv6 Border Gateway Protocol (BGP) routing table.

- show bgp ipv6 community

  Displays routes that belong to specified IPv6 Border Gateway Protocol (BGP) communities.

- show bgp ipv6 community-list

  Displays routes that are permitted by the IPv6 Border Gateway Protocol (BGP) community list.

- show bgp ipv6 filter-list

  Display routes that conform to a specified IPv6 filter list.

- show bgp ipv6 inconsistent-as

  Displays IPv6 Border Gateway Protocol (BGP) routes with inconsistent originating autonomous systems.

- show bgp ipv6 neighbors

  Displays information about IPv6 Border Gateway Protocol (BGP) connections to neighbors.

- show bgp ipv6 paths

  Displays all the IPv6 Border Gateway Protocol (BGP) paths in the database.

- show bgp ipv6 prefix-list

  Displays routes that match a prefix list.

- show bgp ipv6 quote-regexp

  Displays IPv6 Border Gateway Protocol (BGP) routes matching the autonomous system path regular expression as a quoted string of characters.

- show bgp ipv6 regexp

  Displays IPv6 Border Gateway Protocol (BGP) routes matching the autonomous system path regular expression.

- show bgp ipv6 route-map

  Displays IPv6 Border Gateway Protocol (BGP) routes that failed to install in the routing table.

- show bgp ipv6 summary

  Displays the status of all IPv6 Border Gateway Protocol (BGP) connections.

- show bgp neighbors ip_address

  Displays information about BGP and TCP connections to neighbors.

- show bgp paths [LINE]

  Displays all the BGP paths in the database.

- show bgp pending-prefixes

  Displays prefixes that are pending deletion.

- show bgp prefix-list prefix_list_name [WORD]

  Displays routes that match a specified prefix list.

- show bgp regexp regexp

  Displays routes that match the autonomous system path regular expression.

- show bgp replication [index-group | ip-address]

  Displays update replication statistics for BGP update groups.

- show bgp rib-failure

  Displays BGP routes that failed to install in the Routing Information Base (RIB) table.

- show bgp route-map map-name

  Displays entries in the BGP routing table, based on the route map specified.

- show bgp summary

  Display the status of all BGP connections.

- show bgp system-config

  Display the system context specific BGP configuration in multi-context mode.

  This command is available in all user contexts in multi-context mode.

- show bgp update-group

  Display information about the BGP update groups.



Note To disable BGP Log messages, enter the **no bgp log-neighbor-changes** command in the router configuration mode. This disables the logging of neighbor change messages. Enter this command in router configuration mode for the BGP routing process. By default, neighbor changes are logged.

Example for BGP

This example shows how to enable and configure BGPv4 with various optional processes.

1. Define the conditions for redistributing routes from one routing protocol into another, or enable policy routing:

```
ciscoasa(config)# route-map mymap2 permit 10
```

2. Redistribute any routes that have a route address or match packet that is passed by one of the access lists specified:

```
ciscoasa(config-route-map)# match ip address acl_dmz1 acl_dmz2
```

3. Indicate where to output packets that pass a match clause of a route map for policy routing:

```
ciscoasa(config-route-map)# set ip next-hop peer address
```

4. Enable a BGP routing process from the global configuration mode:

```
ciscoasa(config)# router bgp 2
```

5. Configure a fixed router ID for the local Border Gateway Protocol (BGP) routing process in the address family configuration mode:

```
ciscoasa(config)# address-family ipv4
ciscoasa(config-router-af)# bgp router-id 19.168.254.254
```

6. Add an entry to the BGP neighbor table:

```
ciscoasa(config-router-af)# neighbor 10.108.0.0 remote-as 65
```

7. Apply a route map to incoming or outgoing routes:

```
ciscoasa(config-router-af)# neighbor 10.108.0.0 route-map mymap2 in
```

This example shows how to enable and configure BGPv6 with various optional processes.

1. Define the conditions for redistributing routes from one routing protocol into another, or enable policy routing:

```
ciscoasa(config)# route-map mymap1 permit 10
```

2. Redistribute any routes that have a route address or match packet that is passed by one of the access lists specified:

```
ciscoasa(config-route-map)# match ipv6 address acl_dmz1 acl_dmz2
```

3. Indicate where to output packets that pass a match clause of a route map for policy routing:

```
ciscoasa(config-route-map)# set ipv6 next-hop peer address
```

4. Enable a BGP routing process from the global configuration mode:

```
ciscoasa(config)# router bgp 2
```

5. Configure a fixed router ID for the local Border Gateway Protocol (BGP) routing process in the address family configuration mode:

```
ciscoasa(config)# address-family ipv4
ciscoasa(config-router-af)# bgp router-id 19.168.254.254
```

6. Enter address family configuration mode to configure a routing session using standard IP Version 6 (IPv6) address prefixes:

```
address-family ipv6 [unicast]
```

7. Add an entry to the BGP neighbor table:

```
ciscoasa(config-router-af)# neighbor 2001:DB8:0:CC00::1 remote-as 64600
```

8. Apply a route map to incoming or outgoing routes:

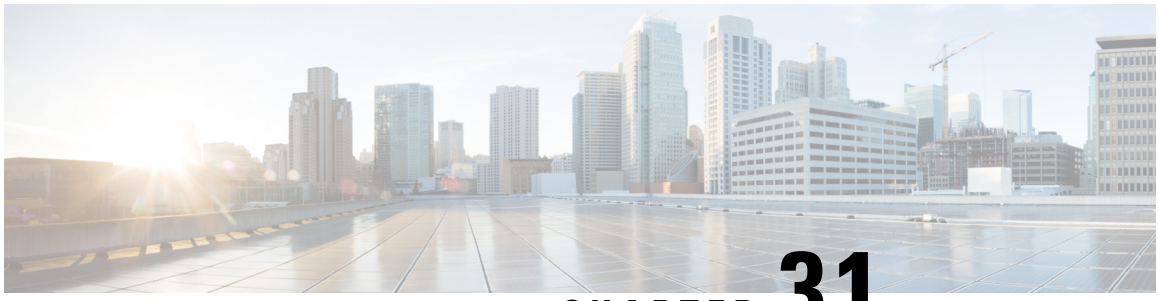
```
ciscoasa(config-router-af)# neighbor 2001:DB8:0:CC00::1 route-map mymap1 in
```

History for BGP

Table 37: Feature History for BGP

Feature Name	Platform Releases	Feature Information
BGP Support	9.2(1)	Support was added for routing data, performing authentication, and redistributing and monitoring routing information using the Border Gateway Protocol. We introduced the following commands: **router bgp, bgp maxas-limit, bgp log-neighbor-changes, bgp transport path-mtu-discovery, bgp fast-external-fallover, bgp enforce-first-as, bgp asnotation dot, timers bgp, bgp default local-preference, bgp always-compare-med, bgp bestpath compare-routerid, bgp deterministic-med, bgp bestpath med missing-as-worst, policy-list, match as-path, match community, match metric, match tag, as-path access-list, community-list, address-family ipv4, bgp router-id, distance bgp, table-map, bgp suppress-inactive, bgp redistribute-internal, bgp scan-time, bgp nexthop, aggregate-address, neighbor, bgp inject-map, show bgp, show bgp cidr-only, show bgp all community, show bgp all neighbors, show bgp community, show bgp community-list, show bgp filter-list, show bgp injected-paths, show bgp ipv4 unicast, show bgp neighbors, show bgp paths, show bgp pending-prefixes, show bgp prefix-list, show bgp regexp, show bgp replication, show bgp rib-failure, show bgp route-map, show bgp summary, show bgp system-config, show bgp update-group, clear route network, maximum-path, network.** We modified the following commands: **show route**, **show route summary**, **show running-config router**, **clear config router**, **clear route all**, **timers lsa arrival**, **timers pacing**, **timers throttle**, **redistribute bgp**.
BGP support for ASA clustering	9.3(1)	We added support for L2 and L3 clustering. We introduced the following command: bgp router-id clusterpool
BGP support for nonstop forwarding	9.3(1)	We added support for Nonstop Forwarding. We introduced the following commands: bgp graceful-restart, neighbor ha-mode graceful-restart
BGP support for advertised maps	9.3(1)	We added support for BGPv4 advertised map. We introduced the following command: neighbor advertise-map

Feature Name	Platform Releases	Feature Information
BGP support for IPv6	9.3(2)	We added support for IPv6. We introduced the following commands: address-family ipv6, ipv6 prefix-list, ipv6 prefix-list description, ipv6 prefix-list sequence-number, match ipv6 next-hop, match ipv6 route-source, match ipv6- address prefix-list, set ipv6-address prefix -list, set ipv6 next-hop, set ipv6 next-hop peer-address We modified the following command: bgp router-id
IPv6 network advertisement for delegated prefixes	9.6(2)	The ASA now supports the DHCPv6 Prefix Delegation client. The ASA obtains delegated prefix(es) from a DHCPv6 server. The ASA can then use these prefixes to configure other ASA interface addresess so that StateLess Address Auto Configuration (SLAAC) clients can autoconfigure IPv6 addresses on the same network. You can configure the BGP router to advertise these prefixes. We modified the following command: **network**



CHAPTER 31

OSPF

This chapter describes how to configure the ASA to route data, perform authentication, and redistribute routing information using the Open Shortest Path First (OSPF) routing protocol.

- OSPF routing, on page 937
- Guidelines for OSPF, on page 941
- Configure OSPFv2, on page 944
- Configure OSPFv2 Router ID, on page 947
- Configure OSPF Fast Hello Packets, on page 948
- Customize OSPFv2, on page 949
- Configure OSPFv3, on page 964
- Configure Graceful Restart, on page 985
- Example for OSPFv2, on page 990
- Examples for OSPFv3, on page 991
- Monitoring OSPF, on page 993
- History for OSPF, on page 996

OSPF routing

OSPF is an interior gateway routing protocol that

- uses link states rather than distance vectors for path selection,
- propagates link-state advertisements rather than routing table updates, and
- converges more quickly than RIP networks because only LSAs are exchanged instead of the entire routing tables.

OSPF characteristics and features

OSPF uses a link-state algorithm to build and calculate the shortest path to all known destinations. Each router in an OSPF area contains an identical link-state database, which is a list of each of the router usable interfaces and reachable neighbors.

The advantages of OSPF over RIP include these characteristics:

- OSPF link-state database updates are sent less frequently than RIP updates, and the link-state database is updated instantly, rather than gradually, AS stale information is timed out.

- Routing decisions are based on cost, which is an indication of the overhead required to send packets across a certain interface. The ASA calculates the cost of an interface based on link bandwidth rather than the number of hops to the destination. The cost can be configured to specify preferred paths.

The disadvantage of shortest path first algorithms is that they require a lot of CPU cycles and memory.

The ASA can run two processes of OSPF protocol simultaneously on different sets of interfaces. You might want to run two processes if you have interfaces that use the same IP addresses (NAT allows these interfaces to coexist, but OSPF does not allow overlapping addresses). Or you might want to run one process on the inside and another on the outside, and redistribute a subset of routes between the two processes. Similarly, you might need to segregate private addresses from public addresses.

You can redistribute routes into an OSPF routing process from another OSPF routing process, a RIP routing process, or from static and connected routes configured on OSPF-enabled interfaces.

The ASA supports these OSPF features:

- Intra-area, inter-area, and external (Type I and Type II) routes.
- Virtual links.
- LSA flooding.
- Authentication to OSPF packets (both password and MD5 authentication).
- Configuring the ASA AS a designated router or a designated backup router. The ASA also can be set up AS an ABR.
- Stub areas and not-so-stubby areas.
- Area boundary router Type 3 LSA filtering.

OSPF supports both MD5 and clear text neighbor authentication. Authentication should be used with all routing protocols when possible because route redistribution between OSPF and other protocols (such AS RIP) can potentially be used by attackers to subvert routing information.

If NAT is used, if OSPF is operating on public and private areas, and if address filtering is required, then you need to run two OSPF processes—one process for the public areas and one for the private areas.

A router that has interfaces in multiple areas is called an Area Border Router (ABR). A router that acts AS a gateway to redistribute traffic between routers using OSPF and routers using other routing protocols is called an Autonomous System Boundary Router (ASBR).

An ABR uses LSAs to send information about available routes to other OSPF routers. Using ABR Type 3 LSA filtering, you can have separate private and public areas with the ASA acting AS an ABR. Type 3 LSAs (inter-area routes) can be filtered from one area to other, which allows you to use NAT and OSPF together without advertising private networks.



Note Only Type 3 LSAs can be filtered. If you configure the ASA AS an ASBR in a private network, it will send Type 5 LSAs describing private networks, which will get flooded to the entire AS, including public areas.

If NAT is employed but OSPF is only running in public areas, then routes to public networks can be redistributed inside the private network, either AS default or Type 5 AS external LSAs. However, you need to configure static routes for the private networks protected by the ASA. Also, you should not mix public and private networks on the same ASA interface.

You can have two OSPF routing processes, one RIP routing process, and one EIGRP routing process running on the ASA at the same time.

OSPF support for fast hello packets

OSPF support for fast hello packets is a feature that

- provides a way to configure the sending of hello packets in intervals less than one second, and
- results in faster convergence in an Open Shortest Path First (OSPF) network.

OSPF fast hello packets configuration prerequisites

OSPF must be configured in the network already or configured at the same time as the OSPF Support for Fast Hello Packets feature.

About OSPF Support for Fast Hello Packets

The key concepts related to OSPF support for fast hello packets and the benefits of OSPF Fast Hello Packets are described below:

OSPF hello interval and dead interval

OSPF hello interval and dead interval are configurable time parameters that

- maintain connectivity between OSPF neighbors through regular hello packet exchanges,
- detect neighbor failures when hello packets are not received within the specified timeframes, and
- must be consistent across all routers within the same network to ensure proper operation.

Hello interval and dead interval configuration details

OSPF hello packets are packets that an OSPF process sends to its OSPF neighbors to maintain connectivity with those neighbors. The hello packets are sent at a configurable interval (in seconds). The defaults are 10 seconds for an Ethernet link and 30 seconds for a non broadcast link.

Hello packets include a list of all neighbors for which a hello packet has been received within the dead interval. The dead interval is also a configurable interval (in seconds), and defaults to four times the value of the hello interval. The value of all hello intervals must be the same within a network. Likewise, the value of all dead intervals must be the same within a network.

These two intervals work together to maintain connectivity by indicating that the link is operational. If a router does not receive a hello packet from a neighbor within the dead interval, it will declare that neighbor to be down.

OSPF fast hello packets

OSPF fast hello packets are hello packets that

- are sent at intervals of less than 1 second,
- are achieved by using the OSPF dead-interval command with a dead interval set to 1 second, and
- use a hello-multiplier value set to the number of hello packets sent during that 1 second.

Fast hello packet behavior

When fast hello packets are configured on the interface, these behaviors occur:

- The hello interval advertised in the hello packets that are sent out this interface is set to 0.
- The hello interval in the hello packets received over this interface is ignored.

The dead interval must be consistent on a segment, whether it is set to 1 second (for fast hello packets) or set to any other value. The hello multiplier need not be the same for the entire segment as long as at least one hello packet is sent within the dead interval.

To understand fast hello packets, you should already understand the relationship between OSPF hello packets and the dead interval. See OSPF hello interval and dead interval, on page 939.

Benefits of OSPF fast hello packets

A fast hello packet is an OSPF protocol feature that

- reduces network convergence time by enabling routers to detect changes in the network faster,
- allows rapid detection of lost neighbors within one second, and
- enhances reliability in LAN environments where neighbor loss may not be detected at the Open Systems Interconnection (OSI) physical or data-link layer.

The benefit of OSPF Fast Hello Packets is that your OSPF network can react swiftly to changes, minimizing downtime and improving overall network stability. This feature is particularly useful in scenarios where routers may not detect lost neighbors via traditional methods, allowing OSPF to maintain accurate routing tables.

If a router on a LAN segment is powered off or disconnected and the physical and data-link layers do not signal neighbor loss, OSPF fast hello packets will detect the neighbor's disappearance within one second.

OSPFv2 and OSPFv3 implementation differences

A protocol distinction is a protocol category that

- prevents backward compatibility between OSPFv2 and OSPFv3,
- requires both OSPF versions to run simultaneously for IPv4 and IPv6 routing, and
- allows OSPFv2 and OSPFv3 to coexist without interacting with each other.

OSPFv3 additional features

OSPFv3 provides additional features that include:

- Protocol processing per link.
- Removal of addressing semantics.
- Addition of flooding scope.
- Support for multiple instances per link.
- Use of the IPv6 link-local address for neighbor discovery and other features.

- LSAs expressed as prefix and prefix length.
- Addition of two LSA types.
- Handling of unknown LSA types.
- Authentication support using the IPsec ESP standard for OSPFv3 routing protocol traffic, as specified by RFC-4552.

Guidelines for OSPF

Context mode guidelines

OSPFv2 supports single and multiple context mode.

- OSPFv2 instances cannot form adjacencies with each other across shared interfaces because, by default, inter-context exchange of multicast traffic is not supported across shared interfaces. However, you can use the static neighbor configuration under OSPFv2 process configuration under OSPFv2 process to bring up OSPFv2 neighbourship on a shared interface.
- Inter-context OSPFv2 on separate interfaces is supported.
- **(For point-to-point topology)** In a multiple context mode, you can configure static OSPFv2 neighbors to advertise OSPFv2 routes over a point-to-point, non-broadcast network. For a successful OSPF adjacency formation, when you remove and reconfigure OSPF on a shared interface, execute the commands in this order: Remove the OSPF configuration using **no router <OSPF name>**, and then remove the OSPF point-to-point non-broadcast configuration on the interface using **no OSPF network point-to-point non-broadcast**. When you reconfigure OSPF on the shared interface, configure the OSPF router using **router <OSPF name>**, and then configure the interface with **OSPF network point-to-point non-broadcast**.
- OSPFv3 supports single mode only.

Key chain authentication guidelines

OSPFv2 supports key chain authentication on both single and multiple mode, both in physical and virtual modes. However, in multiple mode, you can configure the key chain only in context mode.

- The rotating keys are applicable only for OSPFv2 protocol. OSPF area authentication with key chain is not supported.
- The existing MD5 authentication without time range in OSPFv2 is still supported along with new rotating keys.
- Though the platform supports SHA1 and MD5 cryptographic algorithms, only MD5 cryptographic algorithm is used for authentication.

Firewall mode guidelines

OSPF supports routed firewall mode only. OSPF does not support transparent firewall mode.

Failover Guidelines

OSPFv2 and OSPFv3 support Stateful Failover.

IPv6 guidelines

OSPFv2 does not support IPv6, while OSPFv3 provides IPv6 support with specific authentication and routing capabilities.

- OSPFv2 does not support IPv6.
- OSPFv3 supports IPv6.
- OSPFv3 uses IPv6 for authentication.
- The ASA installs OSPFv3 routes into the IPv6 RIB, provided it is the best route.
- OSPFv3 packets can be filtered out using IPv6 ACLs in the **capture** command.

OSPFv3 hello packets and GRE

Typically, OSPF traffic does not pass through GRE tunnel. When OSPFv3 on IPv6 is encapsulated inside GRE, the IPv6 header validation for security check such as Multicast Destination fails. The packet is dropped due to the implicit security check validation, as this packet has destination IPv6 multicast.

- You may define a pre-filter rule to bypass GRE traffic. However, with pre-filter rule, inner packets would not be interrogated by the inspection engine.

Clustering guidelines

OSPFv2 and OSPFv3 have specific limitations and behaviors in clustering environments.

- OSPFv3 encryption is not supported. An error message appears if you try to configure OSPFv3 encryption in a clustering environment.
- In Spanned interface mode, dynamic routing is not supported on management-only interfaces.
- In Individual interface mode, make sure that you establish the control and data units as either OSPFv2 or OSPFv3 neighbors.
- In Individual interface mode, OSPFv2 adjacencies can only be established between two contexts on a shared interface on the control unit. Configuring static neighbors is supported only on point-to point-links; therefore, only one neighbor statement is allowed on an interface.
- When a control role change occurs in the cluster, specific behavior occurs based on interface mode. In spanned interface mode, the router process is active only on the control unit and is in a suspended state on the data units. Each cluster unit has the same router ID because the configuration has been synchronized from the control unit. As a result, a neighboring router does not notice any change in the router ID of the cluster during a role change. In individual interface mode, the router process is active on all the individual cluster units. Each cluster unit chooses its own distinct router ID from the configured cluster pool. A control role change in the cluster does not change the routing topology in any way.

Multiprotocol label switching (MPLS) and OSPF guidelines

When a MPLS-configured router sends Link State (LS) update packets containing opaque Type-10 link-state advertisements (LSAs) that include an MPLS header, authentication fails and the appliance silently drops the update packets, rather than acknowledging them. Eventually the peer router will terminate the neighbor relationship because it has not received any acknowledgments.

- Disable the opaque capability on the ASA to ensure that the neighbor relationship remains stable:

```
router ospf process_ID_number
no nsf ietf helper
no capability opaque
```

These guidelines apply to MPLS environments where opaque LSAs might cause authentication failures.



Note The Firepower 4100/9300 models may have high latency when using MPLS because they lack load balancing across multiple receiving queues.

Route redistribution guidelines

Route redistribution with prefix lists and between routing domains requires specific configuration considerations.

- Redistribution of route maps with IPv4 prefix list on OSPFv2 is supported. However, redistribution of route maps with IPv6 prefix list on OSPFv3 is not supported. Use an access list in the route map on OSPF for redistribution.
- When OSPF is configured on a device that is a part of EIGRP network or vice versa, ensure that OSPF-router is configured to tag the route (EIGRP does not support route tag yet). When redistributing OSPF into EIGRP and EIGRP into OSPF, a routing loop occurs when there is an outage on one of the links, interfaces, or even when the route originator is down. To prevent the redistribution of routes from one domain back into the same domain, a router can tag a route that belongs to a domain while it is redistributing, and those routes can be filtered on the remote router based on the same tag. Because the routes will not be installed into the routing table, they will not be redistributed back into the same domain.

Additional guidelines

OSPFv2 and OSPFv3 support various features and have specific limitations in different deployment scenarios.

- OSPFv2 and OSPFv3 support multiple instances on an interface.
- OSPFv3 supports encryption through ESP headers in a non-clustered environment.
- OSPFv3 supports Non-Payload Encryption.
- OSPFv2 supports Cisco NSF Graceful Restart and IETF NSF Graceful Restart mechanisms as defined in RFCs 4811, 4812 & 3623 respectively.
- OSPFv3 supports Graceful Restart mechanism as defined in RFC 5187.
- There is a limit to the number of intra area (type 1) routes that can be distributed. For these routes, a single type-1 LSA contains all prefixes. Because the system has a limit of 35 KB for packet size, 3000 routes result in a packet that exceeds the limit. Consider 2900 type 1 routes to be the maximum number supported.
- To avoid adjacency flaps due to route updates being dropped if the route update is larger than the minimum MTU on the link, ensure that you configure the same MTU on the interfaces on both sides of the link.
- The ASA virtual cannot use dynamic interior routing protocols like EIGRP and OSPF due to the nature of Azure cloud routing. The Effective Routing Table determines the next hop, regardless of whether a virtual client has any static/dynamic route configured. Currently, you cannot view either the Effective Routing Table or the System Routing Table.

- OSPFv3 drops the LS update if the packet size exceeds 8190. As a result, the neighbourship is terminated. Hence, configure the switch with the "ospfv3 MTU-ignore" command to avoid neighbourship termination.

Configure OSPFv2

This section describes how to enable an OSPFv2 process on the ASA.

After you enable OSPFv2, you need to define a route map. For more information, see Define a Route Map, on page 881. Then you generate a default route. For more information, see Configure a Static Route, on page 855.

After you have defined a route map for the OSPFv2 process, you can customize it for your particular needs, To learn how to customize the OSPFv2 process on the ASA, see Customize OSPFv2, on page 949.

To enable OSPFv2, you need to create an OSPFv2 routing process, specify the range of IP addresses associated with the routing process, then assign area IDs associated with that range of IP addresses.

You can enable up to two OSPFv2 process instances. Each OSPFv2 process has its own associated areas and networks.

To enable OSPFv2, perform the following steps:

Procedure

Step 1 Create an OSPF routing process:

router ospf *process_id*

Example:

```
ciscoasa(config)# router ospf 2
```

The *process_id* argument is an internally used identifier for this routing process and can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.

If there is only one OSPF process enabled on the ASA, then that process is selected by default. You cannot change the OSPF process ID when editing an existing area.

Step 2 Define the IP addresses on which OSPF runs and the area ID for that interface:

network *ip_address mask* **area** *area_id*

Example:

```
ciscoasa(config)# router ospf 2
ciscoasa(config-rtr)# network 10.0.0.0 255.0.0.0 area 0
```

When adding a new area, enter the area ID. You can specify the area ID as either a decimal number or an IP address. Valid decimal values range from 0 to 4294967295. You cannot change the area ID when editing an existing area.

Configure a Key Chain for Authentication

To enhance data security and protection of devices, you can enable rotating keys for authenticating IGP peers. The rotating keys prevent any malicious user from guessing the keys used for routing protocol authentication and thereby protecting the network from advertising incorrect routes and redirecting traffic. Changing the keys frequently reduces the risk of them eventually being guessed. When configuring authentication for routing protocols that provide key chains, configure the keys in a key chain to have overlapping lifetimes. This helps to prevent loss of key-secured communication due to absence of an active key. If the key lifetime expires and no active keys are found, OSPF uses the last valid key to maintain the adjacency with the peers.

This section describes how to create a key chain for OSPF peer authentication. After configuring a key chain object, you can use it in defining the OSPFv2 authentication for an interface and for a virtual link. Use the same authentication type (MD5 or Key Chain) and key ID for the peers to establish a successful adjacency. To learn how to define authentication for an interface, see Configure OSPFv2 Interface Parameters, on page 953.

To configure a key chain, perform the following steps:

Procedure

Step 1 Configure a key chain with a name:

key chain*key-chain-name*

Example:

```
ciscoasa(config)# key chain CHAIN1
ciscoasa(config-keychain)#
```

You can now proceed to define the associated parameters for the key chain.

Step 2 Configure the identifier for the key chain:

key*key-id*

The key id value can be between 0 and 255. Use the value 0 only when you want to signal an invalid key.

Example:

```
ciscoasa(config-keychain)# key 1
ciscoasa(config-keychain-key)#
```

Step 3 Configure the key or password for the key chain:

key-string [**0** | **8**] *key-string-text*

- Use **0** to indicate an unencrypted password follows as shown in the example.
- Use **8** to indicate an encrypted password to follow.

- Even though the **key-string** can accommodate 80 characters, only a maximum of 64 characters is permitted for OSPF authentication.
- The passwords cannot be a single digit nor those starting with a digit followed by a white space. For example, "0 pass" or "1" are invalid.

Example:

```
ciscoasa(config-keychain-key)# key-string 0 CHAIN1KEY1STRING
ciscoasa(config-keychain-key)#
```

Step 4 Configure the cryptographic algorithm for the key chain:

cryptographic-algorithm*md5*

You need to provide the cryptographic authentication algorithm. Though the platform supports SHA1 and MD5, only MD5 is supported for key chain management.

Example:

```
ciscoasa(config-keychain-key)# cryptographic-algorithm md5
ciscoasa(config-keychain-key)#
```

Step 5 (Optional) Configure the lifetime settings for the key chain:

accept-lifetime [**local** | *start-time*] [**duration** *duration value* | **infinite** | *end-time*]

send-lifetime [**ocal** | *start-time*] [**duration** *duration value* | **infinite** | *end-time*]

You can specify the time interval for the device to accept/send the key during key exchange with another device. The end time can be the duration or the absolute time when the accept/send lifetime ends or infinite.

Following are the validation rules for the start and end values:

- Start lifetime cannot be null when the end lifetime is specified.
- The start lifetime for accept or send lifetime must be earlier than the end lifetime.

Example:

```
ciscoasa(config-keychain-key)# accept-lifetime 11:22:33 1 SEP 2018 infinite
ciscoasa(config-keychain-key)#
```

You can use the **show key chain** command to view the start-up key chain configuration on the device; **show run key chain** command to view the key chain configuration that is currently running on the device.

```
ciscoasa# show key chain
Key-chain CHAIN2:
    key 1 -- text "KEY1CHAIN2"
        accept lifetime (always valid) - (always valid) [valid now]
        send lifetime (always valid) - (always valid) [valid now]
  * key 2 -- text "(unset)"
        accept lifetime (11:00:12 UTC Sep 1 2018) - (11:12:12 UTC Sep 1 2018)
        send lifetime (always valid) - (always valid) [valid now]
Key-chain CHAIN1:
```

```
    key 1 -- text "CHAIN1KEY1STRING"
        accept lifetime (11:22:33 UTC Sep 1 2018) - (-1 seconds)
        send lifetime (always valid) - (always valid) [valid now]
ciscoasa#


ciscoasa# sh run key chain
key chain CHAIN2
 key 1
  key-string KEY1CHAIN2
  cryptographic-algorithm md5
 key 2
  accept-lifetime 11:00:12 Sep 1 2018 11:12:12 Sep 1 2018
  cryptographic-algorithm md5
key chain CHAIN1
 key 1
  key-string CHAIN1KEY1STRING
  accept-lifetime 11:22:33 Sep 1 2018 duration -1
  cryptographic-algorithm md5
ciscoasa# sh run key chain CHAIN1
key chain CHAIN1
 key 1
  key-string CHAIN1KEY1STRING
  accept-lifetime 11:22:33 Sep 1 2018 duration -1
  cryptographic-algorithm md5
ciscoasa#
```

What to do next

You can now apply the configured key chain to define the OSPFv2 authentication for an interface.

- Configure OSPFv2 Interface Parameters, on page 953

Configure OSPFv2 Router ID

The OSPF Router-ID is used to identify a specific device within an OSPF database. No two routers in an OSPF system can have the same router-id.

If a router-id is not configured manually in the OSPF routing process the router will automatically configure a router-id determined from the highest IP address of an active interface. When configuring a router-id, the neighbors will not be updated automatically until that router has failed or the OSPF process has been cleared and the neighbor relationship has been re-established.

Manually Configure OSPF Router-ID

This section describes how to manually configure router-id in OSPFv2 process on the ASA.

Procedure

Step 1 To use a fixed router ID, use the **router-id** command.

router-id *ip-address*

Example:

```
ciscoasa(config-router)# router-id 193.168.3.3
```

Step 2 To revert to the previous OSPF router ID behavior, use the **no router-id** command.

no router-id *ip-address*

Example:

```
ciscoasa(config-router)# no router-id 193.168.3.3
```

Router ID Behaviour while Migrating

While migrating OSPF configuration from one ASA , say ASA 1 to another ASA, say ASA 2, the following router id selection behaviour is observed:

1. ASA 2 does not use any IP address for OSPF router-id when all interfaces are in shutdown mode. The possibilities for configuring router-id when all interfaces are in "admin down" state or shutdown mode are:
   - If ASA 2 does not have any router-id configured before, you would see this message:

     ```
     %OSPF: Router process 1 is not running, please configure a router-id
     ```

     After the first interface is brought up, ASA 2 will take IP address of this interface as router id.
   - If ASA 2 had router-id configured before and all interfaces were in "admin down" state when "no router-id" command was issued, ASA 2 will use old router id. ASA 2 uses the old router id, even if IP addresses on the interface that is brought up is changed, until "clear ospf process" command is issued.
2. ASA 2 uses new router id, when ASA 2 had router-id configured before and at least one of interfaces were not in "admin down" state or shutdown mode when "no router-id" command was issued. ASA 2 will use new router id from the IP address of the interfaces even when interfaces are in "down/down" state.

Configure OSPF Fast Hello Packets

This section describes how to configure OSPF Fast Hello Packets.

Procedure

Step 1 Configure an interface:

interface *port-channel number*

Example:

```
ciscoasa(config)# interface port-channel 10
```

The *number* argument indicates the port-channel interface number.

Step 2 Set the interval during which at least one hello packet must be received, or else the neighbor is considered down:

ospf dead-interval minimal hello-multiplier *no.of times*

Example:

```
ciscoasa(config-if)# ospf dead-interval minimal hell0-multiplier 5
ciscoasa
```

The no. of times argument indicates the number of hello packets to be sent every second. Valid values are between 3 and 20.

In this example, OSPF Support for Fast Hello Packets is enabled by specifying the minimal keyword and the hello-multiplier keyword and value. Because the multiplier is set to 5, five hello packets will be sent every second.

Customize OSPFv2

This section explains how to customize the OSPFv2 processes.

Redistribute Routes Into OSPFv2

The ASA can control the redistribution of routes between OSPFv2 routing processes.

Note If you want to redistribute a route by defining which of the routes from the specified routing protocol are allowed to be redistributed into the target routing process, you must first generate a default route. See Configure a Static Route, on page 855, and then define a route map according to Define a Route Map, on page 881.

To redistribute static, connected, RIP, or OSPFv2 routes into an OSPFv2 process, perform the following steps:

Procedure

Step 1 Create an OSPF routing process:

router ospf *process_id*

Example:

```
ciscoasa(config)# router ospf 2
```

The *process_id* argument is an internally used identifier for this routing process and can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.

Step 2 Redistribute connected routes into the OSPF routing process:

redistribute connected [[**metric** *metric-value*] [**metric-type** {**type-1** | **type-2**}] [**tag** *tag_value*] [**subnets**] [**route-map** *map_name*]

Example:

```
ciscoasa(config)# redistribute connected 5 type-1 route-map-practice
```

Step 3 Redistribute static routes into the OSPF routing process:

redistribute static [**subnets**] [**route-map** *map_name*]

Example:

```
ciscoasa(config)# redistribute static subnets
```

This command will pass all the static routes to OSPF. To redistribute selective static routes, ensure to create an access-list with the static route and then include it in a route-map:

Example:

```
ciscoasa(config)# ip access-list extended R1_Loopback
ciscoasa(config-ext-nacl)#permit ip host 1.1.1.1 any
ciscoasa(config-ext-nacl)#exit

ciscoasa(config)#route-map Permit_to_Distribute
ciscoasa(config-route-map)#match ip address R1_Loopback
ciscoasa(config-route-map)#exit
```

After creating the route-map, include it in the redistribute command as follows:

Example:

```
ciscoasa(config)#router ospf 2
ciscoasa(config-router)#redistribute static subnets route-map Permit_to_Distribute
```

Step 4 Redistribute routes from an OSPF routing process into another OSPF routing process:

redistribute ospf pid [**match {internal** | **external** [**1** | **2]** | **nssa-external** [**1** | **2**]}] [**metric** *metric-value*] [**metric-type** {**type-1** | **type-2**}] [**tag** *tag_value*] [**subnets**] [**route-map** *map_name*]

Example:

```
ciscoasa(config)# route-map 1-to-2 permit
ciscoasa(config-route-map)# match metric 1
ciscoasa(config-route-map)# set metric 5
ciscoasa(config-route-map)# set metric-type type-1
ciscoasa(config-route-map)# router ospf 2
ciscoasa(config-rtr)# redistribute ospf 1 route-map 1-to-2
```

You can either use the **match** options in this command to match and set route properties, or you can use a route map. The **subnets** option does not have equivalents in the **route-map** command. If you use both a route map and **match** options in the **redistribute** command, then they must match.

The example shows route redistribution from OSPF process 1 into OSPF process 2 by matching routes with a metric equal to 1. The ASA redistributes these routes as external LSAs with a metric of 5 and a metric type of Type 1.

Step 5 Redistribute routes from a RIP routing process into the OSPF routing process:

redistribute rip [**metric** *metric-value*] [**metric-type** {**type-1** | **type-2**}] [**tag** *tag_value*] [**subnets**] [**route-map** *map_name*]

Example:

```
ciscoasa(config)# redistribute rip 5
ciscoasa(config-route-map)# match metric 1
ciscoasa(config-route-map)# set metric 5
ciscoasa(config-route-map)# set metric-type type-1
ciscoasa(config-rtr)# redistribute ospf 1 route-map 1-to-2
```

Step 6 Redistribute routes from an EIGRP routing process into the OSPF routing process:

redistribute eigrp as-num [**metric** *metric-value*] [**metric-type** {**type-1** | **type-2**}] [**tag** *tag_value*] [**subnets**] [**route-map** *map_name*]

Example:

```
ciscoasa(config)# redistribute eigrp 2
ciscoasa(config-route-map)# match metric 1
ciscoasa(config-route-map)# set metric 5
ciscoasa(config-route-map)# set metric-type type-1
ciscoasa(config-rtr)# redistribute ospf 1 route-map 1-to-2
```

Configure Route Summarization When Redistributing Routes Into OSPFv2

When routes from other protocols are redistributed into OSPF, each route is advertised individually in an external LSA. However, you can configure the ASA to advertise a single route for all the redistributed routes that are included for a specified network address and mask. This configuration decreases the size of the OSPF link-state database.

Routes that match the specified IP address mask pair can be suppressed. The tag value can be used as a match value for controlling redistribution through route maps.

Add a Route Summary Address

To configure the software advertisement on one summary route for all redistributed routes included for a network address and mask, perform the following steps:

Procedure

Step 1 Create an OSPF routing process:

router ospf *process_id*

Example:

```
ciscoasa(config)# router ospf 1
```

The *process_id* argument is an internally used identifier for this routing process and can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.

Step 2 Set the summary address:

summary-address *ip_address mask* [**not-advertise**] [tag *tag*]

Example:

```
ciscoasa(config)# router ospf 1
ciscoasa(config-rtr)# summary-address 10.1.0.0 255.255.0.0
```

In this example, the summary address 10.1.0.0 includes addresses 10.1.1.0, 10.1.2.0, 10.1.3.0, and so on. Only the 10.1.0.0 address is advertised in an external link-state advertisement.

Configure Route Summarization Between OSPFv2 Areas

Route summarization is the consolidation of advertised addresses. This feature causes a single summary route to be advertised to other areas by an area boundary router. In OSPF, an area boundary router advertises networks in one area into another area. If the network numbers in an area are assigned in a way so that they are contiguous, you can configure the area boundary router to advertise a summary route that includes all the individual networks within the area that fall into the specified range.

To define an address range for route summarization, perform the following steps:

Procedure

Step 1 Create an OSPF routing process and enters router configuration mode for this OSPF process:

router ospf *process_id*

Example:

```
ciscoasa(config)# router ospf 1
```

The *process_id* argument is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.

Step 2 Set the address range:

area *area-id* range *ip-address mask* [**advertise** | **not-advertise**]

Example:

```
ciscoasa(config-rtr)# area 17 range 12.1.0.0 255.255.0.0
```

In this example, the address range is set between OSPF areas.

Configure OSPFv2 Interface Parameters

You can change some interface-specific OSPFv2 parameters, if necessary. You are not required to change any of these parameters, but the following interface parameters must be consistent across all routers in an attached network: **ospf hello-interval**, **ospf dead-interval**, and **ospf authentication-key**. If you configure any of these parameters, be sure that the configurations for all routers on your network have compatible values.

To configure OSPFv2 interface parameters, perform the following steps:

Procedure

Step 1 Create an OSPF routing process:

router ospf*process-id*

Example:

```
ciscoasa(config)# router ospf 2
```

The *process-id* argument is an internally used identifier for this routing process and can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.

Step 2 Define the IP addresses on which OSPF runs and the area ID for that interface:

network*ip-address mask***area***area-id*

Example:

```
ciscoasa(config)# router ospf 2
ciscoasa(config-rtr)# network 10.0.0.0 255.0.0.0 area 0
```

Step 3 Enter interface configuration mode:

interface*interface-name*

Example:

```
ciscoasa(config)# interface my_interface
```

Step 4 Specify the authentication type for an interface:

ospf authentication [**key-chain** *key-chain-name* | **message-digest** | **null**]

Provide the key chain name configured. For information on configuring key chain, see Configure a Key Chain for Authentication, on page 945

Example:

```
ciscoasa(config-interface)# ospf authentication message-digest
```

Step 5 Assign a password to be used by neighboring OSPF routers on a network segment that is using the OSPF simple password authentication:

ospf authentication-key*key*

Example:

```
ciscoasa(config-interface)# ospf authentication-key cisco
```

The *key* argument can be any continuous string of characters up to 8 bytes in length.

The password created by this command is used as a key that is inserted directly into the OSPF header when the ASA software originates routing protocol packets. A separate password can be assigned to each network on a per-interface basis. All neighboring routers on the same network must have the same password to be able to exchange OSPF information.

Step 6 Explicitly specify the cost of sending a packet on an OSPF interface:

ospf cost*cost*

Example:

```
ciscoasa(config-interface)# ospf cost 20
```

The *cost* is an integer from 1 to 65535.

In this example, the cost is set to 20.

Step 7 Set the number of seconds that a device must wait before it declares a neighbor OSPF router down because it has not received a hello packet:

ospf dead-interval*seconds*

Example:

```
ciscoasa(config-interface)# ospf dead-interval 40
```

The value must be the same for all nodes on the network.

Step 8 Specify the length of time between the hello packets that the ASA sends on an OSPF interface:

ospf hello-interval*seconds*

Example:

```
ciscoasa(config-interface)# ospf hello-interval 10
```

The value must be the same for all nodes on the network.

Step 9 Enable OSPF MD5 authentication:

ospf message-digest-key*key-id***md5***key*

Example:

```
ciscoasa(config-interface)# ospf message-digest-key 1 md5 cisco
```

The following argument values can be set:

key-id—An identifier in the range from 1 to 255.

key—An alphanumeric password of up to 16 bytes.

Usually, one key per interface is used to generate authentication information when sending packets and to authenticate incoming packets. The same key identifier on the neighbor router must have the same key value.

We recommend that you not keep more than one key per interface. Every time you add a new key, you should remove the old key to prevent the local system from continuing to communicate with a hostile system that knows the old key. Removing the old key also reduces overhead during rollover.

Step 10 Set the priority to help determine the OSPF designated router for a network:

ospf priority *number-value*

Example:

```
ciscoasa(config-interface)# ospf priority 20
```

The *number_value* argument ranges from 0 to 255.

In multiple context mode, for shared interfaces, specify 0 to ensure the device does not become the designated router. OSPFv2 instances cannot form adjacencies with each other across shared interfaces.

Step 11 Specify the number of seconds between LSA retransmissions for adjacencies belonging to an OSPF interface:

ospf retransmit-interval*number-value*

Example:

```
ciscoasa(config-interface)# ospf retransmit-interval seconds
```

The value for *seconds* must be greater than the expected round-trip delay between any two routers on the attached network. The range is from 1 to 8192 seconds. The default value is 5 seconds.

Step 12 Set the estimated number of seconds required to send a link-state update packet on an OSPF interface:

ospf transmit-delay*seconds*

Example:

```
ciscoasa(config-interface)# ospf transmit-delay 5
```

The *seconds* value ranges from 1 to 8192 seconds.The default value is 1 second.

Step 13 Set the number of hello packets sent during 1 second:

ospf dead-interval minimal hello-interval multiplier*integer*

Example:

```
ciscoasa(config-if)# ospf dead-interval minimal hello-multiplier 6
```

Valid values are integers between 3 and 20.

Step 14 Specify the interface as a point-to-point, non-broadcast network:

ospf network point-to-point non-broadcast

Example:

```
ciscoasa(config-interface)# ospf network point-to-point non-broadcast
```

When you designate an interface as point-to-point and non-broadcast, you must manually define the OSPF neighbor; dynamic neighbor discovery is not possible. See Define Static OSPFv2 Neighbors, on page 959 for more information. Additionally, you can only define one OSPF neighbor on that interface.

Configure OSPFv2 Area Parameters

You can configure several OSPF area parameters. These area parameters (shown in the following task list) include setting authentication, defining stub areas, and assigning specific costs to the default summary route. Authentication provides password-based protection against unauthorized access to an area.

Stub areas are areas into which information on external routes is not sent. Instead, there is a default external route generated by the ABR into the stub area for destinations outside the autonomous system. To take advantage of the OSPF stub area support, default routing must be used in the stub area. To further reduce the number of LSAs sent into a stub area, you can use the **no-summary** keyword of the **area stub** command on the ABR to prevent it from sending a summary link advertisement (LSA Type 3) into the stub area.

Procedure

Step 1 Create an OSPF routing process:

router ospf *process_id*

Example:

```
ciscoasa(config)# router ospf 2
```

The *process_id* argument is an internally used identifier for this routing process and can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.

Step 2 Enable authentication for an OSPF area:

area *area-id* authentication

Example:

```
ciscoasa(config-rtr)# area 0 authentication
```

Step 3 Enable MD5 authentication for an OSPF area:

area *area-id* authentication message-digest

Example:

```
ciscoasa(config-rtr)# area 0 authentication message-digest
```

Configure OSPFv2 Filter Rules

Use the following procedure to filter routes or networks received or transmitted in OSPF updates.

Procedure

Step 1 Enable an OSPF routing process and enter router configuration mode:

router ospf *process_id*

Example:

```
ciscoasa(config)# router ospf 2
```

Step 2 Filter routes or networks received in incoming or advertised in outgoing OSPF updates:

distribute-list *acl-number* **in** [**interface** *ifname*]

distribute-list *acl-number* **out** [*protocol process-number* | **connected** | **static**]

The argument *acl-number* specifies IP access list number. The access list defines which networks are to be received and which are to be suppressed in routing updates.

To apply the filter to incoming updates, specify **in**. You can optionally specify an interface to limit the filter to updates received on that interface.

To apply the filter to outbound updates, specify **out**. You can optionally specify a protocol (**bgp**, **eigrp**, **ospf**, or **rip**) with a process number (except for RIP) to apply to the distribution list. You can also filter on whether the peers and networks were learned through **connected** or **static** routes.

Example:

```
ciscoasa(config-rtr)# distribute-list ExampleAcl in interface inside
```

Configure an OSPFv2 NSSA

The OSPFv2 implementation of an NSSA is similar to an OSPFv2 stub area. NSSA does not flood Type 5 external LSAs from the core into the area, but it can import autonomous system external routes in a limited way within the area.

NSSA imports Type 7 autonomous system external routes within an NSSA area by redistribution. These Type 7 LSAs are translated into Type 5 LSAs by NSSA ABRs, which are flooded throughout the whole routing domain. Summarization and filtering are supported during the translation.

You can simplify administration if you are an ISP or a network administrator that must connect a central site using OSPFv2 to a remote site that is using a different routing protocol with NSSA.

Before the implementation of NSSA, the connection between the corporate site border router and the remote router could not be run as an OSPFv2 stub area because routes for the remote site could not be redistributed into the stub area, and two routing protocols needed to be maintained. A simple protocol such as RIP was usually run and handled the redistribution. With NSSA, you can extend OSPFv2 to cover the remote connection by defining the area between the corporate router and the remote router as an NSSA.

Before you use this feature, consider these guidelines:

- You can set a Type 7 default route that can be used to reach external destinations. When configured, the router generates a Type 7 default into the NSSA or the NSSA area boundary router.
- Every router within the same area must agree that the area is NSSA; otherwise, the routers cannot communicate with each other.

Procedure

Step 1 Create an OSPF routing processs:

router ospf *process_id*

Example:

```
ciscoasa(config)# router ospf 2
```

The *process_id* argument is an internally used identifier for this routing process. It can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.

Step 2 Define an NSSA area:

area *area-id* nssa [**no-redistribution**] [**default-information-originate**]

Example:

```
ciscoasa(config-rtr)# area 0 nssa
```

Step 3 Set the summary address and helps reduce the size of the routing table:

summary-address *ip_address mask* [**not-advertise**] [tag *tag*]

Example:

```
ciscoasa(config-rtr)# summary-address 10.1.0.0 255.255.0.0
```

Using this command for OSPF causes an OSPF ASBR to advertise one external route as an aggregate for all redistributed routes that are covered by the address.

In this example, the summary address 10.1.0.0 includes addresses 10.1.1.0, 10.1.2.0, 10.1.3.0, and so on. Only the 10.1.0.0 address is advertised in an external link-state advertisement.

Note
OSPF does not support summary-address 0.0.0.0 0.0.0.0.

Configure an IP Address Pool for Clustering (OSPFv2 and OSPFv3)

You can assign a range of IPv4 addresses for the router ID cluster pool if you are using Individual Interface clustering.

To assign a range of IPv4 addresses for the router ID cluster pool in Individual Interface clustering for OSPFv2 and OSPFv3, enter the following command:

Procedure

Specify the router ID cluster pool for Individual Interface clustering:

router-id cluster-pool hostname | A.B.C.D ip_pool

Example:

```
hostname(config)# ip local pool rpool 1.1.1.1-1.1.1.4
hostname(config)# router ospf 1
hostname(config-rtr)# router-id cluster-pool rpool
hostname(config-rtr)# network 17.5.0.0 255.255.0.0 area 1
hostname(config-rtr)# log-adj-changes
```

The **cluster-pool** keyword enables configuration of an IP address pool when Individual Interface clustering is configured. The **hostname** | **A.B.C.D.** keyword specifies the OSPF router ID for this OSPF process. The *ip_pool* argument specifies the name of the IP address pool.

Note
If you are using clustering, then you do not need to specify an IP address pool for the router ID. If you do not configure an IP address pool, then the ASA uses the automatically generated router ID.

Define Static OSPFv2 Neighbors

You need to define static OSPFv2 neighbors to advertise OSPFv2 routes over a point-to-point, non-broadcast network. This feature lets you broadcast OSPFv2 advertisements across an existing VPN connection without having to encapsulate the advertisements in a GRE tunnel.

Before you begin, you must create a static route to the OSPFv2 neighbor. See Configure a Static Route, on page 855 for more information about creating static routes.

Procedure

Step 1 Create an OSPFv2 routing process:

router ospf *process_id*

Example:

```
ciscoasa(config)# router ospf 2
```

The *process_id* argument is an internally used identifier for this routing process and can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.

Step 2 Define the OSPFv2 neighborhood:

neighbor *addr* [**interface** if_name]

Example:

```
ciscoasa(config-rtr)# neighbor 255.255.0.0 [interface my_interface]
```

The *addr* argument is the IP address of the OSPFv2 neighbor. The *if_name* argument is the interface used to communicate with the neighbor. If the OSPF v2neighbor is not on the same network as any of the directly connected interfaces, you must specify the interface.

Configure Route Calculation Timers

You can configure the delay time between when OSPFv2 receives a topology change and when it starts an SPF calculation. You also can configure the hold time between two consecutive SPF calculations.

Procedure

Step 1 Create an OSPFv2 routing process:

router ospf *process_id*

Example:

```
ciscoasa(config)# router ospf 2
```

The *process_id* argument is an internally used identifier for this routing process and can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.

Step 2 Configure the route calculation times:

timers throttle spf *spf-start spf-hold spf-maximum*

Example:

```
ciscoasa(config-router)# timers throttle spf 500 500 600
```

The *spf-start* argument is the delay time (in milliseconds) between when OSPF receives a topology change and when it starts an SPF calculation. It can be an integer from 0 to 600000.

The *spf-hold* argument is the minimum time (in milliseconds) between two consecutive SPF calculations. It can be an integer from 0 to 600000.

The spf-maximum argument is the maximum time (in milliseconds) between two consecutive SPF calculations. It can be integer from 0 to 600000.

Log Neighbors Going Up or Down

By default, a syslog message is generated when an OSPFv2 neighbor goes up or down.

Configure the **log-adj-changes** command if you want to know about OSPFv2 neighbors going up or down without turning on the **debug ospf adjacency** command. The **log-adj-changes** command provides a higher level view of the peer relationship with less output. Configure the **log-adj-changes detail** command if you want to see messages for each state change.

Procedure

Step 1 Create an OSPFv2 routing process:

router ospf *process_id*

Example:

```
ciscoasa(config)# router ospf 2
```

The *process_id* argument is an internally used identifier for this routing process and can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.

Step 2 Configure logging for neighbors going up or down:

log-adj-changes [**detail**]

Configure a Key Chain for Authentication

To enhance data security and protection of devices, you can enable rotating keys for authenticating IGP peers. The rotating keys prevent any malicious user from guessing the keys used for routing protocol authentication and thereby protecting the network from advertising incorrect routes and redirecting traffic. Changing the keys frequently reduces the risk of them eventually being guessed. When configuring authentication for routing protocols that provide key chains, configure the keys in a key chain to have overlapping lifetimes. This helps

to prevent loss of key-secured communication due to absence of an active key. If the key lifetime expires and no active keys are found, OSPF uses the last valid key to maintain the adjacency with the peers.

This section describes how to create a key chain for OSPF peer authentication. After configuring a key chain object, you can use it in defining the OSPFv2 authentication for an interface and for a virtual link. Use the same authentication type (MD5 or Key Chain) and key ID for the peers to establish a successful adjacency. To learn how to define authentication for an interface, see Configure OSPFv2 Interface Parameters, on page 953.

To configure a key chain, perform the following steps:

Procedure

Step 1 Configure a key chain with a name:

key chain*key-chain-name*

Example:

```
ciscoasa(config)# key chain CHAIN1
ciscoasa(config-keychain)#
```

You can now proceed to define the associated parameters for the key chain.

Step 2 Configure the identifier for the key chain:

key*key-id*

The key id value can be between 0 and 255. Use the value 0 only when you want to signal an invalid key.

Example:

```
ciscoasa(config-keychain)# key 1
ciscoasa(config-keychain-key)#
```

Step 3 Configure the key or password for the key chain:

key-string [**0** | **8**] *key-string-text*

- Use **0** to indicate an unencrypted password follows as shown in the example.
- Use **8** to indicate an encrypted password to follow.
- Even though the **key-string** can accommodate 80 characters, only a maximum of 64 characters is permitted for OSPF authentication.
- The passwords cannot be a single digit nor those starting with a digit followed by a white space. For example, "0 pass" or "1" are invalid.

Example:

```
ciscoasa(config-keychain-key)# key-string 0 CHAIN1KEY1STRING
ciscoasa(config-keychain-key)#
```

Step 4 Configure the cryptographic algorithm for the key chain:

cryptographic-algorithm*md5*

You need to provide the cryptographic authentication algorithm. Though the platform supports SHA1 and MD5, only MD5 is supported for key chain management.

Example:

```
ciscoasa(config-keychain-key)# cryptographic-algorithm md5
ciscoasa(config-keychain-key)#
```

Step 5 (Optional) Configure the lifetime settings for the key chain:

accept-lifetime [**local** | *start-time*] [**duration** *duration value* | **infinite** | *end-time*]

send-lifetime [**ocal** | *start-time*] [**duration** *duration value* | **infinite** | *end-time*]

You can specify the time interval for the device to accept/send the key during key exchange with another device. The end time can be the duration or the absolute time when the accept/send lifetime ends or infinite.

Following are the validation rules for the start and end values:

- Start lifetime cannot be null when the end lifetime is specified.
- The start lifetime for accept or send lifetime must be earlier than the end lifetime.

Example:

```
ciscoasa(config-keychain-key)# accept-lifetime 11:22:33 1 SEP 2018 infinite
ciscoasa(config-keychain-key)#
```

You can use the **show key chain** command to view the start-up key chain configuration on the device; **show run key chain** command to view the key chain configuration that is currently running on the device.

```
ciscoasa# show key chain
Key-chain CHAIN2:
    key 1 -- text "KEY1CHAIN2"
        accept lifetime (always valid) - (always valid) [valid now]
        send lifetime (always valid) - (always valid) [valid now]
  * key 2 -- text "(unset)"
        accept lifetime (11:00:12 UTC Sep 1 2018) - (11:12:12 UTC Sep 1 2018)
        send lifetime (always valid) - (always valid) [valid now]
Key-chain CHAIN1:
    key 1 -- text "CHAIN1KEY1STRING"
        accept lifetime (11:22:33 UTC Sep 1 2018) - (-1 seconds)
        send lifetime (always valid) - (always valid) [valid now]
ciscoasa#
```

```
ciscoasa# sh run key chain
key chain CHAIN2
 key 1
  key-string KEY1CHAIN2
  cryptographic-algorithm md5
 key 2
  accept-lifetime 11:00:12 Sep 1 2018 11:12:12 Sep 1 2018
  cryptographic-algorithm md5
key chain CHAIN1
 key 1
  key-string CHAIN1KEY1STRING
```

```
  accept-lifetime 11:22:33 Sep 1 2018 duration -1
  cryptographic-algorithm md5
ciscoasa# sh run key chain CHAIN1
key chain CHAIN1
 key 1
  key-string CHAIN1KEY1STRING
  accept-lifetime 11:22:33 Sep 1 2018 duration -1
  cryptographic-algorithm md5
ciscoasa#
```

What to do next

You can now apply the configured key chain to define the OSPFv2 authentication for an interface.

- Configure OSPFv2 Interface Parameters, on page 953

Configure OSPFv3

This section describes the tasks involved in configuring an OSPFv3 routing process.

Enable OSPFv3

To enable OSPFv3, you need to create an OSPFv3 routing process, create an area for OSPFv3, enable an interface for OSPFv3, then redistribute the route into the targeted OSPFv3 routing processes.

Procedure

Step 1 Create an OSPFv3 routing process:

ipv6 router ospf *process-id*

Example:

```
ciscoasa(config)# ipv6 router ospf 10
```

The *process-id* argument is an internally used tag for this routing process and can be any positive integer. This tag does not have to match the tag on any other device; it is for internal use only. You can use a maximum of two processes.

Step 2 Enable an interface:

interface *interface_name*

Example:

```
ciscoasa(config)# interface Gigabitethernet0/0
```

Step 3 Create the OSPFv3 routing process with the specified process ID and an area for OSPFv3 with the specified area ID:

ipv6 ospf *process-id* **area** *area_id*

Example:

```
ciscoasa(config)# ipv6 ospf 200 area 100
```

Configure OSPFv3 Interface Parameters

You can change certain interface-specific OSPFv3 parameters, if necessary. You are not required to change any of these parameters, but the following interface parameters must be consistent across all routers in an attached network: the hello interval and the dead interval. If you configure any of these parameters, be sure that the configurations for all routers on your network have compatible values.

Procedure

Step 1 Enable an OSPFv3 routing process:

ipv6 router ospf *process-id*

Example:

```
ciscoasa(config-if)# ipv6 router ospf 10
```

The *process-id* argument is an internally used tag for this routing process and can be any positive integer. This tag does not have to match the tag on any other device; it is for internal use only. You can use a maximum of two processes.

Step 2 Create an OSPFv3 area:.

ipv6 ospf **area** [*area-num*] [**instance**]

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
```

The *area-num* argument is the area for which authentication is to be enabled and can be either a decimal value or an IP address. The **instance** keyword specifies the area instance ID that is to be assigned to an interface. An interface can have only one OSPFv3 area. You can use the same area on multiple interfaces, and each interface can use a different area instance ID.

Step 3 Specify the cost of sending a packet on an interface:

ipv6 ospf cost *interface-cost*

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
```

The *interface-cost* argument specifies an unsigned integer value expressed as the link-state metric, which can range in value from 1 to 65535. The default cost is based on the bandwidth.

Step 4 Filter outgoing LSAs to an OSPFv3 interface:

ipv6 ospf database-filter all out

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
ipv6 ospf database-filter all out
```

All outgoing LSAs are flooded to the interface by default.

Step 5 Set the time period in seconds for which hello packets must not be seen before neighbors indicate that the router is down:

ipv6 ospf dead-interval *seconds*

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
ipv6 ospf dead-interval 60
```

The value must be the same for all nodes on the network and can range from 1 to 65535. The default is four times the interval set by the **ipv6 ospf hello-interval** command.

Step 6 Specify the encryption type for an interface:

ipv6 ospf encryption {**ipsec spi** *spi* **esp** *encryption-algorithm* [[*key-encryption-type*] *key*] *authentication-algorithm* [[*key-encryption-type*] *key* | **null**}

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
ipv6 ospf encryption ipsec spi 1001 esp null sha1 123456789A123456789B123456789C123456789D
```

The **ipsec** keyword specifies the IP security protocol. The **spi** *spi* keyword-argument pair specifies the security policy index, which must be in the range of 256 to 42949667295 and entered as a decimal.

The **esp** keyword specifies the encapsulating security payload. The *encryption-algorithm* argument specifies the encryption algorithm to be used with ESP. Valid values include the following:

- aes-cdc—Enables AES-CDC encryption.
- 3des—Enables 3DES encryption.
- des—Enables DES encryption.
- null—Specifies ESP with no encryption.

The *key-encryption-type* argument can be one of the following two values:

- 0—The key is not encrypted.
- 7—The key is encrypted.

The *key* argument specifies the number used in the calculation of the message digest. The number is 32 hexadecimal digits (16 bytes) long. The size of the key depends on the encryption algorithm used. Some algorithms, such as AES-CDC, allow you to choose the size of the key. The *authentication-algorithm* argument specifies the encryption authentication algorithm to be used, which can be one of the following:

- md5—Enables message digest 5 (MD5).
- sha1—Enables SHA-1.

The **null** keyword overrides area encryption.

If OSPFv3 encryption is enabled on an interface and a neighbor is on different area (for example, area 0), and you want the ASA to form adjacencies with that area, you must change the area on the ASA. After you have changed the area on the ASA to 0, there is a delay of two minutes before the OSPFv3 adjacency comes up.

Step 7 Specify the flood reduction of LSAs to the interface:

ipv6 ospf flood-reduction

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
ipv6 ospf flood reduction
```

Step 8 Specify the interval in seconds between hello packets sent on the interface:

ipv6 ospf hello-interval *seconds*

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
ipv6 ospf hello-interval 15
```

The value must be the same for all nodes on a specific network and can range from 1 to 65535. The default interval is 10 seconds for Ethernet interfaces and 30 seconds for non-broadcast interfaces.

Step 9 Disable the OSPF MTU mismatch detection when DBD packets are received:

ipv6 ospf mtu-ignore

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
ipv6 ospf mtu-ignore
```

OSPF MTU mismatch detection is enabled by default.

Step 10 Set the OSPF network type to a type other than the default, which depends on the network type:

ipv6 ospf network {**broadcast** | **point-to-point non-broadcast**}

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
ipv6 ospf network point-to-point non-broadcast
```

The **point-to-point non-broadcast** keyword sets the network type to point-to-point non-broadcast. The **broadcast** keyword sets the network type to broadcast.

Step 11 Set the router priority, which helps determine the designated router for a network:

ipv6 ospf priority *number-value*

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
ipv6 ospf priority 4
```

Valid values range from 0 to 255.

Step 12 Configure OSPFv3 router interconnections to non-broadcast networks:

ipv6 ospf neighbor *ipv6-address* [**priority** *number*] [**poll-interval** *seconds*] [**cost** *number*] [**database-filter all out**]

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
```

```
ipv6 ospf neighbor FE80::A8BB:CCFF:FE00:C01
```

Step 13 Specify the time in seconds between LSA retransmissions for adjacencies that belong to the interface:

ipv6 ospf retransmit-interval *seconds*

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
ipv6 ospf retransmit-interval 8
```

The time must be greater than the expected round-trip delay between any two routers on the attached network. Valid values range from 1 to 65535 seconds. The default is 5 seconds.

Step 14 Set the estimated time in seconds to send a link-state update packet on the interface:

ipv6 ospf transmit-delay *seconds*

Example:

```
ciscoasa(config-if)# interface GigabitEthernet3/2.200
 vlan 200
 nameif outside
 security-level 100
 ip address 10.20.200.30 255.255.255.0 standby 10.20.200.31
 ipv6 address 3001::1/64 standby 3001::8
 ipv6 address 6001::1/64 standby 6001::8
 ipv6 enable
 ospf priority 255
 ipv6 ospf cost 100
 ipv6 ospf 100 area 10 instance 200
ipv6 ospf retransmit-delay 3
```

Valid values range from 1 to 65535 seconds. The default is 1 second.

Configure OSPFv3 Router Parameters

Procedure

Step 1 Enable an OSPFv3 routing process:

ipv6 router ospf *process-id*

Example:

```
ciscoasa(config)# ipv6 router ospf 10
```

The *process-id* argument is an internally used identifier for this routing process, is locally assigned, and can be any positive integer from 1 to 65535. This ID does not have to match the ID on any other device; it is for internal administrative use only. You can use a maximum of two processes.

Step 2 Configure OSPFv3 area parameters:

area

Example:

```
ciscoasa(config-rtr)# area 10
```

Supported parameters include the area ID as a decimal value from 0 to 4294967295 and the area ID in the IP address format of **A.B.C.D**.

Step 3 Set a command to its default value:

default

Example:

```
ciscoasa(config-rtr)# default originate
```

The **originate** parameter distributes the default route.

Step 4 Control distribution of default information:

default-information

Step 5 Define the OSPFv3 route administrative distance based on the route type:

distance

Example:

```
ciscoasa(config-rtr)# distance 200
```

Supported parameters include the administrative distance with values from 1 to 254 and **ospf** for the OSPFv3 distance.

Step 6 Suppress the sending of syslog messages with the **lsa** parameter when the router receives a link-state advertisement (LSA) for Type 6 Multicast OSPF (MOSPF) packets:

ignore

Example:

```
ciscoasa(config-rtr)# ignore lsa
```

Step 7 Configure the router to send a syslog message when an OSPFv3 neighbor goes up or down:

log-adjacency-changes

Example:

```
ciscoasa(config-rtr)# log-adjacency-changes detail
```

With the **detail** parameter, all state changes are logged.

Step 8 Suppress the sending and receiving of routing updates on an interface:

passive-interface [*interface_name*]

Example:

```
ciscoasa(config-rtr)# passive-interface inside
```

The *interface_name* argument specifies the name of the interface on which the OSPFv3 process is running.

Step 9 Configure the redistribution of routes from one routing domain into another:

redistribute {**connected** | **ospf** | **static**}

Where:

- **connected**—Specifies connected routes.
- **ospf**—Specifies OSPFv3 routes.
- **static**—Specifies static routes.

Example:

```
ciscoasa(config-rtr)# redistribute ospf
```

Step 10 Create a fixed router ID for a specified process:

router-id {*A.B.C.D* | **cluster-pool** | **static**}

Where:

A.B.C.D—Specifies the OSPF router ID in IP address format.

cluster-pool—Configures an IP address pool when Individual Interface clustering is configured. For more information about IP address pools used in clustering, see Configure an IP Address Pool for Clustering (OSPFv2 and OSPFv3), on page 959.

Example:

```
ciscoasa(config-rtr)# router-id 10.1.1.1
```

Step 11 Configure IPv6 address summaries with valid values from 0 to 128:

summary-prefix *X:X:X:X::X/*

Example:

```
ciscoasa(config-if)# ipv6 router ospf 1
ciscoasa(config-router)# router-id 192.168.3.3
ciscoasa(config-router)# summary-prefix FECO::/24
```

```
ciscoasa(config-router)# redistribute static
```

The *X:X:X:X::X/* parameter specifies the IPv6 prefix.

Step 12 Adjust routing timers:

timers

The routing timer parameters are the following:

- **lsa**—Specifies OSPFv3 LSA timers.
- **nsf**—Specifies OSPFv3 NSF wait timers.
- **pacing**—Specifies OSPFv3 pacing timers.
- **throttle**—Specifies OSPFv3 throttle timers.

Example:

```
ciscoasa(config)# ipv6 router ospf 10
ciscoasa(config-rtr)# timers throttle spf 6000 12000 14000
```

Configure OSPFv3 Area Parameters

Procedure

Step 1 Enable an OSPFv3 routing process:

ipv6 router ospf *process-id*

Example:

```
ciscoasa(config)# ipv6 router ospf 1
```

The *process-id* argument is an internally used identifier for this routing process, is locally assigned, and can be any positive integer from 1 to 65535.

This ID does not have to match the ID on any other device; it is for internal administrative use only. You can use a maximum of two processes.

Step 2 Set the summary default cost of an NSSA area or a stub area:

area *area-id* **default-cost** *cost*

Example:

```
ciscoasa(config-rtr)# area 1 default-cost nssa
```

Step 3 Summarize routes that match the address and mask for border routers only:

area *area-id* **range** *ipv6-prefix*/ *prefix-length* [**advertise** | **not advertise**] [**cost** *cost*]

Example:

```
ciscoasa(config-rtr)# area 1 range FE01:1::1/64
```

- The *area-id* argument identifies the area for which routes are to be summarized. The value can be specified as a decimal or an IPv6 prefix.
- The *ipv6-prefix* argument specifies the IPv6 prefix. The *prefix-length* argument specifies the prefix length.
- The **advertise** keyword sets the address range status to advertised and generates a Type 3 summary LSA.
- The **not-advertise** keyword sets the address range status to DoNotAdvertise.
- The Type 3 summary LSA is suppressed, and the component networks remain hidden from other networks.
- The **cost** *cost* keyword-argument pair specifies the metric or cost for the summary route, which is used during OSPF SPF calculations to determine the shortest paths to the destination.
- Valid values range from 0 to 16777215.

Step 4 Specify an NSSA area:

area *area-id* **nssa**

Example:

```
ciscoasa(config-rtr)# area 1 nssa
```

Step 5 Specify a stub area:

area *area-id* **stub**

Example:

```
ciscoasa(config-rtr)# area 1 stub
```

Step 6 Define a virtual link and its parameters:

area *area-id* **virtual-link** *router-id* [**hello-interval** *seconds*] [**retransmit-interval** *seconds*] [**transmit-delay** *seconds*] [**dead-interval** *seconds*] [**ttl-security hops** *hop-count*]

Example:

```
ciscoasa(config-rtr)# area 1 virtual-link 192.168.255.1 hello-interval 5
```

- The *area-id* argument identifies the area for which routes are to be summarized. The **virtual link** keyword specifies the creation of a virtual link neighbor.
- The *router-id* argument specifies the router ID that is associated with the virtual link neighbor.
- Enter the **show ospf** or **show ipv6 ospf** command to display the router ID. There is no default value.
- The **hello-interval** keyword specifies the time in seconds between the hello packets that are sent on an interface. The hello interval is an unsigned integer that is to be advertised in the hello packets. The value

must be the same for all routers and access servers that are attached to a common network. Valid values range from 1 to 8192. The default is 10.

- The **retransmit-interval** *seconds* keyword-argument pair specifies the time in seconds between LSA retransmissions for adjacencies that belong to the interface. The retransmit interval is the expected round-trip delay between any two routers on the attached network. The value must be greater than the expected round-trip delay, and can range from 1 to 8192. The default is 5.
- The **transmit-delay** *seconds* keyword-argument pair specifies the estimated time in seconds that is required to send a link-state update packet on the interface. The integer value must be greater than zero. LSAs in the update packet have their own ages incremented by this amount before transmission. The range of values can be from 1 to 8192. The default is 1.
- The **dead-interval** *seconds* keyword-argument pair specifies the time in seconds that hello packets are not seen before a neighbor indicates that the router is down. The dead interval is an unsigned integer. The default is four times the hello interval, or 40 seconds. The value must be the same for all routers and access servers that are attached to a common network. Valid values range from 1 to 8192.
- The **ttl-security hops** keyword configures the time-to-live (TTL) security on a virtual link. The *hop-count* argument value can range from 1 to 254.

Configure OSPFv3 Passive Interfaces

Procedure

Step 1 Enable an OSPFv3 routing process:

ipv6 router ospf *process_id*

Example:

```
ciscoasa(config-if)# ipv6 router ospf 1
```

The *process_id* argument is an internally used identifier for this routing process, is locally assigned, and can be any positive integer from 1 to 65535. This ID does not have to match the ID on any other device; it is for internal administrative use only. You can use a maximum of two processes.

Step 2 Suppress the sending and receiving of routing updates on an interface:

passive-interface [*interface_name*]

Example:

```
ciscoasa(config-rtr)# passive-interface inside
```

The *interface_name* argument specifies the name of the interface on which the OSPFv3 process is running. If the *no interface_name* argument is specified, all of the interfaces in the OSPFv3 process *process_id* are made passive.

Configure OSPFv3 Administrative Distance

Procedure

Step 1 Enable an OSPFv3 routing process:

ipv6 router ospf *process_id*

Example:

```
ciscoasa(config-if)# ipv6 router ospf 1
```

The *process_id* argument is an internally used identifier for this routing process, is locally assigned, and can be any positive integer from 1 to 65535. This ID does not have to match the ID on any other device; it is for internal administrative use only. You can use a maximum of two processes.

Step 2 Set the administrative distance for OSPFv3 routes:

distance [**ospf** {**external** | **inter-area** | **intra-area**}] *distance*

Example:

```
ciscoasa(config-rtr)# distance ospf external 200
```

The **ospf** keyword specifies OSPFv3 routes. The **external** keyword specifies the external Type 5 and Type 7 routes for OSPFv3. The **inter-area** keyword specifies the inter-area routes for OSPVv3. The **intra-area** keyword specifies the intra-area routes for OSPFv3. The *distance* argument specifies the administrative distance, which is an integer from 10 to 254.

Configure OSPFv3 Timers

You can set LSA arrival, LSA pacing, and throttling timers for OSPFv3.

Procedure

Step 1 Enable an OSPFv3 routing process:

ipv6 router ospf *process-id*

Example:

```
ciscoasa(config-if)# ipv6 router ospf 1
```

The *process-id* argument is an internally used identifier for this routing process, is locally assigned, and can be any positive integer from 1 to 65535. This ID does not have to match the ID on any other device; it is for internal administrative use only. You can use a maximum of two processes.

Step 2 Set the minimum interval at which the ASA accepts the same LSA from OSPF neighbors:

timers lsa arrival *milliseconds*

Example:

```
ciscoasa(config-rtr)# timers lsa arrival 2000
```

The *milliseconds* argument specifies the minimum delay in milliseconds that must pass between acceptance of the same LSA arriving from neighbors. The range is from 0 to 6,000,000 milliseconds. The default is 1000 milliseconds.

Step 3 Configure LSA flood packet pacing:

timers pacing flood *milliseconds*

Example:

```
ciscoasa(config-rtr)# timers lsa flood 20
```

The *milliseconds* argument specifies the time in milliseconds at which LSAs in the flooding queue are paced in between updates. The configurable range is from 5 to 100 milliseconds. The default value is 33 milliseconds.

Step 4 Change the interval at which OSPFv3 LSAs are collected into a group and refreshed, checksummed, or aged:

timers pacing lsa-group *seconds*

Example:

```
ciscoasa(config-rtr)# timers pacing lsa-group 300
```

The *seconds* argument specifies the number of seconds in the interval at which LSAs are grouped, refreshed, check summed, or aged. The range is from 10 to 1800 seconds. The default value is 240 seconds.

Step 5 Configure LSA retransmission packet pacing:

timers pacing retransmission *milliseconds*

Example:

```
ciscoasa(config-rtr)# timers pacing retransmission 100
```

The *milliseconds* argument specifies the time in milliseconds at which LSAs in the retransmission queue are paced. The configurable range is from 5 to 200 milliseconds. The default value is 66 milliseconds.

Step 6 Configure OSPFv3 LSA throttling:

timers throttle lsa *milliseconds1 milliseconds2 milliseconds3*

Example:

```
ciscoasa(config-rtr)# timers throttle lsa 500 6000 8000
```

- The *milliseconds1* argument specifies the delay in milliseconds to generate the first occurrence of the LSA. The *milliseconds2* argument specifies the maximum delay in milliseconds to originate the same LSA. The *milliseconds3* argument specifies the minimum delay in milliseconds to originate the same LSA.
- For LSA throttling, if the minimum or maximum time is less than the first occurrence value, then OSPFv3 automatically corrects to the first occurrence value. Similarly, if the maximum delay specified is less than the minimum delay, then OSPFv3 automatically corrects to the minimum delay value.
- For *milliseconds1*, the default value is 0 milliseconds.
- For *milliseconds2* and *milliseconds3*, the default value is 5000 milliseconds.

Step 7 Configure OSPFv3 SPF throttling:

timers throttle spf *milliseconds1 milliseconds2 milliseconds3*

Example:

```
ciscoasa(config-rtr)# timers throttle spf 5000 12000 16000
```

- The *milliseconds1* argument specifies the delay in milliseconds to receive a change to the SPF calculation. The *milliseconds2* argument specifies the delay in milliseconds between the first and second SPF calculations. The *milliseconds3* argument specifies the maximum wait time in milliseconds for SPF calculations.
- For SPF throttling, if *milliseconds2* or *milliseconds3* is less than *milliseconds1*, then OSPFv3 automatically corrects to the *milliseconds1* value. Similarly, if *milliseconds3* is less than *milliseconds2*, then OSPFv3 automatically corrects to the *milliseconds2* value.
- For *milliseconds1*, the default value of SPF throttling is 5000 milliseconds.
- For *milliseconds2* and *milliseconds3*, the default value of SPF throttling is 10000 milliseconds.

Define Static OSPFv3 Neighbors

You need to define static OSPFv3 neighbors to advertise OSPF routes over a point-to-point, non-broadcast network. This feature lets you broadcast OSPFv3 advertisements across an existing VPN connection without having to encapsulate the advertisements in a GRE tunnel.

Before you begin, you must create a static route to the OSPFv3 neighbor. See Configure a Static Route, on page 855 for more information about creating static routes.

Procedure

Step 1 Enable an OSPFv3 routing process and enters IPv6 router configuration mode.

ipv6 router ospf *process-id*

Example:

```
ciscoasa(config)# ipv6 router ospf 1
```

The *process-id* argument is an internally used identifier for this routing process, is locally assigned, and can be any positive integer from 1 to 65535. This ID does not have to match the ID on any other device; it is for internal administrative use only. You can use a maximum of two processes.

Step 2 Configure OSPFv3 router interconnections to non-broadcast networks.

ipv6 ospf neighbor *ipv6-address* [**priority** *number*] [**poll-interval** *seconds*] [**cost** *number*] [**database-filter all out**]

Example:

```
ciscoasa(config-if)# interface ethernet0/0 ipv6 ospf neighbor FE80::A8BB:CCFF:FE00:C01
```

Reset OSPFv3 Default Parameters

To return an OSPFv3 parameter to its default value, perform the following steps:

Procedure

Step 1 Enable an OSPFv3 routing process:

ipv6 router ospf *process-id*

Example:

```
ciscoasa(config-if)# ipv6 router ospf 1
```

The *process_id* argument is an internally used identifier for this routing process, is locally assigned, and can be any positive integer from 1 to 65535. This ID does not have to match the ID on any other device; it is for internal administrative use only. You can use a maximum of two processes.

Step 2 Return an optional parameter to its default value:

default [**area** | **auto-cost** | **default-information** | **default-metric** | **discard-route** | **discard-route | distance** | **distribute-list** | **ignore** | **log-adjacency-changes** | **maximum-paths** | **passive-interface** | **redistribute** | **router-id** | **summary-prefix** | **timers**]

Example:

```
ciscoasa(config-rtr)# default metric 5
```

- The **area** keyword specifies the OSPFv3 area parameters. The **auto-cost** keyword specifies the OSPFv3 interface cost according to bandwidth.

- The **default-information** keyword distributes default information. The **default-metric** keyword specifies the metric for a redistributed route
- The **discard-route** keyword enables or disables the discard-route installation. The **distance** keyword specifies the administrative distance.
- The **distribute-list** keyword filters networks in routing updates.
- The **ignore** keyword ignores a specific event. The **log-adjacency-changes** keyword logs changes in the adjacency state.
- The **maximum-paths** keyword forwards packets over multiple paths.
- The **passive-interface** keyword suppresses routing updates on an interface.
- The **redistribute** keyword redistributes IPv6 prefixes from another routing protocol.
- The **router-id** keyword specifies the router ID for the specified routing process.
- The **summary-prefix** keyword specifies the IPv6 summary prefix.
- The **timers** keyword specifies the OSPFv3 timers.

Send Syslog Messages

Configure the router to send a syslog message when an OSPFv3 neighbor goes up or down.

Procedure

Step 1 Enable an OSPFv3 routing process:

ipv6 router ospf *process-id*

Example:

```
ciscoasa(config-if)# ipv6 router ospf 1
```

The *process-id* argument is an internally used identifier for this routing process, is locally assigned, and can be any positive integer from 1 to 65535. This ID does not have to match the ID on any other device; it is for internal administrative use only. You can use a maximum of two processes.

Step 2 Configure the router to send a syslog message when an OSPFv3 neighbor goes up or down:

log-adjacency-changes [**detail**]

Example:

```
ciscoasa(config-rtr)# log-adjacency-changes detail
```

The **detail** keyword sends a syslog message for each state, not only when an OSPFv3 neighbor goes up or down.

Suppress Syslog Messages

To suppress the sending of syslog messages when the route receives unsupported LSA Type 6 multicast OSPF (MOSPF) packets, perform the following steps:

Procedure

Step 1 Enable an OSPFv2 routing process:

router ospf *process_id*

Example:

```
ciscoasa(config-if)# router ospf 1
```

The *process_id* argument is an internally used identifier for this routing process, is locally assigned, and can be any positive integer from 1 to 65535. This ID does not have to match the ID on any other device; it is for internal administrative use only. You can use a maximum of two processes.

Step 2 Suppress the sending of syslog messages when the router receives unsupported LSA Type 6 MOSPF packets:

ignore lsa mospf

Example:

```
ciscoasa(config-rtr)# ignore lsa mospf
```

Calculate Summary Route Costs

Procedure

Restore the methods that are used to calculate summary route costs according to RFC 1583:

compatible rfc1583

Example:

```
ciscoasa (config-rtr)# compatible rfc1583
```

Generate a Default External Route into an OSPFv3 Routing Domain

Procedure

Step 1 Enable an OSPFv3 routing process:

ipv6 router ospf *process-id*

Example:

```
ciscoasa(config-if)# ipv6 router ospf 1
```

The *process-id* argument is an internally used identifier for this routing process, is locally assigned, and can be any positive integer from 1 to 65535. This ID does not have to match the ID on any other device; it is for internal administrative use only. You can use a maximum of two processes.

Step 2 Generate a default external route into an OSPFv3 routing domain:

default-information originate [**always**] **metric** *metric-value* [**metric-type** *type-value*] [**route-map** *map-name*]

Example:

```
ciscoasa(config-rtr)# default-information originate always metric 3 metric-type 2
```

- The **always** keyword advertises the default route whether or not the default route exists.
- The **metric** *metric-value* keyword-argument pair specifies the metric used for generating the default route.
- If you do not specify a value using the **default-metric** command, the default value is 10. Valid metric values range from 0 to 16777214.
- The **metric-type** *type-value* keyword-argument pair specifies the external link type that is associated with the default route that is advertised into the OSPFv3 routing domain. Valid values can be one of the following:
  - 1—Type 1 external route
  - 2—Type 2 external route

  The default is the type 2 external route.
- The **route-map** *map-name* keyword-argument pair specifies the routing process that generates the default route if the route map is satisfied.

Configure an IPv6 Summary Prefix

Procedure

Step 1 Enable an OSPFv3 routing process:

ipv6 router ospf *process-id*

Example:

```
ciscoasa(config-if)# ipv6 router ospf 1
```

The *process_id* argument is an internally used identifier for this routing process, is locally assigned, and can be any positive integer from 1 to 65535. This ID does not have to match the ID on any other device; it is for internal administrative use only. You can use a maximum of two processes.

Step 2 Configure an IPv6 summary prefix:

summary-prefix *prefix* [**not-advertise | tag** *tag-value*]

Example:

```
ciscoasa(config-if)# ipv6 router ospf 1
ciscoasa(config-rtr)# router-id 192.168.3.3
ciscoasa(config-rtr)# summary-prefix FECO::/24
ciscoasa(config-rtr)# redistribute static
```

The *prefix* argument is the IPv6 route prefix for the destination. The **not-advertise** keyword suppresses routes that match the specified prefix and mask pair. This keyword applies to OSPFv3 only. The **tag** *tag-value* keyword-argument pair specifies the tag value that can be used as a match value for controlling redistribution through route maps. This keyword applies to OSPFv3 only.

Redistribute IPv6 Routes

Procedure

Step 1 Enable an OSPFv3 routing process:

ipv6 router ospf *process-id*

Example:

```
ciscoasa(config-if)# ipv6 router ospf 1
```

The *process-id* argument is an internally used identifier for this routing process, is locally assigned, and can be any positive integer from 1 to 65535. This ID does not have to match the ID on any other device; it is for internal administrative use only. You can use a maximum of two processes.

Step 2 Redistribute IPv6 routes from one OSPFv3 process into another:

redistribute *source-protocol* [*process-id*] [**include-connected** {[**level-1** | **level-2**}] [*as-number*] [**metric** [*metric-value* | **transparent**}] **[metric-type** *type-value*] [**match** {**external** [**1**|**2**] | **internal** | **nssa-external** [**1**|**2**]}] [**tag** *tag-value*] [**route-map** *map-tag*]

Example:

```
ciscoasa(config-rtr)# redistribute connected 5 type-1
```

- The *source-protocol* argument specifies the source protocol from which routes are being redistributed, which can be static, connected, or OSPFv3.
- The *process-id* argument is the number that is assigned administratively when the OSPFv3 routing process is enabled.
- The **include-connected** keyword allows the target protocol to redistribute routes learned by the source protocol and connected prefixes on those interfaces over which the source protocol is running.
- The **level-1** keyword specifies that for Intermediate System-to-Intermediate System (IS-IS), Level 1 routes are redistributed into other IP routing protocols independently.
- The **level-1-2** keyword specifies that for IS-IS, both Level 1 and Level 2 routes are redistributed into other IP routing protocols.
- The **level-2** keyword specifies that for IS-IS, Level 2 routes are redistributed into other IP routing protocols independently.
- For the **metric** *metric-value* keyword-argument pair, when redistributing routes from one OSPFv3 process into another OSPFv3 process on the same router, the metric is carried through from one process to the other if no metric value is specified. When redistributing other processes into an OSPFv3 process, the default metric is 20 when no metric value is specified.
- The **metric transparent** keyword causes RIP to use the routing table metric for redistributed routes as the RIP metric.
- The **metric-type** *type-value* keyword-argument pair specifies the external link type that is associated with the default route that is advertised into the OSPFv3 routing domain. Valid values can be one of the following: 1 for a Type 1 external route or 2 for a Type 2 external route. If no value is specified for the **metric-type** keyword, the ASA adopts a Type 2 external route. For IS-IS, the link type can be one of the following: internal for an IS-IS metric that is less than 63 or external for an IS-IS metric that is greater than 64 and less than 128. The default is internal.
- The **match** keyword redistributes routes into other routing domains and is used with one of the following options: **external** [**1**|**2**] for routes that are external to the autonomous system, but are imported into OSPFv3 as Type 1 or Type 2 external routes; **internal** for routes that are internal to a specific autonomous system; **nssa-external** [**1**|**2**] for routes that are external to the autonomous system, but are imported into OSPFv3 in an NSSA for IPv6 as Type 1 or Type 2 external routes.
- The **tag** *tag-value* keyword-argument pair specifies the 32-bit decimal value that is attached to each external route, which may be used to communicate information between ASBRs. If none is specified, then the remote autonomous system number is used for routes from BGP and EGP. For other protocols, zero is used. Valid values range from 0 to 4294967295.
- The **route-map** keyword specifies the route map to check for filtering the importing of routes from the source routing protocol to the current routing protocol. If this keyword is not specified, all routes are

redistributed. If this keyword is specified, but no route map tags are listed, no routes are imported. The *map-tag* argument identifies a configured route map.

Configure Graceful Restart

The ASA may experience some known failure situations, that should not affect packet forwarding across the switching platform. The Non-Stop Forwarding (NSF) capability allows data forwarding to continue along known routes, while the routing protocol information is being restored.

In a high availability mode, the OSPF process restarts when the active unit becomes inactive and the standby unit becomes the new active. Similarly, in a cluster mode, the OSPF process restarts when the control unit becomes inactive and the data unit is elected as the new control unit. Such OSPF transitioning processes involve a considerable amount of delay. You can configure NSF to avoid traffic loss during the OSPF process state change. The NSF capability is also useful when there is a scheduled hitless software upgrade.

Graceful restart is supported on both OSPFv2 and OSPFv3. You can configure graceful restart on OSPFv2 by using either using NSF Cisco (RFC 4811 and RFC 4812) or NSF IETF (RFC 3623). You can configure graceful restart on OSPFv3 using graceful-restart (RFC 5187).

Configuring the NSF graceful-restart feature involves two steps; configuring capabilities and configuring a device as NSF-capable or NSF-aware. A NSF-capable device can indicate its own restart activities to neighbors and a NSF-aware device can help a restarting neighbor.

A device can be configured as NSF-capable or NSF-aware, depending on some conditions:

- A device can be configured as NSF-aware irrespective of the mode in which it is.
- A device has to be in either Failover or Spanned Etherchannel (L2) cluster mode to be configured as NSF-capable.
- For a device to be either NSF-aware or NSF-capable, it should be configured with the capability of handling opaque Link State Advertisements (LSAs)/ Link Local Signaling (LLS) block as required.



Note When fast hellos are configured for OSPFv2, graceful restart does not occur when the active unit reloads and the standby unit becomes active. This is because the time taken for the role change is more than the configured dead interval.

Configure Capabilities

The Cisco NSF Graceful Restart mechanism depends on the LLS capability as it sends an LLS block with the RS-bit set in the Hello packet, to indicate the restart activity. The IETF NSF mechanism depends on the opaque LSA capability as it sends opaque-LSAs of type-9 to indicate the restart activity. To configure capabilities enter the following commands:

Procedure

Step 1 Create an OSPF routing process and enters router configuration mode for the OSPF process that you want to redistribute:

router ospf *process_id*

Example:

```
ciscoasa(config)# router ospf 2
```

The process_id argument is an internally used identifier for this routing process and can be any positive integer. This ID does not have to match the ID on any other device; it is for internal use only. You can use a maximum of two processes.

Step 2 Enable the use of LLS data block or opaque LSAs to enable NSF:

capability {lls|opaque}

The lls keyword is used to enable LLS capability for Cisco NSF Graceful Restart mechanism.

The opaque keyword is used to enable opaque LSA capability for IETF NSF Graceful Restart mechanism.

Configuring Graceful Restart for OSPFv2

There are two graceful restart mechanisms for OSPFv2, Cisco NSF and IETF NSF. Only one of these graceful restart mechanisms can be configured at a time for an ospf instance. An NSF-aware device can be configured as both Cisco NSF helper and IETF NSF helper but a NSF-capable device can be configured in either Cisco NSF or IETF NSF mode at a time for an ospf instance.

Configure Cisco NSF Graceful Restart for OSPFv2

Configure Cisco NSF Graceful Restart for OSPFv2, for a NSF-capable or NSF-aware device.

Procedure

Step 1 Enable Cisco NSF on a NSF-capable device:

nsf cisco [enforce global]

Example:

```
ciscoasa(config-router)# nsf cisco
```

The enforce global keyword cancels NSF restart when non-NSF-aware neighbor devices are detected.

Step 2 Enable Cisco NSF helper mode on NSF-aware device:

capability {lls|opaque}

Example:

```
ciscoasa(config-router)# capability lls
```

This command is enabled by default. Using the no form of the command disables it.

Configure IETF NSF Graceful Restart for OSPFv2

Configure IETF NSF Graceful Restart for OSPFv2, for a NSF-capable or NSF-aware device.

Procedure

Step 1 Enable IETF NSF on a NSF-capable device:

nsf ietf [restart-interval *seconds***]**

Example:

```
ciscoasa(config-router)# nsf ietf restart-interval 80
```

You can specify the length of the graceful restart interval, in seconds. Valid values are from 1 to 1800 seconds. The default value is 120 seconds.

Graceful restart might be terminated when restart interval is configured with a value less than the time taken for the adjacency to come up. For example, a restart interval below 30 seconds, is not supported.

Step 2 Enable IETF NSF helper mode on NSF-aware device:

nsf ietf helper [strict-lsa-checking]

Example:

```
ciscoasa(config-router)# nsf ietf helper
```

The strict-LSA-checking keyword indicates that the helper router will terminate the process of the restarting router if it detects that there is a change to a LSA that would be flooded to the restarting router, or if there is a changed LSA on the retransmission list of the restarting router when the graceful restart process is initiated.

This command is enabled by default. Using the no form of the command disables it.

Configuring Graceful Restart for OSPFv3

Configuring the NSF graceful-restart feature for OSPFv3 involves two steps; configuring a device to be NSF-capable and then configuring a device to be NSF-aware.

Procedure

Step 1 Enable IPv6 processing on an interface that has not been configured with an explicit IPv6 address:

interface physical_interface ipv6 enable

Example:

```
ciscoasa(config)# interface ethernet 0/0
ciscoasa(config-if)# ipv6 enable
```

The physical_interface argument identifies the interface that participates in OSPFv3 NSF.

Step 2 Enable graceful-restart for OSPFv3 on a NSF-capable device:

graceful-restart [restart interval seconds]

Example:

```
ciscoasa(config-router)# graceful-restart restart interval 80
```

The restart interval seconds specifies the length of the graceful restart interval, in seconds. Valid values are from 1 to 1800 seconds. The default value is 120 seconds.

Graceful restart might be terminated when restart interval is configured with a value less than the time taken for the adjacency to come up.For example, a restart interval below 30 seconds, is not supported.

Step 3 Enable graceful-restart for OSPFv3 on a NSF-aware device:

graceful-restart helper [strict-lsa-checking]

Example:

```
ciscoasa(config-router)# graceful-restart helper strict-lsa-checking
```

The strict-LSA-checking keyword indicates that the helper router will terminate the process of the restarting router if it detects that there is a change to a LSA that would be flooded to the restarting router, or if there is a changed LSA on the retransmission list of the restarting router when the graceful restart process is initiated.

The graceful-restart helper mode is enabled by default.

Configuring Graceful Restart Wait Timer for OSPF

OSPF routers are expected to set the RS-bit in the EO-TLV attached to a Hello packet when it is not known that all neighbors are listed in the packet, but the restarting routers require to preserve their adjacencies. However, the RS-bit value must not be longer than the RouterDeadInterval seconds. Hence the **timers nsf wait** command is introduced to set the RS-bit in Hello packets lesser than RouterDeadInterval seconds. The default value of NSF wait timer is 20 seconds.

Before you begin

- To configure Cisco NSF wait time for OSPF, the device must be NSF-aware or NSF-capable.

Procedure

Step 1 Enter into OSPF router configuration mode.

Example:

```
ciscoasa(config)# router ospf
```

Step 2 Enter timers and specify nsf.

Example:

```
ciscoasa(config-router)# timers?
router mode commands/options:
  lsa       OSPF LSA timers
  nsf       OSPF NSF timer
  pacing    OSPF pacing timers
  throttle  OSPF throttle timers
ciscoasa(config-router)# timers nsf ?
```

Step 3 Enter the graceful restart wait interval. This value can range between 1 and 65535.

Example:

```
ciscoasa(config-router)# timers nsf wait 200
```

By using the graceful restart wait interval, you can ensure that the wait interval is not longer than the router dead interval.

Remove the OSPFv2 Configuration

Remove the OSPFv2 configuration.

Procedure

Remove the entire OSPFv2 configuration that you have enabled.

clear configure router ospf *pid*

Example:

```
ciscoasa(config)# clear configure router ospf 1000
```

After the configuration is cleared, you must reconfigure OSPF using the **router ospf** command.

Remove the OSPFv3 Configuration

Remove the OSPFv3 configuration.

Procedure

Remove the entire OSPFv3 configuration that you have enabled:

clear configure ipv6 router ospf *process-id*

Example:

```
ciscoasa(config)# clear configure ipv6 router ospf 1000
```

After the configuration is cleared, you must reconfigure OSPFv3 using the **ipv6 router ospf** command.

Example for OSPFv2

The following example shows how to enable and configure OSPFv2 with various optional processes:

1. To enable OSPFv2, enter the following commands:

```
ciscoasa(config)# router ospf 2
ciscoasa(config-rtr)# network 10.0.0.0 255.0.0.0 area 0
```

2. (Optional) To redistribute routes from one OSPFv2 process to another OSPFv2 process, enter the following commands:

```
ciscoasa(config)# route-map 1-to-2 permit
ciscoasa(config-route-map)# match metric 1
ciscoasa(config-route-map)# set metric 5
ciscoasa(config-route-map)# set metric-type type-1
ciscoasa(config-route-map)# router ospf 2
ciscoasa(config-rtr)# redistribute ospf 1 route-map 1-to-2
```

3. (Optional) To configure OSPFv2 interface parameters, enter the following commands:

```
ciscoasa(config)# router ospf 2
ciscoasa(config-rtr)# network 10.0.0.0 255.0.0.0 area 0
ciscoasa(config-rtr)# interface inside
ciscoasa(config-interface)# ospf cost 20
ciscoasa(config-interface)# ospf retransmit-interval 15
ciscoasa(config-interface)# ospf transmit-delay 10
ciscoasa(config-interface)# ospf priority 20
ciscoasa(config-interface)# ospf hello-interval 10
ciscoasa(config-interface)# ospf dead-interval 40
ciscoasa(config-interface)# ospf authentication-key cisco
ciscoasa(config-interface)# ospf message-digest-key 1 md5 cisco
```

```
ciscoasa(config-interface)# ospf authentication message-digest
```

4. (Optional) To configure OSPFv2 area parameters, enter the following commands:

```
ciscoasa(config)# router ospf 2
ciscoasa(config-rtr)# area 0 authentication
ciscoasa(config-rtr)# area 0 authentication message-digest
ciscoasa(config-rtr)# area 17 stub
ciscoasa(config-rtr)# area 17 default-cost 20
```

5. (Optional) To configure the route calculation timers and show the log neighbor up and down messages, enter the following commands:

```
ciscoasa(config-rtr)# timers spf 10 120
ciscoasa(config-rtr)# log-adj-changes [detail]
```

6. (Optional) To show current OSPFv2 configuration settings, enter the **show ospf** command.

   The following is sample output from the **show ospf** command:

```
ciscoasa(config)# show ospf

Routing Process "ospf 2" with ID 10.1.89.2 and Domain ID 0.0.0.2
Supports only single TOS(TOS0) routes
Supports opaque LSA
SPF schedule delay 5 secs, Hold time between two SPFs 10 secs
Minimum LSA interval 5 secs. Minimum LSA arrival 1 secs
Number of external LSA 5. Checksum Sum 0x 26da6
Number of opaque AS LSA 0. Checksum Sum 0x     0
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
External flood list length 0
    Area BACKBONE(0)
        Number of interfaces in this area is 1
        Area has no authentication
        SPF algorithm executed 2 times
        Area ranges are
        Number of LSA 5. Checksum Sum 0x 209a3
        Number of opaque link LSA 0. Checksum Sum 0x     0
        Number of DCbitless LSA 0
        Number of indication LSA 0
        Number of DoNotAge LSA 0
        Flood list length 0
```

7. To clear the OSPFv2 configuration, enter the following command:

```
ciscoasa(config)# clear configure router ospf pid
```

Examples for OSPFv3

The following example shows how to enable and configure OSPFv3 at the interface level:

```
ciscoasa (config)# interface GigabitEthernet3/1
ciscoasa (config-if)# ipv6 enable
ciscoasa (config-if)# ipv6 ospf 1 area 1
```

The following is sample output from the **show running-config ipv6** command:

```
ciscoasa (config)# show running-config ipv6
ipv6 router ospf 1
 log-adjacency-changes
```

The following is sample output from the **show running-config interface** command:

```
ciscoasa (config-if)# show running-config interface GigabitEthernet3/1
interface GigabitEthernet3/1
 nameif fda
 security-level 100
 ip address 1.1.11.1 255.255.255.0 standby 1.1.11.2
 ipv6 address 9098::10/64 standby 9098::11
 ipv6 enable
 ipv6 ospf 1 area 1
```

The following examples show how to configure OSPFv3-specific interfaces:

```
ciscoasa (config)# interface GigabitEthernet3/1
ciscoasa (config-if)# nameif fda
ciscoasa (config-if)# security-level 100
ciscoasa (config-if)# ip address 10.1.11.1 255.255.255.0 standby 10.1.11.2
ciscoasa (config-if)# ipv6 address 9098::10/64 standby 9098::11
ciscoasa (config-if)# ipv6 enable
ciscoasa (config-if)# ipv6 ospf cost 900
ciscoasa (config-if)# ipv6 ospf hello-interval 20
ciscoasa (config-if)# ipv6 ospf network broadcast
ciscoasa (config-if)# ipv6 ospf database-filter all out
ciscoasa (config-if)# ipv6 ospf flood-reduction
ciscoasa (config-if)# ipv6 ospf mtu-ignore
ciscoasa (config-if)# ipv6 ospf 1 area 1 instance 100
ciscoasa (config-if)# ipv6 ospf encryption ipsec spi 890 esp null md5
123456789012345678901234567890 12

ciscoasa (config)# ipv6 router ospf 1
ciscoasa (config)# area 1 nssa
ciscoasa (config)# distance ospf intra-area 190 inter-area 100 external 100
ciscoasa (config)# timers lsa arrival 900
ciscoasa (config)# timers pacing flood 100
ciscoasa (config)# timers throttle lsa 900 900 900
ciscoasa (config)# passive-interface fda
ciscoasa (config)# log-adjacency-changes
ciscoasa (config)# redistribute connected metric 100 metric-type 1 tag 700
```

For an example of how to configure an OSPFv3 virtual link, see the following URL:

http://www.cisco.com/en/US/tech/tk365/technologies_configuration_example09186a0080b8fd06.shtml

Monitoring OSPF

You can display specific statistics such as the contents of IP routing tables, caches, and databases. You can also use the information provided to determine resource utilization and solve network problems. You can also display information about node reachability and discover the routing path that your device packets are taking through the network.

To monitor or display various OSPFv2 routing statistics, enter one of the following commands:

Command	Purpose
show ospf [*process-id* [*area-id*]]	Displays general information about OSPFv2 routing processes.
show ospf border-routers	Displays the internal OSPFv2 routing table entries to the ABR and ASBR.
show ospf [*process-id* [*area-id*]] database	Displays lists of information related to the OSPFv2 database for a specific router.
show ospf flood-list *if-name*	Displays a list of LSAs waiting to be flooded over an interface (to observe OSPF v2packet pacing). OSPFv2 update packets are automatically paced so they are not sent less than 33 milliseconds apart. Without pacing, some update packets could get lost in situations where the link is slow, a neighbor could not receive the updates quickly enough, or the router could run out of buffer space. For example, without pacing, packets might be dropped if either of the following topologies exist: • A fast router is connected to a slower router over a point-to-point link. • During flooding, several neighbors send updates to a single router at the same time. Pacing is also used between resends to increase efficiency and minimize lost retransmissions. You also can display the LSAs waiting to be sent out of an interface. Pacing enables OSPFv2 update and retransmission packets to be sent more efficiently. There are no configuration tasks for this feature; it occurs automatically.
show ospf interface [*if_name*]	Displays OSPFv2-related interface information.
show ospf neighbor [*interface-name*] [*neighbor-id*] [**detail**]	Displays OSPFv2 neighbor information on a per-interface basis.
show ospf request-list *neighbor if_name*	Displays a list of all LSAs requested by a router.

Command	Purpose
show ospf retransmission-list *neighbor if_name*	Displays a list of all LSAs waiting to be resent.
show ospf [*process-id*] summary-address	Displays a list of all summary address redistribution information configured under an OSPFv2 process.
show ospf [*process-id*] **traffic**	Displays a list of different types of packets being sent or received by a specific OSPFv2 instance.
show ospf [*process-id*] virtual-links	Displays OSPFv2-related virtual links information.
show route cluster	Displays additional OSPFv2 route synchronization information in clustering.

To monitor or display various OSPFv3 routing statistics, enter one of the following commands:

Command	Purpose
show ipv6 ospf [*process-id* [*area-id*]]	Displays general information about OSPFv3 routing processes.
show ipv6 ospf [*process-id*] **border-routers**	Displays the internal OSPFv3 routing table entries to the ABR and ASBR.
show ipv6 ospf [*process-id* [*area-id*]] database [**external** | **inter-area prefix** | **inter-area-router** | **network** | **nssa-external** | **router** | **area** | **as** | **ref-lsa** | [*destination-router-id*] [**prefix** *ipv6-prefix*] [*link-state-id*]] [**link** [**interface** *interface-name*] [**adv-router** *router-id*] | **self-originate**] [**internal**] [**database-summary**]	Displays lists of information related to the OSPFv3 database for a specific router.
show ipv6 ospf [*process-id* [*area-id*]] **events**	Displays OSPFv3 event information.

Command	Purpose
show ipv6 ospf [*process-id*] [*area-id*] flood-list *interface-type interface-number*	Displays a list of LSAs waiting to be flooded over an interface (to observe OSPFv3 packet pacing). OSPFv3 update packets are automatically paced so they are not sent less than 33 milliseconds apart. Without pacing, some update packets could get lost in situations where the link is slow, a neighbor could not receive the updates quickly enough, or the router could run out of buffer space. For example, without pacing, packets might be dropped if either of the following topologies exist: • A fast router is connected to a slower router over a point-to-point link. • During flooding, several neighbors send updates to a single router at the same time. Pacing is also used between retransmissions to increase efficiency and minimize lost retransmissions. You also can display the LSAs waiting to be sent out of an interface. Pacing enables OSPFv3 update and retransmission packets to be sent more efficiently. There are no configuration tasks for this feature; it occurs automatically.
show ipv6 ospf [*process-id*] [*area-id*] **interface** [*type number*] [**brief**]	Displays OSPFv3-related interface information.
show ipv6 ospf neighbor [*process-id*] [*area-id*] [*interface-type interface-number*] [*neighbor-id*] [**detail**]	Displays OSPFv3 neighbor information on a per-interface basis.
show ipv6 ospf [*process-id*] [*area-id*] **request-list** [*neighbor*] [*interface*] [*interface-neighbor*]	Displays a list of all LSAs requested by a router.
show ipv6 ospf [*process-id*] [*area-id*] retransmission-list [*neighbor*] [*interface*] [*interface-neighbor*]	Displays a list of all LSAs waiting to be resent.
show ipv6 ospf statistic [*process-id*] [**detail**]	Displays various OSPFv3 statistics.
show ipv6 ospf [*process-id*] summary-prefix	Displays a list of all summary address redistribution information configured under an OSPFv3 process.
show ipv6 ospf [*process-id*] **timers** [**lsa-group** | **rate-limit**]	Displays OSPFv3 timers information.
show ipv6 ospf [*process-id*] **traffic** [*interface_name*]	Displays OSPFv3 traffic-related statistics.
show ipv6 ospf virtual-links	Displays OSPFv3-related virtual links information.

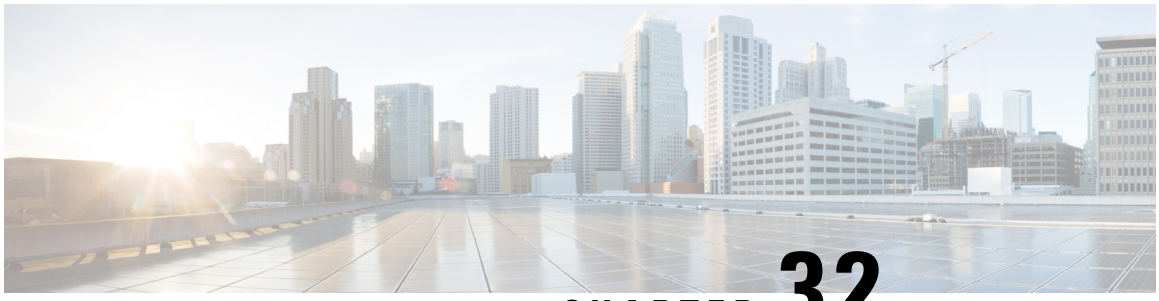
Command	Purpose
show ipv6 route cluster [**failover**] [**cluster**] [**interface**] [**ospf**] [**summary**]	Displays the IPv6 routing table sequence number, IPv6 reconvergence timer status, and IPv6 routing entries sequence number in a cluster.

History for OSPF

Table 38: Feature History for OSPF

Feature Name	Platform Releases	Feature Information
OSPF Support	7.0(1)	Support was added for route data, authentication, and redistribution and monitoring of routing information using the Open Shortest Path First (OSPF) routing protocol. We introduced the following command: **route ospf**
Dynamic Routing in Multiple Context Mode	9.0(1)	OSPFv2 routing is supported in multiple context mode.
Clustering	9.0(1)	For OSPFv2 and OSPFv3, bulk synchronization, route synchronization, and Spanned EtherChannel load balancing are supported in the clustering environment. We introduced or modified the following commands: **show route cluster**, **show ipv6 route cluster**, **debug route cluster, router-id cluster-pool**.
OSPFv3 Support for IPv6	9.0(1)	OSPFv3 routing is supported for IPv6. We introduced or modified the following commands: **ipv6 ospf**, **ipv6 ospf area**, **ipv6 ospf cost, ipv6 ospf database-filter all out, ipv6 ospf dead-interval, ipv6 ospf encryption, ipv6 ospf hello-interval, ipv6 ospf mtu-ignore, ipv6 ospf neighbor, ipv6 ospf network, ipv6 ospf flood-reduction, ipv6 ospf priority, ipv6 ospf retransmit-interval, ipv6 ospf transmit-delay, ipv6 router ospf, ipv6 router ospf area, ipv6 router ospf default, ipv6 router ospf default-information, ipv6 router ospf distance, ipv6 router ospf exit, ipv6 router ospf ignore, ipv6 router ospf log-adjacency-changes, ipv6 router ospf no, ipv6 router ospf passive-interface, ipv6 router ospf redistribute, ipv6 router ospf router-id, ipv6 router ospf summary-prefix, ipv6 router ospf timers, area encryption, area range, area stub, area nssa, area virtual-link, default, default-information originate, distance, ignore lsa mospf, log-adjacency-changes, redistribute, router-id, summary-prefix, timers lsa arrival, timers pacing flood, timers pacing lsa-group, timers pacing retransmission, timers throttle, show ipv6 ospf, show ipv6 ospf border-routers, show ipv6 ospf database, show ipv6 ospf events, show ipv6 ospf flood-list, show ipv6 ospf graceful-restart, show ipv6 ospf interface, show ipv6 ospf neighbor, show ipv6 ospf request-list, show ipv6 ospf retransmission-list, show ipv6 ospf statistic, show ipv6 ospf summary-prefix, show ipv6 ospf timers, show ipv6 ospf traffic, show ipv6 ospf virtual-links, show ospf, show running-config ipv6 router, clear ipv6 ospf, clear configure ipv6 router, debug ospfv3, ipv6 ospf neighbor**.

Feature Name	Platform Releases	Feature Information
OSPF support for Fast Hellos	9.2(1)	OSPF supports the Fast Hello Packets feature, resulting in a configuration that results in faster convergence in an OSPF network. We modified the following command: **ospf dead-interval**
Timers	9.2(1)	New OSPF timers were added; old ones were deprecated. We introduced the following commands: timers lsa arrival, timers pacing, timers throttle We removed the following commands: Timers spf, timers lsa-grouping-pacing
Route filtering using access-list	9.2(1)	Route filtering using ACL is now supported. We introduced the following command: distribute-list
OSPF Monitoring enhancements	9.2(1)	Additional OSPF monitoring information was added. We modified the following commands: show ospf events, show ospf rib, show ospf statistics, show ospf border-routers [detail], show ospf interface brief
OSPF redistribute BGP	9.2(1)	OSPF redistribution feature was added. We added the following command: redistribute bgp
OSPF Support for Non-Stop Forwarding (NSF)	9.3(1)	OSPFv2 and OSPFv3 support for NSF was added. We added the following commands: capability, nsf cisco, nsf cisco helper, nsf ietf, nsf ietf helper, nsf ietf helper strict-lsa-checking, graceful-restart, graceful-restart helper, graceful-restart helper strict-lsa-checking
OSPF Support for Non-Stop Forwarding (NSF)	9.13(1)	NSF wait timer was added. We added a new command for setting the timer for the NSF restart interval. This command was introduced to ensure the wait interval is not longer that the router dead interval. We introduced the following command: **timers nsf wait <seconds>**



CHAPTER 32

IS-IS

This chapter describes the Intermediate System to Intermediate System (IS-IS) routing protocol.

- About IS-IS, on page 999
- Prerequisites for IS-IS, on page 1005
- Guidelines for IS-IS, on page 1005
- Configure IS-IS, on page 1006
- Monitoring IS-IS, on page 1036
- History for IS-IS, on page 1038
- Examples for IS-IS, on page 1038

About IS-IS

IS-IS routing protocol is a link state Interior Gateway Protocol (IGP). Link-state protocols are characterized by the propagation of the information required to build a complete network connectivity map on each participating device. That map is then used to calculate the shortest path to destinations. The IS-IS implementation supports IPv4 and IPv6.

You can divide a routing domain into one or more subdomains. Each subdomain is called an area and is assigned an area address. Routing within an area is known as Level-1 routing. Routing between Level-1 areas is known as Level-2 routing. A router is referred to as an Intermediate System (IS). An IS can operate at Level 1, Level 2, or both. ISes that operate at Level 1 exchange routing information with other Level-1 ISes in the same area. ISes that operate at Level 2 exchange routing information with other Level-2 routers regardless of whether they are in the same Level-1 area. The set of Level-2 routers and the links that interconnect them form the Level-2 subdomain, which must not be partitioned in order for routing to work properly.

About NET

An IS is identified by an address known as a Network Entity Title (NET). The NET is the address of a Network Service Access Point (NSAP), which identifies an instance of the IS-IS routing protocol running on an IS. The NET is 8 to 20 octets in length and has the following three parts:

- Area address—This field is 1 to 13 octets in length and is composed of high-order octets of the address.

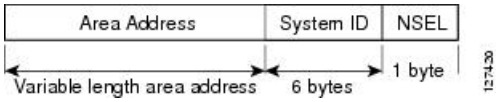
Note You can assign multiple area addresses to an IS-IS instance; in this case, all area addresses are considered synonymous. Multiple synonymous area addresses are useful when merging or splitting areas in the domain. Once the merge or split has been completed, you do not need to assign more than one area address to an IS-IS instance.

- System ID—This field is 6 octets long and immediately follows the area address. When the IS operates at Level 1, the system ID must be unique among all the Level-1 devices in the same area. When the IS operates at Level 2, the system ID must be unique among all devices in the domain.

Note You assign one system ID to an IS instance.

- NSEL—The N-selector field is 1 octet in length and immediately follows the system ID. It must be set to 00.

Figure 60: NET Format



IS-IS Dynamic Hostname

In the IS-IS routing domain, the system ID is used to represent each ASA. The system ID is part of the NET that is configured for each IS-IS ASA. For example, an ASA with a configured NET of 49.0001.0023.0003.000a.00 has a system ID of 0023.0003.000a. ASA-name-to-system-ID mapping is difficult for network administrators to remember during maintenance and troubleshooting on the ASAs.

Entering the **show isis hostname** command displays the entries in the system-ID-to-ASA-name mapping table.

The dynamic hostname mechanism uses link-state protocol (LSP) flooding to distribute the ASA-name-to-system-ID mapping information across the entire network. Every ASA on the network will try to install the system ID-to-ASA name mapping information in its routing table.

If an ASA that has been advertising the dynamic name type, length, value (TLV) on the network suddenly stops the advertisement, the mapping information last received will remain in the dynamic host mapping table for up to one hour, allowing the network administrator to display the entries in the mapping table during a time when the network experiences problems.

IS-IS PDU Types

ISes exchange routing information with their peers using protocol data units (PDUs). Intermediate System-to-Intermediate System Hello PDUs (IIHs), Link-State PDUs (LSPs), and Sequence Number PDUs (SNPs) types of PDUs are used.

IIHs

IIHs are exchanged between IS neighbors on circuits that have the IS-IS protocol enabled. IIHs include the system ID of the sender, the assigned area address(es), and the identity of neighbors on that circuit that are known to the sending IS. Additional optional information can also be included.

There are two types of IIHs:

- Level-1 LAN IIHs—These are sent on multiaccess circuits when the sending IS operates as a Level-1 device on that circuit.
- Level-2 LAN IIHs—These are sent on multiaccess circuits when the sending IS operates as a Level-2 device on that circuit.

LSPs

An IS generates LSPs to advertise its neighbors and the destinations that are directly connected to the IS. An LSP is uniquely identified by the following:

- System ID of the IS that generated the LSP
- Pseudonode ID—This value is always 0 except when the LSP is a pseudonode LSP
- LSP number (0 to 255)
- 32-bit sequence number

Whenever a new version of an LSP is generated, the sequence number is incremented.

Level-1 LSPs are generated by ISs that support Level 1. The Level-1 LSPs are flooded throughout the Level-1 area. The set of Level-1 LSPs generated by all Level-1 ISs in an area is the Level-1 LSP Database (LSPDB). All Level-1 ISs in an area have an identical Level-1 LSPDB and therefore have an identical network connectivity map for the area.

Level-2 LSPs are generated by ISs that support Level 2. Level-2 LSPs are flooded throughout the Level-2 subdomain. The set of Level-2 LSPs generated by all Level-2 ISs in the domain is the Level-2 LSP Database (LSPDB). All Level-2 ISs have an identical Level-2 LSPDB and therefore have an identical connectivity map for the Level-2 subdomain.

SNPs

SNPs contain a summary description of one or more LSPs. There are two types of SNPs for both Level 1 and Level 2:

- Complete Sequence Number PDUs (CSNPs) are used to send a summary of the LSPDB that an IS has for a given level.
- Partial Sequence Number PDUs (PSNPs) are used to send a summary of a subset of the LSPs for a given level that an IS either has in its database or needs to obtain.

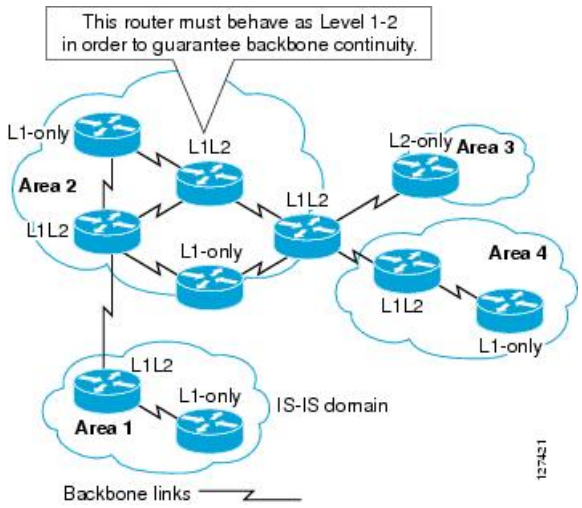
Operation of IS-IS on Multiaccess Circuits

Multiaccess circuits support multiple ISes, that is, two or more operating on the circuit. For multiaccess circuits a necessary prerequisite is the ability to address multiple systems using a multicast or broadcast address. An IS that supports Level 1 on a multiaccess circuit sends Level-1 LAN IIHs on the circuit. An IS that supports Level 2 on a multiaccess circuit sends Level-2 LAN IIHs on the circuit. ISes form separate adjacencies for each level with neighbor ISes on the circuit.

An IS forms a Level-1 adjacency with other ISes that support Level 1 on the circuit and has a matching area address. Two ISes with disjointed sets of area addresses supporting Level 1 on the same multiaccess circuit is NOT supported. An IS forms a Level-2 adjacency with other ISes that support Level 2 on the circuit.

The devices in the IS-IS network topology in the following figure perform Level 1, Level 2, or Level 1 and 2 routing along the backbone of the network.

Figure 61: Level-1, Level-2, Level 1-2 Devices in an IS-IS Network Topology



IS-IS Election of the Designated IS

If each IS advertised all of its adjacencies on a multiaccess circuit in its LSPs, the total number of advertisements required would be N 2 (where N is the number of ISes that operate at a given level on the circuit). To address this scalability issue, IS-IS defines a pseudonode to represent the multiaccess circuit. All ISes that operate on the circuit at a given level elect one of the ISes to act as the Designated Intermediate System (DIS) on that circuit. A DIS is elected for each level that is active on the circuit.

The DIS is responsible for issuing pseudonode LSPs. The pseudonode LSPs include neighbor advertisements for all of the ISes that operate on that circuit. All ISes that operate on the circuit (including the DIS) provide a neighbor advertisement to the pseudonode in their non-pseudonode LSPs and do not advertise any of their neighbors on the multiaccess circuit. In this way the total number of advertisements required varies as a function of N-the number of ISes that operate on the circuit.

A pseudonode LSP is uniquely classified by the following identifiers:

- System ID of the DIS that generated the LSP
- Pseudonode ID (ALWAYS NON-ZERO)
- LSP number (0 to 255)
- 32-bit sequence number

The nonzero pseudonode ID is what differentiates a pseudonode LSP from a non-pseudonode LSP and is chosen by the DIS to be unique among any other LAN circuits for which it is also the DIS at this level.

The DIS is also responsible for sending periodic CSNPs on the circuit. This provides a complete summary description of the current contents of the LSPDB on the DIS. Other ISes on the circuit can then perform the

following activities, which efficiently and reliably synchronizes the LSPDBs of all ISes on a multiaccess circuit:

- Flood LSPs that are absent from or are newer than those that are described in the CSNPs sent by the DIS.
- Request an LSP by sending a PSNP for LSPs that are described in the CSNPs sent by the DIS that are absent from the local database or older than what is described in the CSNP set.

IS-IS LSPDB Synchronization

Proper operation of IS-IS requires a reliable and efficient process to synchronize the LSPDBs on each IS. In IS-IS this process is called the update process. The update process operates independently at each supported level. Locally generated LSPs are always new LSPs. LSPs received from a neighbor on a circuit may be generated by some other IS or may be a copy of an LSP generated by the local IS. Received LSPs can be older, the same age, or newer than the current contents of the local LSPDB.

Handling Newer LSPs

When a newer LSP is added to the local LSPDB, it replaces an older copy of the same LSP in the LSPDB. The newer LSP is marked to be sent on all circuits on which the IS currently has an adjacency in the UP state at the level associated with the newer LSP—excluding the circuit on which the newer LSP was received.

For multiaccess circuits, the IS floods the newer LSP once. The IS examines the set of CNSPs that are sent periodically by the DIS for the multiaccess circuit. If the local LSPDB contains one or more LSPs that are newer than what is described in the CSNP set (this includes LSPs that are absent from the CSNP set), those LSPs are reflooded over the multiaccess circuit. If the local LSPDB contains one or more LSPs that are older than what is described in the CSNP set (this includes LSPs described in the CSNP set that are absent from the local LSPDB), a PSNP is sent on the multiaccess circuit with descriptions of the LSPs that require updating. The DIS for the multiaccess circuit responds by sending the requested LSPs.

Handling Older LSPs

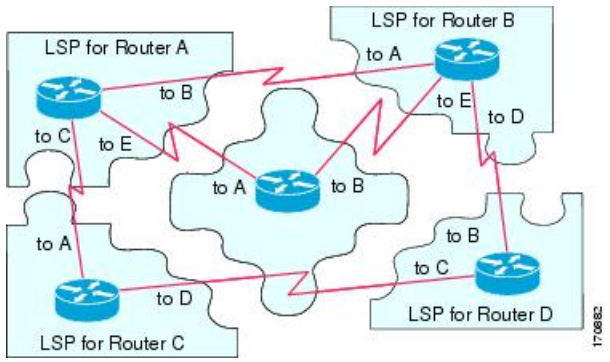
An IS may receive an LSP that is older than the copy in the local LSPDB. An IS may receive an SNP (complete or partial) that describes an LSP that is older than the copy in the local LSPDB. In both cases the IS marks the LSP in the local database to be flooded on the circuit on which the older LSP or SNP that contained the older LSP was received. Actions taken are the same as described above after a new LSP is added to the local database.

Handling Same-Age LSPs

Because of the distributed nature of the update process, it is possible than an IS may receive copies of an LSP that is the same as the current contents of the local LSPDB. In multiaccess circuits receipt of a same-age LSP is ignored. Periodic transmission of a CSNP set by the DIS for that circuit serves as an implicit acknowledgment to the sender that the LSP has been received.

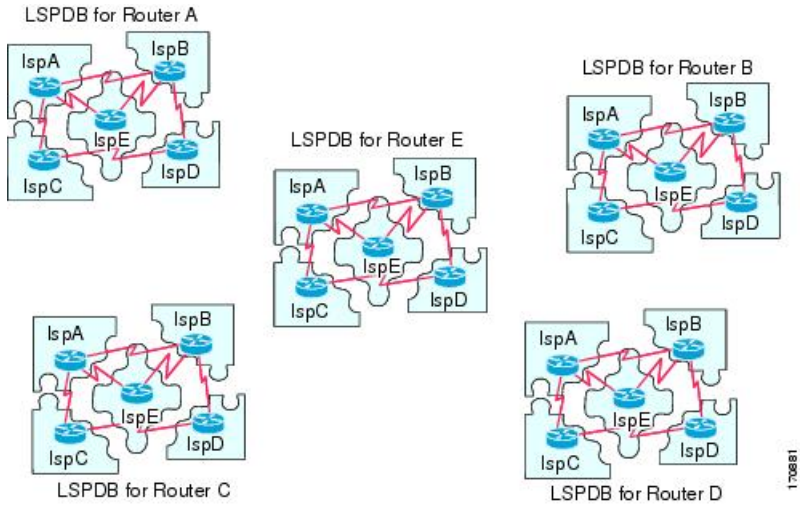
The following figure shows how LSPs are used to create a network map. Think of the network topology as a jigsaw puzzle. Each LSP (representing an IS) is one of the pieces. It is applicable to all Level-1 devices in an area or to all Level-2 devices in a Level-2 subdomain.

Figure 62: IS-IS Network Map



The following figure shows each device in the IS-IS network with its fully updated link-state database after the adjacencies have been formed among the neighbor devices. It is applicable to all Level-1 devices in an area or to all Level-2 devices in a Level-2 subdomain.

Figure 63: IS-IS Devices with Synchronized LSPDBs



IS-IS Shortest Path Calculation

When the contents of the LSPDB change, each IS independently reruns a shortest path calculation. The algorithm is based on the well-known Dijkstra algorithm for finding the shortest paths along a directed graph where the ISes are the vertices of the graph and the links between the ISes are edges with a nonnegative weight. A two-way connectivity check is performed before considering a link between two ISes as part of the graph. This prevents the use of stale information in the LSPDB, for example, when one IS is no longer operating in the network but did not purge the set of LSPs that it generated before stopping operation.

The output of the SPF is a set of tuples (destination, next hop). The destinations are protocol-specific. Multiple equal-cost paths are supported, in which case multiple next hops would be associated with the same destination.

Independent SPFs are performed for each level supported by the IS. When the same destination is reachable by both Level-1 and Level-2 paths, the Level-1 path is preferred.

A Level-2 IS that indicates that it has one or more Level-2 neighbors in other areas may be used by Level-1 devices in the same area as the path of last resort, also called the default route. The Level-2 IS indicates its attachment to other areas by setting an attached bit (ATT) in its Level-1 LSP 0.



Note An IS can generate up to 256 LSPs at each level. The LSPs are identified by the numbers 0 through 255. LSP 0 has special properties, including the significance of the setting of the ATT bit to indicate attachment to other areas. When LSPs that are numbered 1 through 255 have the ATT bit set, it is not significant.

IS-IS Shutdown Protocol

You can shut down IS-IS (placing it in an administrative down state) to make changes to the IS-IS protocol configuration without losing your configuration parameters. You can shut down IS-IS at the global IS-IS process level or at the interface level. If the device was rebooted when the protocol was turned off, the protocol would be expected to come back up in the disabled state. When the protocol is set to the administrative down state, network administrators are allowed to administratively turn off the operation of the IS-IS protocol without losing the protocol configuration, to make a series of changes to the protocol configuration without having the operation of the protocol transition through intermediate-and perhaps undesirable-states, and to then reenable the protocol at a suitable time.

Prerequisites for IS-IS

The following prerequisites are necessary before configuring IS-IS:

- Knowledge of IPv4 and IPv6.
- Knowledge of your network design and how you want traffic to flow through it before configuring IS-IS.
- Define areas, prepare an addressing plan for the devices (including defining the NETs), and determine the interfaces that will run IS-IS.
- Before you configure your devices, prepare a matrix of adjacencies that shows what neighbors should be expected in the adjacencies table. This will facilitate verification.

Guidelines for IS-IS

Firewall Mode Guidelines

Supported only in routed firewall mode. Transparent firewall mode is not supported.

Cluster Guidelines

Supported only in Individual Interface mode; Spanned EtherChannel mode is not supported.

Additional Guidelines

IS-IS is not supported with bidirectional forwarding.

Configure IS-IS

This section describes how to enable and configure the IS-IS process on your system.

Procedure

Step 1 Enable IS-IS Routing Globally, on page 1006.
Step 2 Enable IS-IS Authentication, on page 1010.
Step 3 Configure IS-IS LSP, on page 1013
Step 4 Configure IS-IS Summary Addresses, on page 1017.
Step 5 Configure IS-IS Passive Interfaces, on page 1018.
Step 6 Configure IS-IS Interfaces, on page 1019.
Step 7 Configure IS-IS Interface Hello Padding, on page 1023
Step 8 Configure IS-IS IPv4 Address Family, on page 1026.
Step 9 Configure IS-IS IPv6 Address Family, on page 1031.

Enable IS-IS Routing Globally

IS-IS configuration is done in two parts. First, you configure the IS-IS process in global configuration mode, then specify the NET and the routing level for IS-IS in router configuration mode. There are other general parameters you can configure in router configuration mode that may make more sense for your network than configuring them per interface. This section contains those commands.

Second, you enable IS-IS protocol on individual interfaces in interface configuration mode so that the interface participates in dynamic routing and forms adjacencies with neighboring devices. You must enable routing on one or more interfaces before adjacencies can be established and dynamic routing is possible. See Configure IS-IS Interfaces, on page 1019 for the procedures for configuring IS-IS on interfaces.

This procedure describes how to enable IS-IS as an IP routing protocol on the ASA and other general options in router configuration mode.

Before you begin

In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the **changeto context** *name* command.

Procedure

Step 1 Enable IS-IS as a routing protocol on the ASA:

router isis

Example:

```
ciscoasa(config)# router isis
ciscoasa(config-router)#
```

Step 2 Specify the NET for the routing process:

net *network-entity-title*

Example:

```
ciscoasa(config-router)# net 49.1234.aaaa.bbbb.cccc.00
```

The NET identifies the device for IS-IS. See About NET, on page 999 for more information on the NET.

Step 3 (Optional) Assign the routing level for the IS-IS routing process:

is-type [level-1 | level-2-only | level-1-2]

Example:

```
ciscoasa(config-router)# is-type level-1
```

- (Optional) **level-1**—Indicates intra-area routing. The ASA only learns destinations inside its area.
- (Optional) **level-2-only**—Indicates inter-area routing. The ASA is part of the back bone and does not communicate with Level-1 routers in its own area.
- (Optional) **level-1-2**—The ASA performs both Level 1 and Level 2 routing. This router runs two instances of the routing process. It has one LSDB for destinations inside the area (Level 1 routing) and runs an SPF calculation to discover the area topology. It also has another LSDB with LSPs of all other backbone (Level 2) routers, and runs another SPF calculation to discover the topology of the backbone, and the existence of all other areas.

In conventional IS-IS configurations, the ASA acts as both a Level 1 (intra-area) and a Level 2 (inter-area) router. In multi-area IS-IS configurations, the first instance of the IS-IS routing process configured is by default a Level 1-2 (intra-area and inter-area) router. The remaining instances of the IS-IS process configured by default are Level 1 routers.

Note
We highly recommend that you configure the type of IS-IS routing process.

Step 4 Enable IS-IS dynamic hostname capability on the ASA:

hostname dynamic

This command is enabled by default. See IS-IS Dynamic Hostname, on page 1000 for detailed information about the dynamic hostname in IS-IS.

Step 5 Configure hello padding for all interfaces on the ASA:

hello padding multi-point

This command is enabled by default. It configures IS-IS hellos to the full MTU size. This allows for early detection of errors that result from transmission problems with large frames or errors that result from mismatched MTUs on adjacent interfaces.

You can disable hello padding (**no hello padding multi-point** for all interfaces on a router for the IS-IS routing process) to avoid wasting network bandwidth in case the MTU of both interfaces is the same or in the case of translational bridging. When hello padding is disabled, the ASA still sends the first five IS-IS hellos padded to the full MTU size to maintain the benefits of discovering MTU mismatches.

Enter the **show clns interface** command in privileged EXEC mode to show that hello padding has been turned off at the router level, See Monitoring IS-IS, on page 1036 for more information.

Step 6 (Optional) Enable the ASA to generate a log message when an NLSP IS-IS adjacency changes state (up or down):

log-adjacency-changes [all]

This command is disabled by default. Logging adjacency changes is useful when monitoring large networks. Messages are in the following form:

Example:

```
%CLNS-5-ADJCHANGE: ISIS: Adjacency to 0000.0000.0034 (Serial0) Up, new adjacency
%CLNS-5-ADJCHANGE: ISIS: Adjacency to 0000.0000.0034 (Serial0) Down, hold time expired
```

all—(Optional) Includes changes generated by non_IIH events.

Step 7 (Optional) Disable the IS-IS protocol so that it cannot form any adjacency on any interface and will clear the LSP database:

protocol shutdown

This command lets you disable the IS-IS protocol for a specific routing instance without removing any existing IS-IS configurations parameters. When you enter this command, the IS-IS protocol continues to run on the router, and you can use the current IS-IS configuration, but IS-IS does not form any adjacencies on any interface, and it also clears the IS-IS LSP database. To disable IS-IS for a specific interface, use the **isis protocol shutdown** command. See Configure IS-IS Interfaces, on page 1019 for the procedure.

Step 8 (Optional) Assign a high priority to an IS-IS IP prefix:

route priority high tag *tag-value*

Example:

```
ciscoasa(config-router)# route priority high tag 100
```

tag *tag-value*—Assigns a high priority to IS-IS IP prefixed with a specific route tag. The range is 1 to 4294967295.

Use this command to tag higher priority IS-IS IP prefixes for faster processing and installation in the global routing table, which results in faster convergence. For example, you can help VoIP gateway addresses get processed first to help VoIP traffic get updated faster than other types of packets.

Step 9 (Optional) Globally change the metric value for all IS-IS interfaces:

metric *default-value* **[level-1 | level-2]**

Example:

```
ciscoasa(config-router)# metric 55 level-1
```

- *default-value*—The metric value to be assigned to the link and used to calculate the path cost via the links to destinations. The range is 1 to 63. The default is 10.
- (Optional) **level-1**— Sets Level 1 IPv4 or IPv6 metric.
- (Optional) **level-2**— Sets Level 2 IPv4 or IPv6 metric.

We recommend you use the **metric** command when you need to change the default metric for all IS-IS interfaces. This prevents user errors, such as unintentionally removing a set metric from an interface without configuring a new value and unintentionally allowing the interface to revert to the default metric of 10, thereby becoming a highly preferred interface in the network.

Step 10 (Optional) Configure the ASA to generate and only accept new-style, length, value objects (TLVs):

metric-style narrow | transition | wide [level-1 | level-2 | level-1-2]

Example:

```
ciscoasa(config-router)# metric-style wide level-1
```

- **narrow**—Uses the old style of TLVs with narrow metrics.
- **transition**— Instructs the ASA to accept both old- and new-style TLVs.
- **wide**—Use the new style of TLVs to carry wider metrics.
- (Optional) **level-1**—Enables this command on routing Level 1.
- (Optional) **level-2**—Enables this command on routing Level 2.
- (Optional) **level-1-2**—Enables this command on routing Level 1 and Level 2.

This command causes the ASA to generate and accept only new-style TLVs, which causes the ASA to use less memory and other resources than if it generates both old-style and new-style TLVs.

Step 11 (Optional) Configure the priority of designated ASAs on all interfaces:

priority *number-value*

Example:

```
ciscoasa(config-router)# priority 80
```

number-value—The priority of the ASA. The range is 0 to 127. The default is 64.

Step 12 (Optional) Configure additional manual addresses for an IS-IS area:

max-area-addresses *number*

Example:

```
ciscoasa(config-router)# max-area-addresses 3
```

number—The number of manual addresses to add. The range is 3 to 254. There is no default value.

This command lets you maximize the size of an IS-IS area by configuring additional manual addresses. You specify the number of addresses you want to add and assign a NET address to create each manual address. See About NET, on page 999 for information on the NET.

Step 13 Configure multipath load sharing for IS-IS:

maximum-paths *number-of-paths*

Example:

```
ciscoasa(config-router)# maximum-paths 8
```

number-of-paths—The number of routes to install in the routing table. The range is 1 to 8. The default is 1.

The **maximum-path** command is used to configure IS-IS multi-load sharing when ECMP is configured in the ASA.

Enable IS-IS Authentication

IS-IS route authentication prevents the introduction of unauthorized or false routing messages from unapproved sources. You can set a password for each IS-IS area or domain to prevent unauthorized routers from injecting false routing information into the link-state database, or you can configure a type of IS-IS authentication, either IS-IS MD5 or enhanced clear text authentication. You can also set authentication per interface. All IS-IS neighbors on interfaces configured for IS-IS message authentication must be configured with the same authentication mode and key for adjacencies to be established.

See About IS-IS, on page 999 for more information on areas and domains.

Before you begin

Before you can enable IS-IS route authentication, you must enable IS-IS and set up an area. See Enable IS-IS Routing Globally, on page 1006 for the procedure.

Procedure

Step 1 Enter IS-IS router configuration mode and configure an IS-IS area authentication password:

area-password *password* [**authenticate snp {validate | send-only}**]

Example:

```
ciscoasa(config)# router isis
ciscoasa(config-router)# area-password track authenticate snp validate
```

- *password*—The password you assign.
- (Optional) **authenticate snp**—Causes the system to insert the password into SNPs.
- **validate**—Causes the system to insert the password into the SNPs and check the password in SNPs that it receives.

- **send-only**—Causes the system to insert only the password into the SNPs, but not check the password in SNPs that it receives. Use this keyword during a software upgrade to ease the transition.

Using this command on all ASAs in an area prevents unauthorized routers from injecting false routing information in the link-state database. However, this password is exchanged as plain text and thus provides only limited security.

The password is inserted in Level 1 (station router level) PDU LSPs, CSNPs, and PSNPs. If you do not specify the **authenticate snp** keyword with either the **validate** or **send-only** keyword, the IS-IS protocol does not insert the password into SNPs.

Step 2 Enter IS-IS router configuration mode and configure an IS-IS domain authentication password:

domain-password *password* [**authenticate snp {validate | send-only}**]

Example:

```
ciscoasa(config-router)# domain-password users2j45 authenticate snp validate
```

- *password*—The password you assign.
- (Optional) **authenticate snp**—Causes the system to insert the password into sequence number PDUs (SNPs).
- **validate**—Causes the system to insert the password into the SNPs and check the password in SNPs that it receives.
- **send-only**—Causes the system to insert only the password into the SNPs, but not check the password in SNPs that it receives. Use this keyword during a software upgrade to ease the transition.

This password is exchanged as plain text and thus provides only limited security.

The password is inserted in Level 2 (area router level) PDU LSPs, CSNPs, and PSNPs. If you do not specify the **authenticate snp** keyword with either the **validate** or **send-only** keyword, the IS-IS protocol does not insert the password into SNPs.

Step 3 Configure the IS-IS instance globally or per interface to have authentication performed only on IS-IS packets being sent (not received):

Router mode:**authentication send-only [level-1 | level-2]**

Example:

```
ciscoasa(config-router)# authentication send-only level-1
```

Interface mode:**isis authentication send-only [level-1 | level-2]**

Example:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# isis authentication send-only level-1
```

- (Optional) **level-1**—Authentication is performed only on Level 1 packets that are being sent (not received).
- (Optional) **level-2**—Authentication is performed only on Level 2 packets that are being sent (not received).

Use this command before configuring the authentication mode and authentication key chain so that the implementation of authentication goes smoothly. If you do not specify Level 1 or Level 2, send only applies to both levels.

Note
ASAs will have more time for the keys to be configured on each ASA if authentication is inserted only on the packets being sent, not checked on packets being received. After all of the ASAs that must communicate are configured with this command, enable the authentication mode and key chain on each ASA.

Step 4 Specify the type of authentication mode used in IS-IS packets for the IS-IS instance globally or per interface:

Router mode:**authentication mode {md5 | text} [level-1 | level-2]**

Example:

```
ciscoasa(config-router)# authentication mode md5 level-1
```

Interface mode:**isis authentication mode {md5 | text} [level-1 | level-2]**

Example:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)#  isis authentication mode md5 level-1
```

- **md5**—Enables Message Digest 5 authentication.
- **text**—Uses clear text authentication.
- (Optional) **level-1**—Enables the specified authentication for Level 1 packets only.
- (Optional) **level-2**—Enables the specified authentication for Level 2 packets only.

If you have clear text authentication configured by using the **area-password** or **domain-password**, the isis authentication mode overrides both of those commands. If you configure **isis authentication mode** and then try to configure the **area-password** or **domain-password**, you are not allowed to do so. If you do not specify Level 1 or Level 2, the mode applies to both levels.

Step 5 Enable authentication for IS-IS globally or per interface:

Router mode: **authentication key [0 | 8]** *password* **[level-1 | level-2]**

Example:

```
ciscoasa(config-router)# authentication key 0 site1 level-1
```

Interface mode: **isis authentication key [0 | 8]** *password* **[level-1 | level-2]**

Example:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# router isis
ciscoasa(config-if)# isis authentication key 0 second level-1
```

- **0**—Specifies an unencrypted password will follow.
- **8**—Specifies an encrypted password will follow.

- *password*—Enables authentication and specifies the key.
- (Optional) **level-1**—Enables authentication for Level 1 packets only.
- (Optional) **level-2**—Enables authentication for Level 2 packets only.

If no password is configured with the **key** command, no key authentication is performed. Key authentication can apply to clear text or MD5 authentication. See Step 4 to set the mode. Only one authentication key is applied to IS-IS at one time. If you configure a second key, the first is overridden. If you do not specify Level 1 or Level 2, the password applies to both levels.

Step 6 Configure the authentication password for an interface:

isis password *password* **[level-1 | level-2]**

Example:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# isis password analyst level-1
```

- *password*—Authentication password you assign to an interface.
- (Optional) **level-1**—Configures the authentication password for Level 1 independently. For Level 1 routing, the ASA acts as a station router only.
- (Optional) **level-2**—Configures the authentication password for Level 2 independently. For level 2 routing, the ASA acts as an area router only.

This command lets you prevent unauthorized routers from forming adjacencies with this ASA and thus protects the network from intruders. The password is exchanged as plain text and thus provides limited security. You can assign different passwords for different routing levels using the **level-1** and **level-2** keywords.

Examples

The following example shows an IS-IS instance with MD5 authentication performed on Level 1 packets and to send any key belonging to the key chain named site1:

```
ciscoasa(config)# router isis
ciscoasa(config-router)# net 49.0000.0101.0101.0101.00
ciscoasa(config-router)# is-type level-1
ciscoasa(config-router)# authentication send-only level-1
ciscoasa(config-router)# authentication mode md5 level-1
ciscoasa(config-router)# authentication key 0 site1 level-1
```

Configure IS-IS LSP

An IS generates LSPs to advertise its neighbors and the destinations that are directly connected to IS-IS. See IS-IS PDU Types, on page 1000 for more detailed information on LSPs.

Use the following commands to configure LSPs so that you have a faster convergence configuration.

Before you begin

In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the **changeto context** *name* command.

Procedure

Step 1 Enter router configuration mode:

router isis

Example:

```
ciscoasa(config)# router isis
ciscoasa(config-router)#
```

Step 2 Configure the ASA to ignore IS-IS LSPs that are received with internal checksum errors rather than purging the LSPs:

ignore-lsp-errors

Example:

```
ciscoas(config-router)# ignore-lsp-errors
```

IS-IS requires that an LSP with an incorrect data link checksum be purged by the receiver, which causes the initiator of the packet to regenerate it. If a network has a link that causes data corruption while still delivering LSPs with correct data link checksums, a continuos cycle of purging and regenerating large numbers of packets can occur, which can render the network nonfunctional. Use this command to ignore the LSPs rather than purge them. The default is enabled.

Step 3 Configure IS-IS to advertise only prefixes that belong to passive interfaces:

advertise passive-only

This command excludes IP prefixes of connected networks from LSP advertisements and thus reduces IS-IS convergence time, because fewer prefixes are advertised in the router non-pseudonode LSP.

Step 4 Configure IS-IS LSPs to be full:

fast-flood *lsp-number*

Example:

```
ciscoasa(config-router)# fast-flood 7
```

(Optional) *lsp-number*—The number of LSPs to be flooded before starting SPF.

This command sends a specified number of LSPs from the ASA. The LSPs invoke SPF before running SPF. Speeding up the LSP flooding process improves overall convergence time. The range is 1 to 15. The default is 5.

Note
We recommend that you enable fast flooding of LSPs before the router runs the SPF computation.

Step 5 Configure the MTU size of IS-IS LSPs:

lsp-mtu *bytes*

Example:

```
ciscoasa(config-router)# lsp-mtu 1300
```

bytes—The maximum packet size in bytes. The number of bytes must be less than or equal to the smallest MTU of any link in the network. The range is 128 to 4352.

Step 6 Set the maximum time that LSPs persist in the ASA's database without being refreshed:

max-lsp-lifetime *seconds*

Example:

```
ciscoasa(config-router)# max-lsp-lifetime 2400
```

seconds—The lifetime of the LSP in seconds. The range is 1 to 65,535. The default is 1200.

If the lifetime is exceeded before a refresh LSP arrives, the LSP is dropped from the database.

Step 7 Customize IS-IS throttling of SPF calculations:

spf-interval [level-1 | level-2] *spf-max-wait [spf-intial-wait spf-second wait]*

Example:

```
ciscoasa(config-router)#  spf-interval level-1 5 10 20
```

- (Optional) **level-1**—Apply intervals to Level 1 areas only.
- (Optional) **level-2**—Apply intervals to Level 2 areas only.
- *spf-max-wait*— Indicates the maximum interval between two consecutive SPF calculations. The range is 1 to 120 seconds. The default is 10 seconds.
- (Optional) *spf-initial-wait*— Indicates the initial wait time after a topology change before the first SPF calculation. The range is 1 to 120,000 milliseconds. The default is 5500 milliseconds (5.5 seconds).

  Each subsequent wait interval is twice as long as the previous one until the wait interval reaches the SPF maximum wait interval specified.
- (Optional) *spf-second-wait*—Indicates the interval between the first and second SPF calculation. The range is 1 to 120,000 milliseconds. The default is 5500 milliseconds (5.5 seconds).

SPF calculations are performed only when the topology changes. This command controls how often the software performs the SPF calculation.

Note
The SPF calculation is processor-intensive. Therefore, it may be useful to limit how often this is done, especially when the area is large and the topology changes often. Increasing the SPF interval reduces the processor load of the ASA, but potentially slows down the rate of convergence.

Step 8 Customize IS-IS throttling of LSP generation:

lsp-gen-interval [level-1 | level-2] *lsp-max-wait [lsp-intial-wait lsp-second wait]*

Example:

```
ciscoasa(config-router)#  lsp-gen-interval level-1 2 50 100
```

- (Optional) **level-1**—Apply intervals to Level 1 areas only.
- (Optional) **level-2**—Apply intervals to Level 2 areas only.
- *lsp-max-wait*— Indicates the maximum interval between two consecutive occurrences of an LSP being generated. The range is 1 to 120 seconds. The default is 5 seconds.
- (Optional) *lsp-initial-wait*— Indicates the initial wait time before generating the first LSP. The range is 1 to 120,000 milliseconds. The default is 50 milliseconds.

  Each subsequent wait interval is twice as long as the previous one until the wait interval reaches the LSP maximum wait interval specified.
- (Optional) *lsp-second-wait*—Indicates the interval between the first and second LSP generation. The range is 1 to 120,000 milliseconds. The default is 5000 milliseconds (5 seconds).

  This command controls the delay between LSPs being generated.

Step 9 Set the LSP refresh interval:

lsp-refresh-interval *seconds*

Example:

```
ciscoasa(config-router)# lsp-refresh-interval 1080
```

(Optional) *seconds*— The interval at which LSPs are refreshed. The range is 1 to 65535 seconds. The default value is 900 seconds (15 minutes).

The refresh interval determines the rate at which the software periodically transmits in LSPs the route topology information that it originates. This is done to keep the database information from becoming too old.

Note
LSPs must be periodically refreshed before their lifetimes expire. The value set for the **lsp-refresh-interval** command should be less than the value set for the **max-lsp-lifetime** command; otherwise, LSPs will time out before they are refreshed. If you set the LSP lifetime too low compared to the LSP refresh interval, the software reduces the LSP refresh interval to prevent the LSPs from timing out.

Step 10 Customize IS-IS throttling of PRCs:

prc-interval *prc-max-wait [prc-intial-wait prc-second wait]*

Example:

```
ciscoasa(config-router)#  prc-interval 5 10 20
```

- *prc-max-wait*— Indicates the maximum interval between two consecutive PRC calculations. The range is 1 to 120 seconds. The default is 5 seconds.

- (Optional) *prc-initial-wait*— Indicates the initial PRC wait time after a topology change. The range is 1 to 120,000 milliseconds. The default is 2000 milliseconds.

  Each subsequent wait interval is twice as long as the previous one until the wait interval reaches the PRC maximum wait interval specified.

- (Optional) *prc-second-wait*—Indicates the interval between the first and second PRC calculation. The range is 1 to 120,000 milliseconds. The default is 5000 milliseconds (5 seconds).

  PRC is the software process of calculating routes without performing an SPF calculation. This is possible when the topology of the routing system itself has not changed, but a change is detected in the information announced by a particular IS or when it is necessary to attempt to reinstall such routes in the RIB.

Step 11 Configure which routes are suppressed when the PDU becomes full:

lsp-full suppress {external [interlevel] | interlevel [external] | none}

Example:

```
ciscoasa(config-router)# lsp-full suppress interlevel external
```

- **external**—Suppresses any redistributed routes on this ASA.
- **interlevel**—Suppresses any routes coming from the other level. For example, if the Level 2 LSP becomes full, routes from Level 1 are suppressed.
- **none**—Suppresses no routes.

In networks where there is no limit placed on the number of redistributed routes into IS-IS (that is, the **redistribute maximum-prefix** command is not configured), it is possible that the LSP will fill up and routes are dropped. Use the **lsp-full suppress** command to define in advance which routes are suppressed if the LSP gets full.

Configure IS-IS Summary Addresses

Multiple groups of addresses can be summarized for a given level. Routes learned from other routing protocols can also be summarized. The metric used to advertise the summary is the smallest metric of all the more specific routes. This helps to reduce the size of the routing table.

You need to manually define summary addresses if you want to create summary addresses that do not occur at a network number boundary or if you want to use summary addresses on an ASA with automatic route summarization disabled.

Procedure

Step 1 Enter router configuration mode:

router isis

Example:

```
ciscoasa(config)# router isis
ciscoasa(config-router)#
```

Step 2 Create aggregate addresses for IS-IS:

summary-address *address mask* **[level-1 | level-1-2 | level-2] tag** *tag-number* **metric** *metric-value*

Example:

```
ciscoasa(config-router)# summary-address 10.1.0.0 255.255.0.0 tag 100 metric 110
```

- *address*—Summary address designated for a range of IP addresses.
- *mask*—IP subnet mask used for the summary route.
- (Optional) **level-1**—Only routes redistributed into Level 1 are summarized with the configured address and mask value.
- (Optional) **level-1-2**—Summary routes are applied when redistributing routes into Level 1 and Level 2 and when Level 2 IS-IS advertises Level 1 routes as reachable in its area.
- (Optional) **level-2**—Routes learned by Level 1 routing are summarized into the Level 2 backbone with the configured address and mask value. Redistributed routes into Level 2 IS-IS are summarized also.
- (Optional) **tag** *tag-number*—Specifies the number used to tag the summary route. The range is 1 to 4294967295.
- (Optional) **metric** *metric-value* —Specifies the metric value applied to the summary route. The **metric** keyword is assigned to the link and used to calculate the path cost via the links to destinations. You can configure this metric for Level 1 or Level 2 routing only. The range is 1 to 4294967295. The default value is 10.

Enter the **show clns interface** command to verify metric values for interfaces, See Monitoring IS-IS, on page 1036 for more information.

Configure IS-IS Passive Interfaces

You can disable IS-IS hello packets and routing updates on interfaces while still including the interface addresses in the topology database. These interfaces will not form IS-IS neighbor adjacencies

If you have an interface that you do not want to participate in IS-IS routing, but that is attached to a network that you want advertised, configure the passive interfaces (using the **passive-interface** command) to prevent that interface from using IS-IS. Additionally, you can specify the version of IS-IS that is used by the ASA for updates. Passive routing assists in controlling the advertisement of IS-IS routing information and disables the sending and receiving of IS-IS routing updates on an interface.

Procedure

Step 1 Enter router configuration mode:

router isis

Example:

```
ciscoasa(config)# router isis
ciscoasa(config-router)#
```

Step 2 Configure a passive interface on the ASA:

passive-interface *interface-name*

Example:

```
ciscoasa(config-router)# passive-interface inside
```

- **default**—Suppress routing updates on all interfaces.
- **management**—Suppress updates on Management 0/1 interface.
- **management2**—Suppress updates on Management 0/2 interface.
- **inside**—Suppress updates on the inside interface.

This command configures interfaces NOT to form IS-IS neighbor adjacencies yet to include the interface addresses in the IS-IS database.

Step 3 Configure the ASA to advertise passive interfaces:

advertise passive-only

Example:

```
ciscoasa(config-router)# advertise passive-only
```

This command configures IS-IS to advertise only prefixes that belong to passive interfaces. It excludes IP prefixes of connected networks from LSP advertisements, which reduces IS-IS convergence time.

Configure IS-IS Interfaces

This procedure describes how to modify individual ASA interfaces for IS-IS routing. You can modify the following:

- General settings such as enabling IS-IS, enabling IS-IS shutdown protocol, priorities, tags, and adjacency filters on an interface.
- Authentication key and mode—See Enable IS-IS Authentication, on page 1010 for the procedures for configuring authentication on interfaces.
- Hello padding values—See Configure IS-IS Interface Hello Padding, on page 1023 for the procedures for configuring hello padding on interfaces.
- LSP settings
- The interface delay metric used in IS-IS metric calculations.

Before you begin

Before the IS-IS routing process is useful, you must assign a NET and some interfaces must have IS-IS enabled. You can configure only one process to perform Level 2 (inter-area) routing. If Level 2 routing is configured on any process, all additional processes are automatically configured as Level 1. You can configure this process to perform intra-area (Level 1) routing at the same time. An interface cannot be part of more than one area, except in the case where the associated routing process is performing both Level 1 and Level 2 routing. See Enable IS-IS Routing Globally, on page 1006 for the procedure.

Procedure

Step 1 Enter interface configuration mode:

interface *interface_id*

Example:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# isis
```

Step 2 Filter the establishment of IS-IS adjacencies:

isis adjacency-filter *name* **[match-all]**

Example:

```
ciscoasa(config-if)# isis adjacency-filter ourfriends match-all
```

- *name*—The name of the filter set or expression to apply.
- (Optional) **match-all**—All NSAP addresses must match the filter to accept the adjacency. If not specified (the default), only one address needs to match the filter for the adjacency to be accepted.

Filtering is performed by building NSAP addresses out of incoming IS-IS hello packets by combining each area address in the hello with the system ID. Each of these NSAP addresses is then passed through the filter. If any one NSAP matches, the filter is considered passed, unless the **match-all** keyword is specified, in which case all addresses must pass. The functionality of the **match-all** keyword is useful in performing negative tests, such as accepting an adjacency only if a particular address is not present.

Step 3 Advertise IS-IS prefixes of connected networks in LSP advertisements on an IS-IS interface:

isis advertise prefix

Example:

```
ciscoasa(config-if)# isis advertise prefix
```

By default, this command is enabled. Thus, the connected routes are distributed even though they are not intended to get distributed. To stop unnecessary redistribution of connected routes and to improve IS-IS convergence time, use the **no isis advertise prefix** command. This excludes IP prefixes of connected network from LSP advertisements and reduces IS-IS convergence time.

Note

Configuring the **no** form of this command per IS-IS interface is a small-scale solution to reduce IS-IS convergence time because fewer prefixes are advertised in the router non-pseudonode LSP. An alternative to the **isis advertise prefix** command is the **advertise passive-only** command, which is a scalable solution because it is configured per IS-IS instance.

Step 4 Enable IPv6 on an IS-IS interface:

ipv6 router isis

Example:

```
ciscoasa(config-if)# ipv6 router isis
```

Step 5 Configure the time-delay between successive IS-IS LSP transmissions per interface:

isis lsp-interval *milliseconds*

Example:

```
ciscoasa(config-if)# isis lsp-interval 100
```

milliseconds—The time delay between successive LSPs. The range is 1 to 4294967298. The default is 33 milliseconds.

In topologies with a large number of IS-IS neighbors and interfaces, an ASA may have difficulty with the CPU load imposed by LSP transmission and reception. This command reduces the LSP transmission rate (and by implication the reception rate of other systems).

Step 6 Configure the value of an IS-IS metric:

isis metric {*metric-value* | **maximum**} [**level-1 | level-2**]

Example:

```
ciscoasa(config-if)# isis metric 15 level-1
```

- *metric-value*—Metric assigned to the link and used to calculate the cost from each other router via the links in the network to other destinations. You can configure this metric for Level 1 or Level 2 routing. The range is from 1 to 63. The default value is 10.
- **maximum**—Excludes a link or adjacency from the SPF calculation.
- (Optional) **level-1**—Specifies that this metric should be used only in the SPF calculation for Level 1 (intra-area) routing. If no optional keyword is specified, the metric is enabled on routing Level 1 and Level 2.
- (Optional) **level-2**—Specifies that this metric should be used only in the SPF calculation for Level 2 (inter-area) routing. If no optional keyword is specified, the metric is enabled on routing Level 1 and Level 2.

Step 7 Configure the priority of designated ASAs on the interface:

isis priority *number-value* **[level-1 | level-2]**

Example:

```
ciscoasa(config-if)# isis priority 80 level-1
```

- *number-value*—Sets the priority of an ASA. The range is 0 to 127. The default is 64.
- (Optional) **level-1**—Sets the priority for Level 1 independently.
- (Optional) **level-2**—Sets the priority for Level 2 independently.

The priority is used to determine which ASA on a LAN will be the designated router or DIS. The priorities are advertised in the hello packets. The ASA with the highest priority becomes the DIS.

Note
In IS-IS there is no backup designated router. Setting the priority to 0 lowers the chance of this system becoming the DIS, but does not prevent it. If a router with a higher priority comes on line, it takes over the role from the current DIS. In the case of equal priorities, the highest MAC address breaks the tie.

Step 8 Disable IS-IS protocol so that it cannot form adjacencies on a specified interface and place the IP address of the interface into the LSP that is generated by the ASA:

isis protocol shutdown

Example:

```
ciscoasa(config-if)# isis protocol shutdown
```

This command lets you disable the IS-IS protocol for a specified interface without removing the configuration parameters. The IS-IS protocol does not form any adjacencies for the interface for which this command has been configured, and the IP address of the interface is put into the LSP that is generated by the router. Use the **protocol shutdown** command if you do not want IS-IS to form any adjacency on any interface and to clear the IS-IS LSP database. See Enable IS-IS Routing Globally, on page 1006 for the procedure.

Step 9 Configure the amount of time between retransmission of each IS-IS LSP:

isis retransmit-interval *seconds*

Example:

```
ciscoasa(config-if)#  isis retransmit-interval 60
```

(Optional) *seconds*— Time between retransmission of each LSP. The number should be greater than the expected round-trip delay between any two routers on the attached network. The range is 0 to 65535. The default is 5 seconds.

Make sure the *seconds* argument is conservative, otherwise needless retransmission results. This command has no effect on LAN (multi-point) interfaces.

Step 10 Configure the amount of time between retransmissions of each IS-IS LSP:

isis retransmit-throttle-interval *milliseconds*

Example:

```
ciscoasa(config-if)#  isis retransmit-throttle-interval 300
```

(Optional) *milliseconds*— Minimum delay between LSP retransmissions on the interface. The range is 0 to 65535.

This command can be useful in very large networks with many LSPs and many interfaces as a way of controlling LSP retransmission traffic. This command controls the rate at which LSPs can be resent on the interface.

This command is distinct from the rate at which LSPs are sent on the interface (controlled by the **isis lsp-interval** command) and the period between retransmissions of a single LSP (controlled by the **isis retransmit-interval** command). You can use these commands in combination to control the offered load of routing traffic from one ASA to its neighbors.

Step 11 Set a tag on the IP address configured for an interface when the IP prefix is put into an IS-IS LSP:

isis tag *tag-number*

Example:

```
ciscoasa(config-if)# isis tag 100
```

tag-number—The number that serves as a tag on an IS-IS route. The range is 1 to 4294967295.

No action occurs on a tagged route until the tag is used, for example, to redistribute routes or summarize routes. Configuring this command triggers the ASA to generate new LSPs because the tag is a new piece of information in the packet.

Examples

In this example, two interfaces are tagged with different tag values. By default, these two IP addresses would have been put into the IS-IS Level 1 and Level 2 database. However, if you use the **redistribute** command with a route map to match tag 110, only IP address 172.16. 0.0 is put into the Level 2 database.

```
ciscoasa (config)# interface GigabitEthernet1/0
ciscoasa (config-if)# ip address 10.1.1.1 255.255.255.0
ciscoasa (config-if)# isis
ciscoasa (config-if)# isis tag 120
ciscoasa (config)# interface GigabitEthernet1/1
ciscoasa (config-if)# ip address 172.16.0.0
ciscoasa (config-if)# isis
ciscoasa (config-if)# isis tag 110
ciscoasa (config-router)# route-map match-tag permit 10
ciscoasa (config-router)# match tag 110
ciscoasa (config)# router isis
ciscoasa (config-router)# net 49.0001.0001.0001.0001.00
ciscoasa (config-router)# redistribute isis ip level-1 into level-2 route-map match-tag
```

Configure IS-IS Interface Hello Padding

Hello packets are responsible for discovering and maintaining neighbors. You can configure the following hello padding parameters at the interface level. See Enable IS-IS Routing Globally, on page 1006 to enable/disable hello padding for the whole IS-IS.

Procedure

Step 1 Enter interface configuration mode:

interface *interface_id*

Example:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# isis
```

Step 2 Enter interface configuration mode to configure padding on IS-IS hello protocol data units (IIH PDUs) for all interfaces on the ASA:

isis hello padding

Example:

```
ciscoasa(config-if)# isis hello padding
```

Hellos are padded to the full MTU, which allows for early detection of errors that result from transmission problems with large frames or errors that result from mismatched MTUs on adjacent interfaces. IS-IS hello padding is enabled by default.

Note
You can disable hello padding to avoid wasting network bandwidth in case the MTU of both interfaces is the same or in case of translational bridging. While hello padding is disabled, the ASAs still send the first five IS-IS hellos padded to the full MTU size to maintain the benefits of discovering MTU mismatches.

Step 3 Specify the length of time between consecutive hello packets sent by IS-IS:

isis hello-interval {*seconds* | **minimal**} **[level-1 | level-2]**

Example:

```
ciscoasa(config-if)# isis hello-interval 5 level-1
```

- *seconds*—The length of time between hello packets. By default, a value three times the hello interval seconds is advertised as the hold time in the hello packets sent. You can change the multiplier of 3 by configuring the **isis hello-multiplier** command. With smaller hello intervals, topological changes are detected faster, but there is more routing traffic. The range is 0 to 65535. The default is 10.
- **minimal**—Causes the system to compute the hello interval based on the hello multiplier (specified by the **isis hello-multiplier** command) so that the resulting hold time is 1 second.
- (Optional) **level-1**—Configures the hello interval for Level 1 independently. Use this on X.25, Switched Multimegabit Data Service (SMDS), and Frame Relay multi-access networks.
- (Optional) **level-2**—Configures the hello interval for Level 2 independently. Use this on X.25, SMDS, and Frame Relay multi-access networks.

Note

Although a slower hello interval saves bandwidth and CPU usage, there are some situations when a faster hello interval is preferred, for example, a large configuration that uses Traffic Engineering (TE) tunnels. If the TE tunnel uses IS-IS as the Interior Gateway Protocol (IGP), and the IP routing process is restarted at the router at the ingress point of the network (head-end), then all the TE tunnels get resignaled with the default hello interval. A faster hello interval prevents this resignaling. To configure a faster hello interval, you need to increase the IS-IS hello interval manually using the **isis hello-multiplier** command.

Step 4 Specify the number of IS-IS hello packets a neighbor must miss before the ASA declares the adjacency as down:

isis hello-multiplier *multiplier* **[level-1 | level-2]**

Example:

```
ciscoasa(config-if)# isis hello-multiplier 10 level-1
```

- *multipler*—The advertised hold time in IS-IS hello packets is set to the hello multiplier times the hello interval. Neighbors declare an adjacency to this ASA down after not having received any IS-IS hello packets during the advertised hold time. You can set the hold time (and thus the hello multiplier and the hello interval) on a per-interface basis, and it can be different between different routers in one area. The range is 3 to 1000. The default is 3.
- (Optional) **level-1**—Configures the hello multiplier independently for Level 1 adjacencies.
- (Optional) **level-2**—Configures the hello multiplier independently for Level 2 adjacencies.

Use this command in circumstances where hello packets are lost frequently and IS-IS adjacencies are failing unnecessarily.

Note
Using a smaller hello multiplier will give fast convergence, but can result in more routing instability. Change the hello multiplier to a larger value to help network stability when needed. Never configure a hello multiplier lower than the default value of 3.

Step 5 Configure the type of adjacency used for the IS-IS:

isis circuit-type [level-1 | level-1-2 | level-2-only]

Example:

```
ciscoasa(config-if)# isis circuit-type level-2-only
```

- (Optional) **level-1**— Configures an ASA for Level 1 adjacency only.
- (Optional) **level-1-2**—Configures an ASA for Level 1 and Level 2 adjacency.
- (Optional) **level-2**—Configures an ASA for Level 2 adjacency only.

You do not normally need to configure this command. The correct way is to configure the level on an ASA. See Enable IS-IS Routing Globally, on page 1006 for the procedure. You should configure some interfaces as Level 2 only on ASAs that are between areas (Level 1-2 routers). This saves bandwidth by sending out unused Level 1 hello packets.

Step 6 Configure the interval at which periodic CSNP packets are sent on broadcast interfaces:

isis csnp-interval *seconds* **[level-1 | level-1-2 | level-2]**

Example:

```
ciscoasa(config-if)#  isis csnp-interval 30 level-1
```

- *seconds*— Interval of time between transmission of CSNPs on multi-access networks. This interval only applies for the designated ASA. The range is 0 to 65,535. The default is 10 seconds.
- (Optional) **level-1**—Configures the interval of time between transmission of CSNPs for Level 1 independently.
- (Optional) **level-2**—Configures the interval of time between transmission of CSNPs for Level 2 independently.

It is unlikely that you will need to change the default value for this command.

This command applies only for the DR for a specified interface. Only DRs send CSNP packets to maintain database synchronization. You can configure the CSNP interval independently for Level 1 and Level 2.

Configure IS-IS IPv4 Address Family

Routers are allowed to redistribute external prefixes or routes that are learned form any other routing protocol, static configuration, or connected interface. The redistributed routes are allowed in either a Level 1 router or a Level 2 router.

You can set up adjacency, Shortest Path First (SPF), and you can define conditions for redistributing routes from another routing domain into ISIS (redistribution) for IPv4 addresses.

Before you begin

Before you can enable IS-IS route authentication, you must enable IS-IS and set up an area. See Enable IS-IS Routing Globally, on page 1006 for the procedure.

Procedure

Step 1 Enter router configuration mode to configure an IPv4 address familiy:

router isis

Example:

```
ciscoasa(config)# router isis
cisco(config-router)#
```

Step 2 Perform an adjacency check to check for IS-IS protocol support:

adjacency-check

Example:

```
cisco(config-router)# adjacency-check
```

Step 3 Define the administrative distance assigned to routes discovered by the IS-IS protocol:

distance *weight*

weight—Administrative distance assigned to IS-IS routes. The range is 1 to 255. The default is 115.

Example:

```
ciscoasa(config-router)# distance 20
```

This command configures the distances applied to IS-IS routes when they are inserted in the RIB and influence the likelihood of these routes being preferred over routes to the same destination addresses discovered by other protocols.

Note
In general, the higher the value of the administrative distance, the lower the trust rating. An administrative distance of 255 means that the routing information source cannot be trusted at all and should be ignored. Weight values are subjective; no quantitative method exists for choosing weight values.

Step 4 Configure multi-path load sharing for IS-IS:

maximum-paths *number-of-paths*

Example:

```
ciscoasa(config-router)# maximum-paths 8
```

number-of-paths—Number of routes to install in the routing table. The range is 1 to 8. The default is 1.

The **maximum-path** command is used to configure IS-IS multi-load sharing when ECMP is configured in the ASA.

Step 5 Generate a default route into an IS-IS routing domain:

default-information originate [**route-map** *map-name*]

Example:

```
ciscoasa(config-router)# default-information originate route-map RMAP
```

(Optional) **route-map** *map-name*—The routing process generates the default route if the route map is satisfied.

If an ASA configured with this command has a route to 0.0.0.0 in the routing table, IS-IS will originate an advertisement for 0.0.0.0 in its LSPs. Without a route map, the default is advertised only in Level 2 LSPs. For Level 1 routing, there is another mechanism to find the default route, which is to look for the closest Level 1 or Level 2 router. The closest Level 1 or Level 2 router can be found by looking at the ATT in Level 1 LSPs. With a **match ip address** *standard-access-list* command, you can specify one or more IP routes that must exist before the ASA will advertise 0/0.

Step 6 Set the IS-IS metric globally for Level 1 and Level 2:

metric *default-value* **[level-1 | level-2]**

Example:

```
ciscoasa(config-router)# metric 55 level-1
ciscoasa(config-router)# metric 45 level-2
```

- *default-value*—The metric value to be assigned to the link and used to calculate the path cost via the links to destinations. The range is 1 to 63. The default is 10.
- (Optional) **level-1**— Sets Level 1 IPv4 or IPv6 metric.
- (Optional) **level-2**— Sets Level 2 IPv4 or IPv6 metric.

Step 7 Specify the metric style and which levels to apply it to:

metric-style [narrow | transition | wide] [level-1 | level-2 | level-1-2]

Example:

```
ciscoasa(config-router)# metric-style wide level-1
```

- **narrow**—Instructs the ASA to use the old style of TLVs with the narrow metric.
- **transition**— Instructs the ASA to accept both old- and new-style TLVs during transition.
- **wide**—Instructs the ASA to use the new style of TLVs to carry the wider metric.
- (Optional) **level-1**— Sets Level 1 IPv4 or IPv6 metric.
- (Optional) **level-2**— Sets Level 2 IPv4 or IPv6 metric.
- (Optional) **level-1-2**—Sets Level 1 and Level 2 IPv4 or IPv6 metric.

Step 8 Specify constraints for when a Level 1-Level 2 router should set its attached bit:

set-attached-bit route-map *map-tag*

Example:

```
ciscoasa(config-router)# set-attached-bit route-map check-for-L2_backbone_connectivity
```

route-map *map-tag*—Identifier of a configured route map. If the specified route map is matched, the router continues to set its attached bit. This command is disabled by default.

In the current IS-IS implementation, as specified in ISO 10589, Level 1-Level 2 routers set their Level 1 LSP attached bit when they see other areas in their own domain or see other domains. However, in some network topologies, adjacent Level 1-Level 2 routers in different areas may lose connectivity to the Level 2 backbone. Level 1 routers may then send traffic destined outside of the area or domain to Level 1-Level 2 routers that may not have such connectivity.

This command allows more control over the attached bit setting for Level 1-Level 2 routers. The route map can specify one or more CLNS routes. If at least one of the match address route map clauses matches a route in the Level 2 CLNS routing table, and if all other requirements for setting the attached bit are met, the Level 1-Level 2 router continues to set the attached bit in its Level 1 LSP. If the requirements are not met or no match address route map clauses match a route in the Level 2 CLNS routing table, the attached bit is not set.

Step 9 Configure the ASA to signal other routers not to use it as an intermediate hop in their SPF calculations:

set-overload-bit [on-startup {*seconds* | **wait-for bgp**}] [**suppress [[interlevel] [external**]]]

Example:

```
ciscoasa(config-router)# set-overload-bit on-startup wait-for-bgp suppress interlevel
external
```

- (Optional) **on-startup**—Sets the overload bit at system startup. The overload bit remains set for the number of seconds configured or until BGP has converged, depending on the subsequent argument or keyword specified.
- (Optional) *seconds*—The number of seconds the overload bit is set at system startup and remains set. The range is 5 to 86400.
- (Optional) **wait-for-bgp**—When the **on-startup** keyword is configured, causes the overload bit to be set at system startup and remain set until BGP has converged.
- (Optional) **suppress**—Causes the type of prefix identified by the subsequent keyword or keywords to be suppressed.
- (Optional) **interlevel**—When the **suppress** keyword is configured, prevents the IP prefixes learned from another IS-IS level from being advertised.
- (Optional) **external**—When the **suppress** keyword is configured, prevents the IP prefixes learned from other protocols being advertised.

This command forces the ASA to set the overload bit (also known as the hippity bit) in its non-pseudonode LSPs. Normally, the setting of the overload bit is allowed only when an ASA runs into problems. For example, when an ASA is experiencing a memory shortage, it might be that the link-state database is not complete, which results in an incomplete or inaccurate routing table. By setting the overload bit in its LSPs, other routers can ignore the unreliable router in their SPF calculations until the router has recovered from its problems. The result is that no paths through this router are seen by other routes in the IS-IS area. However, IP and CLNS prefixes are directly connected to this router.

Step 10 Customize IS-IS throttling of PRCs:

prc-interval *prc-max-wait [prc-intial-wait prc-second wait]*

Example:

```
ciscoasa(config-router)#  prc-interval 5 10 20
```

- *prc-max-wait*— Indicates the maximum interval between two consecutive PRC calculations. The range is 1 to 120 seconds. The default is 5 seconds.
- (Optional) *prc-initial-wait*— Indicates the initial PRC wait time after a topology change. The range is 1 to 120,000 milliseconds. The default is 2000 milliseconds.

  Each subsequent wait interval is twice as long as the previous one until the wait interval reaches the PRC maximum wait interval specified.
- (Optional) *prc-second-wait*—Indicates the interval between the first and second PRC calculation. The range is 1 to 120,000 milliseconds. The default is 5000 milliseconds (5 seconds).

  PRC is the software process of calculating routes without performing an SPF calculation. This is possible when the topology of the routing system itself has not changed, but a change is detected in the information announced by a particular IS or when it is necessary to attempt to reinstall such routes in the RIB.

Step 11 Customize IS-IS throttling of SPF calculations:

spf-interval [level-1 | level-2] *spf-max-wait [spf-intial-wait spf-second wait]*

Example:

```
ciscoasa(config-router)#  spf-interval level-1 5 10 20
```

- (Optional) **level-1**—Apply intervals to Level 1 areas only.
- (Optional) **level-2**—Apply intervals to Level 2 areas only.
- *spf-max-wait*—Indicates the maximum interval between two consecutive SPF calculations. The range is 1 to 120 seconds. The default is 10 seconds.
- (Optional) *spf-initial-wait*—Indicates the initial wait time after a topology change before the first SPF calculation. The range is 1 to 120,000 milliseconds. The default is 5500 milliseconds (5.5 seconds).

  Each subsequent wait interval is twice as long as the previous one until the wait interval reaches the SPF maximum wait interval specified.
- (Optional) *spf-second-wait*—Indicates the interval between the first and second SPF calculation. The range is 1 to 120,000 milliseconds. The default is 5500 milliseconds (5.5 seconds).

SPF calculations are performed only when the topology changes. This command controls how often the software performs the SPF calculation.

Note
The SPF calculation is processor-intensive. Therefore, it may be useful to limit how often this is done, especially when the area is large and the topology changes often. Increasing the SPF interval reduces the processor load of the ASA, but potentially slows down the rate of convergence.

Step 12 Configure IS-IS to honor external metrics during SFP calculations:

use external-metrics

Step 13 Configure a BGP, connected, IS-IS, OSPF, or Static route redistribution:

redistribute bgp | connected | isis | ospf | static | level-1 | level-2 | level 1-2 metric-type internal | external metric *number*

Example:

```
ciscoasa(config-router)# redistribute bgp level-1 metric-type internal metric 6
```

Note
Redistribution of static routes with a route-map matching a metric is not supported.

metric *number*—Value for metric. The range is 1 to 4294967295.

Attached Bit Configuration

In the following example, the attached-bit will stay set when the router matches 49.00aa in the L2 CLNS routing table:

```
ciscoasa(config)# router isis
 ciscoasa(config-router)# clns filter-set L2_backbone_connectivity permit 49.00aa
 ciscoasa(config-router)# route-map check-for-L2_backbone_connectivity
 ciscoasa(config-router)# match clns address L2_backbone_connectivity
ciscoasa(config)# router isis
 ciscoasa(config-router)#set-attached-bit route-map check-for-L2_backbone_connectivity
 ciscoasa(config-router)# end
 ciscoasa# show clns route 49.00aa

Known via "isis", distance 110, metric 30, Dynamic Entry
Routing Descriptor Blocks:
 via tr2, Serial0
   isis, route metric is 30, route version is 58
```

Configure IS-IS IPv6 Address Family

You can set up adjacency, SPF, and you can define conditions for redistributing routes from another routing domain into IS-IS (redistribution) for IPv6 addresses.

Before you begin

Before you can enable IS-IS route authentication, you must enable IS-IS and set up an area. See Enable IS-IS Routing Globally, on page 1006 for the procedure.

Procedure

Step 1 Enter router configuration mode:

router isis

Example:

```
cisco(config-router)#
```

Step 2 Specify the metric style as wide:

metric-style wide [transition] [level-1 | level-2 | level-1-2]

Example:

```
ciscoas(config)# router isis
ciscoasa(config-router)# metric-style wide level-1
```

- (Optional) **transition**— Instructs the router to accept both old- and new-style TLVs.
- (Optional) **level-1**— Sets Level 1 IPv4 or IPv6 metric.
- (Optional) **level-2**— Sets Level 2 IPv4 or IPv6 metric.
- (Optional) **level-1-2**—Sets Level 1 and Level 2 IPv4 or IPv6 metric.

We recommend you use the **metric** command when you need to change the default metric for all IS-IS interfaces. this prevents user errors, such as unintentionally removing a set metric from an interface without

configuring a new value and unintentionally allowing the interface to revert to the default metric of 10, thereby becoming a highly preferred interface in the network.

Step 3 Enter address family configuration mode to configure IS-IS routing sessions that use standard IPv4 or IPv6 address prefixes:

address-family ipv6 [unicast]

Example:

```
ciscoasa(config-router)# address-family ipv6 unicast
cisco(config-router-af)#
```

Step 4 Perform an adjacency check to check for IS-IS protocol support:

adjacency-check

Example:

```
cisco(config-router-af)# adjacency-check
```

Step 5 Configure multi-path load sharing for IS-IS:

maximum-paths *number-of-paths*

Example:

```
ciscoasa(config-router-af)# maximum-paths 8
```

number-of-paths—The number of routes to install in the routing table. The range is 1 to 8. The default is 1.

The **maximum-path** command is used to configure IS-IS multi-load sharing when ECMP is configured in the ASA.

Step 6 Define the administrative distance assigned to routes discovered by the IS-IS protocol:

distance *weight*

weight—The administrative distance assigned to IS-IS routes. The range is 1 to 255. The default is 115.

Example:

```
ciscoasa(config-router-af)# distance 20
```

This command configures the distances applied to IS-IS routes when they are inserted in the RIB and influence the likelihood of these routes being preferred over routes to the same destination addresses discovered by other protocols.

Note
In general, the higher the value of the administrative distance, the lower the trust rating. An administrative distance of 255 means that the routing information source cannot be trusted at all and should be ignored. Weight values are subjective; no quantitative method exists for choosing weight values.

Step 7 Generate a default route into an IS-IS routing domain:

default-information originate [**route-map** *map-name*]

Example:

```
ciscoasa(config-router-af)# default-information originate route-map TEST7
```

(Optional) **route-map** *map-name*—The routing process generates the default route if the route map is satisfied.

If an ASA configured with this command has a route to 0.0.0.0 in the routing table, IS-IS will originate an advertisement for 0.0.0.0 in its LSPs. Without a route map, the default is advertised only in Level 2 LSPs. For Level 1 routing, there is another mechanism to find the default route, which is to look for the closest Level 1 or Level 2 router. The closest Level 1 or Level 2 router can be found by looking at the ATT in Level 1 LSPs. With a **match ip address** *standard-access-list* command, you can specify one or more IP routes that must exist before the ASA will advertise 0/0.

Step 8 Configure the ASA to signal other routers not to use it as an intermediate hop in their SPF calculations:

set-overload-bit [on-startup {*seconds* | **wait-for bgp**}] [**suppress [[interlevel] [external**]]]

Example:

```
ciscoasa(config-router-af)# set-overload-bit on-startup wait-for-bgp suppress interlevel
external
```

- (Optional) **on-startup**—Sets the overload bit at system startup. The overload bit remains set for the number of seconds configured or until BGP has converged, depending on the subsequent argument or keyword specified.
- (Optional) *seconds*—The number of seconds the overload bit is set at system startup and remains set. The range is 5 to 86400.
- (Optional) **wait-for-bgp**—When the **on-startup** keyword is configured, causes the overload bit to be set at system startup and remain set until BGP has converged.
- (Optional) **suppress**—Causes the type of prefix identified by the subsequent keyword or keywords to be suppressed.
- (Optional) **interlevel**—When the **suppress** keyword is configured, prevents the IP prefixes learned from another IS-IS level from being advertised.
- (Optional) **external**—When the **suppress** keyword is configured, prevents the IP prefixes learned from other protocols being advertised.

This command forces the ASA to set the overload bit (also known as the hippity bit) in its non-pseudonode LSPs. Normally, the setting of the overload bit is allowed only when an ASA runs into problems. For example, when an ASA is experiencing a memory shortage, it might be that the link-state database is not complete, which results in an incomplete or inaccurate routing table. By setting the overload bit in its LSPs, other routers can ignore the unreliable router in their SPF calculations until the router has recovered from its problems. The result is that no paths through this router are seen by other routes in the IS-IS area. However, IP and CLNS prefixes are directly connected to this router.

Step 9 Customize IS-IS throttling of PRCs:

prc-interval *prc-max-wait [prc-intial-wait prc-second wait]*

Example:

```
ciscoasa(config-router-af)#  prc-interval 5 10 20
```

- *prc-max-wait*— Indicates the maximum interval between two consecutive PRC calculations. The range is 1 to 120 seconds. The default is 5 seconds.
- (Optional) *prc-initial-wait*— Indicates the initial PRC wait time after a topology change. The range is 1 to 120,000 milliseconds. The default is 2000 milliseconds.

  Each subsequent wait interval is twice as long as the previous one until the wait interval reaches the PRC maximum wait interval specified.
- (Optional) *prc-second-wait*—Indicates the interval between the first and second PRC calculation. The range is 1 to 120,000 milliseconds. The default is 5000 milliseconds (5 seconds).

  PRC is the software process of calculating routes without performing an SPF calculation. This is possible when the topology of the routing system itself has not changed, but a change is detected in the information announced by a particular IS or when it is necessary to attempt to reinstall such routes in the RIB.

Step 10 Customize IS-IS throttling of SPF calculations:

spf-interval [level-1 | level-2] *spf-max-wait [spf-intial-wait spf-second wait]*

Example:

```
ciscoasa(config-router-af)#  spf-interval level-1 5 10 20
```

- (Optional) **level-1**—Apply intervals to Level 1 areas only.
- (Optional) **level-2**—Apply intervals to Level 2 areas only.
- *spf-max-wait*—Indicates the maximum interval between two consecutive SPF calculations. The range is 1 to 120 seconds. The default is 10 seconds.
- (Optional) *spf-initial-wait*—Indicates the initial wait time after a topology change before the first SPF calculation. The range is 1 to 120,000 milliseconds. The default is 5500 milliseconds (5.5 seconds).

  Each subsequent wait interval is twice as long as the previous one until the wait interval reaches the SPF maximum wait interval specified.
- (Optional) *spf-second-wait*—Indicates the interval between the first and second SPF calculation. The range is 1 to 120,000 milliseconds. The default is 5500 milliseconds (5.5 seconds).

SPF calculations are performed only when the topology changes. This command controls how often the software performs the SPF calculation.

Note
The SPF calculation is processor-intensive. Therefore, it may be useful to limit how often this is done, especially when the area is large and the topology changes often. Increasing the SPF interval reduces the processor load of the ASA, but potentially slows down the rate of convergence.

Step 11 Configure a BGP, connected, IS-IS, OSPF, or Static route redistribution:

redistribute bgp | connected | isis | ospf | static | level-1 | level-2 | level 1-2 metric-type internal | external metric *number*

Example:

```
ciscoasa(config-router-af)# redistribute static level-1 metric-type internal metric 6
```

metric *number*—The value for metric. The range is 1 to 4294967295.

Step 12 Redistribute IS-IS routes specifically from Level 1 into Level 2 or from Level 2 into Level 1:

redistribute isis {level-1 | level-2} into {level-2 | level-1} [[distribute-list *list-number* | **[route-map** *map-tag***]]**

Example:

```
ciscoasa(config-router-af)# redistribute isis level-1 into level-2
distribute-list 100
```

- **level-1 | level-2**—The level from which and to which you are redistributing IS-IS routes.
- **into**—The keyword that separates the level of routes being redistributed from the level into which you are redistributing routes.
- (Optional) **distribute-list** *list-number*—The number of a distribute list that controls the IS-IS redistribution. You can specify either a distribute list or a route map, but not both.
- (Optional) **route-map** *map-tag*—The name of a route map that controls the IS-IS redistribution. You can specify either a distribute list or a route map, but not both.

Note
You must specify the **metric-style wide** command for the **redistribute isis** command to work. See Step 1 of this procedure.

In IS-IS, all areas are stub areas, which means that no routing information is leaked from the backbone (Level 2) into areas (Level 1). Level 1-only routers use default routing to the closest Level 1-Level 2 router in their area. This command lets you redistribute Level 2 IP routes into Level 1 areas. This redistribution enables Level 1-only routers to pick the best path for an IP prefix to get out of the area. This is an IP-only feature, CLNS routing is still stub routing.

Note
For more control and stability you can configure a distribute list or route map to control which Level 2 IP routes can be redistributed into Level 1. This allows large IS-IS-IP networks to use area for better scalability.

Step 13 Create aggregate prefixes for IS-IS IPv6 routes:

summary-prefix *ipv6-prefix* **[level-1 | level-1-2 | level-2]**

Example:

```
cisco(config-router-af)# summary-prefix 2001::/96 level-1
```

- *ipv6 address*—The IPv6 prefix in the form X.X.X.X.::X/0-128.
- (Optional) **level-1**—Only routes redistributed into Level 1 are summarized with the configured address and mask value.
- (Optional) **level-1-2**—Summary routes are applied when redistributing routes into Level 1 and Leve2 IS-IS and when Level 2 IS-IS advertises Level 1 routes as reachable in it area.

- (Optional) **level-2**—Routes learned by Level 1 routing are summarized into the Level 2 backbone with the configured address and mask value. Redistributed routes into Level 2 IS-IS are summarized also.

Monitoring IS-IS

You can use the following commands to monitor the IS-IS routing process. For examples and descriptions of the command output, see the command reference.

Monitoring the IS-IS Database

Use the following commands to monitor the IS-IS database:

- **show isis database [level-1 | l1] [level-2 | l2] [detail]** —Displays the IS-IS link-state database for Level 1, Level 2, and the detailed contents of each LSP.
- **show isis database verbose** —Displays more information about the IS-IS database, such as sequence number, checksum, and holdtime for LSPs.

Monitoring IS-IS Mapping Table Entries

Use the following command to monitor IS-IS hostnames:

show isis hostname—Displays the router-name-to-system-ID mapping table entries for an IS-IS router.

Monitoring IS-IS IPv4

Use the following commands to monitor IS-IS IPv4:

- **show isis ip rib**—Displays the IPv4 address family-specific RIB for an IS-IS routing process.
- **show isis ip spf-log**—Displays the IPv4 address family-specific SPF logs for an IS-IS routing process.
- **show isis ip topology**—Displays the IPv4 address family-specific topology for an IS-IS routing process.
- **show isis ip redistribution** [**level-1** | **level-2**] [*network-prefix*]—Displays IS-IS learned and installed IPv6 routes.
- **show isis ip unicast**—Displays the IPv4 address family-specific RIB, SPF logs, and paths to ISes.

Monitoring IS-IS IPv6

Use the following commands to monitor IS-IS IPV6:

- **show isis ipv6 rib**—Displays the IPv6 address family-specific RIB for an IS-IS routing process.
- **show isis ipv6 spf-log**—Displays the IPv6 address family-specific SPF logs for an IS-IS routing process.
- **show isis ipv6 topology**—Displays the IPv6 address family-specific topology for an IS-IS routing process.
- **show isis ipv6 redistribution** [**level-1** | **level-2**] [*network-prefix*]—Displays IS-IS learned and installed IPv6 routes.

- **show isis ipv6 unicast**—Displays the IPv6 address family-specific RIB, SPF logs, and paths to ISes.

Monitoring IS-IS Logs

Use the following commands to monitor IS-IS logs:

- **show isis lsp-log**—Displays the Level 1 and Level 2 IS-IS LSP log of the interfaces that triggered the new LSP.
- **show isis spf-log**—Displays how often and why the ASA has run an SPF calculation.

Monitoring IS-IS Protocol

Use the following command to monitor IS-IS protocol:

show clns protocol —Displays the protocol information for each IS-IS routing process on the ASA.

Monitoring IS-IS Neighbors and Routes

Use the following commands to monitor IS-IS neighbors:

- **show isis topology** —Displays a list of all connected routers in all areas. This command verifies the presence and connectivity between all routers in all areas.
- **show isis neighbors [detail]** —Displays IS-IS adjacency information.
- **show clns neighbors** [*process-tag*] [*interface-name*] [**detail**]—Displays end system (ES), intermediate system (IS) and multi-topology IS-IS (M-ISIS) neighbors. This command displays the adjacency learned through multitopology IS-IS for IPv6.
- **show clns is-neighbors** *[interface-name]* **[detail]** —Displays IS-IS information for IS-IS device adjacencies.

Monitoring IS-IS RIB

Use the following commands to monitor IS-IS RIB:

- **show isis rib** [*ip-address* | *ip-address-mask*]—Displays paths for a specific route or for all routes under a major network that are stored in the RIB.
- **show isis rib redistribution** [**level-1** | **level-2**] [*network-prefix*]—Displays the prefixes in the local redistribution cache.
- **show route isis** Displays the current state of the routing table.

Monitoring IS-IS Traffic

Use the following command to monitor IS-IS traffic:

show clns traffic [since {bootup | show}] —Displays the CLNS traffic statistics that the ASA has seen.

Debugging IS-IS

Use the following commands to debug IS-IS:

debug isis [adj-packets | authentication | checksum-errors | ip | ipv6 | local-updates | [rptpcp;-errors | rob | snp-packets | spf-events | spf-statistics | spf-triggers | update-packets]—Debugs various aspects of the IS-IS routing protocol.

History for IS-IS

Table 39: Feature History for IS-IS

Feature Name	Platform Releases	Feature Information
IS-IS routing	9.6(1)	The ASA now supports the Intermediate System to Intermediate System (IS-IS) routing protocol. Support was added for routing data, performing authentication, and redistributing and monitoring routing information using the IS-IS routing protocol. We introduced the following commands: **advertise passive-only, area-password, authentication key, authentication mode, authentication send-only, clear, debug isis, distance, domain-password, fast-flood, hello padding, hostname dynamic, ignore-lsp-errors, isis adjacency-filter, isis advertise prefix, isis authentication key, isis authentication mode, isis authentication send-only, isis circuit-type, isis csnp-interval, isis hello-interval, isis hello-multiplier, isis hello padding, isis lsp-interval, isis metric, isis password, isis priority, isis protocol shutdown, isis retransmit-interval, isis retransmit-throttle-interval, isis tag, is-type, log-adjacency-changes, lsp-full suppress, lsp-gen-interval, lsp-refresh-interval, max-area-addresses, max-lsp-lifetime, maximum-paths, metric, metric-style, net, passive-interface, prc-interval, protocol shutdown, redistribute isis, route priority high, router isis, set-attached-bit, set-overload-bit, show clns, show isis, show route isis, spf-interval, summary-address**.

Examples for IS-IS

This section describes configuration examples with topology for different aspects of IS-IS.

IS-IS Routing Configuration

```
router isis
   net 49.1234.aaaa.bbbb.cccc.00

interface GigabitEthernet0/0
   nameif outside
   security-level 0
   ip address 192.16.32.1 255.255.255.0
   isis
```

IS-IS IPv6 Routing Configuration

```
router isis
```

```
   net 49.1234.aaaa.bbbb.cccc.00

interface GigabitEthernet0/0
   ipv6 address 2001:192:16:32::1/64
   ipv6 router isis
```

Dynamic Routing Within the Same Area

```
iRouter --------(inside G0/1) ASA (G0/0 outside)------------ oRouter

ASA Configuration
  interface GigabitEthernet0/0
   nameif outside
   security-level 0
   ip address 192.16.32.1 255.255.255.0
   ipv6 address 2001:192:16:32::1/64
   isis
   ipv6 router isis

  interface GigabitEthernet0/1
   nameif inside
   security-level 100
   ip address 172.16.32.1 255.255.255.0 standby 172.16.32.2
   ipv6 address 2001:172:16:32::1/64 standby 2001:172:16:32::2
   isis
   ipv6 router isis

  router isis
   net 49.1234.2005.2005.2005.00
   is-type level-1
   metric-style wide

  interface GigabitEthernet0/0
   ip address 172.16.32.3 255.255.255.0
   ip router isis
   ipv6 address 2001:172:16:32::3/64
   ipv6 router isis
   isis priority 120

  interface GigabitEthernet0/1
   ip address 172.26.32.3 255.255.255.0
   ip router isis
   ipv6 address 2001:172:26:32::3/64
   ipv6 router isis

IOS Configuration
  iRouter
  router isis
   net 49.1234.2035.2035.2035.00
   is-type level-1
   metric-style wide

  oRouter
  interface GigabitEthernet0/0
   ip address 192.16.32.3 255.255.255.0
   ip router isis
   ipv6 address 2001:192:16:32::3/64
   ipv6 router isis

  oRouter
  interface GigabitEthernet0/1
   ip address 192.26.32.3 255.255.255.0
   ip router isis
   ipv6 address 2001:192:26:32::3/64
```

```
   ipv6 router isis

  oRouter
  router isis
   net 49.1234.2036.2036.2036.00
   is-type level-1
   metric-style wide
```

Dynamic Routing in More Than One Area

```
iRouter ---------- ASA ------------ oRouter

ASA Configuration
 interface GigabitEthernet0/0
  nameif outside
  security-level 80
  ip address 192.16.32.1 255.255.255.0 standby 192.16.32.2
  ipv6 address 2001:192:16:32::1/64 standby 2001:192:16:32::2
  isis
  ipv6 router isis

 interface GigabitEthernet0/1.201
  nameif inside
  security-level 100
  ip address 172.16.32.1 255.255.255.0 standby 172.16.32.2
  ipv6 address 2001:172:16:32::1/64 standby 2001:172:16:32::2
  isis
  ipv6 router isis

 router isis
  net 49.1234.2005.2005.2005.00
  metric-style wide
  maximum-paths 5
 !
 address-family ipv6 unicast
  maximum-paths 5
  exit-address-family
 !

IOS Configuration
 iRouter
 interface GigabitEthernet0/0
  ip address 172.16.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:172:16:32::3/64
  ipv6 router isis
  isis priority 120


 iRouter
 interface GigabitEthernet0/1
  ip address 172.26.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:172:26:32::3/64
  ipv6 router isis


 iRouter
 router isis
  net 49.1234.2035.2035.2035.00
  net 49.2001.2035.2035.2035.00
  is-type level-2-only
  metric-style wide
```

```
 oRouter
 interface GigabitEthernet0/0
  ip address 192.16.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:192:16:32::3/64
  ipv6 router isis


 oRouter
 interface GigabitEthernet0/1
  ip address 192.26.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:192:26:32::3/64
  ipv6 router isis


 oRouter
 router isis
  net 49.1234.2036.2036.2036.00
  is-type level-1
  metric-style wide

oRouter
 interface GigabitEthernet0/0
  ip address 192.16.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:192:16:32::3/64
  ipv6 router isis


 oRouter
 interface GigabitEthernet0/1
  ip address 192.26.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:192:26:32::3/64
  ipv6 router isis


 oRouter
 router isis
  net 49.1234.2036.2036.2036.00
  is-type level-1
  metric-style wide
```

Dynamic Routing in Overlapping Areas

```
iRouter ---------- ASA ------------ oRouter

ASA Configuration
interface GigabitEthernet0/1
 nameif inside
 security-level 100
 ip address 172.16.32.1 255.255.255.0
 ipv6 address 2001:172:16:32::1/64
 isis
 ipv6 router isis

interface GigabitEthernet0/0.301
 nameif outside
 security-level 80
 ip address 192.16.32.1 255.255.255.0
 ipv6 address 2001:192:16:32::1/64
 isis
```

```
 ipv6 router isis

router isis
 net 49.1234.2005.2005.2005.00
 authentication mode md5
 authentication key cisco#123 level-2
 metric-style wide
 summary-address 172.16.0.0 255.255.252.0
 maximum-paths 5
!
 address-family ipv6 unicast
  redistribute static level-1-2
  maximum-paths 6
 exit-address-family

IOS Configuration
iRouter
interface GigabitEthernet0/0
 ip address 172.16.32.3 255.255.255.0
 ip router isis
 ipv6 address 2001:172:16:32::3/64
 ipv6 enable
 ipv6 router isis
 isis priority 120
 isis ipv6 metric 600


interface GigabitEthernet0/1
 ip address 172.26.32.3 255.255.255.0
 ip router isis
 ipv6 address 2001:172:26:32::3/64
 ipv6 router isis


iRouter
router isis
 net 49.1234.2035.2035.2035.00
 net 49.2001.2035.2035.2035.00
 is-type level-2-only
 authentication mode md5
 authentication key-chain KeyChain level-2
 metric-style wide
 maximum-paths 6
!
 address-family ipv6
 summary-prefix 2001::/8 tag 301
 summary-prefix 6001::/16 level-1-2 tag 800
 redistribute static metric 800 level-1-2
exit-address-family


oRouter
interface GigabitEthernet0/0
 ip address 192.16.32.3 255.255.255.0
 ip pim sparse-dense-mode
 ip router isis
 ipv6 address 2001:192:16:32::3/64
 ipv6 router isis
 isis tag 301


oRouter
router isis
 net 49.1234.2036.2036.2036.00
```

```
 is-type level-1
 metric-style wide

ASA Configuration
router isis
 net 49.1234.2005.2005.2005.00
 authentication mode md5
 authentication key cisco#123 level-2
 metric-style wide
 summary-address 172.16.0.0 255.255.252.0
 maximum-paths 5
!
 address-family ipv6 unicast
  redistribute static level-1-2
  maximum-paths 6
 exit-address-family
!
```

Route Redistribution

```
iRouter ---------- ASA ------------ oRouter


 ASA Configuration
 interface GigabitEthernet0/0
  nameif outside
  security-level 80
  ip address 192.16.32.1 255.255.255.0 standby 192.16.32.2
  ipv6 address 2001:192:16:32::1/64 standby 2001:192:16:32::2
  isis
  ipv6 router isis


 interface GigabitEthernet0/1.201
  nameif inside
  security-level 100
  ip address 172.16.32.1 255.255.255.0 standby 172.16.32.2
  ipv6 address 2001:172:16:32::1/64 standby 2001:172:16:32::2
  isis
  ipv6 router isis


 router isis
  net 49.1234.2005.2005.2005.00
  metric-style wide
  redistribute isis level-2 into level-1 route-map RMAP
  maximum-paths 5
 !
 address-family ipv6 unicast
  maximum-paths 6
 exit-address-family
!

IOS Configuration
iRouter
 interface GigabitEthernet0/0
  ip address 172.16.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:172:16:32::3/64
  ipv6 router isis
  isis priority 120


 iRouter
```

```
 interface GigabitEthernet0/1
  ip address 172.26.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:172:26:32::3/64
  ipv6 router isis


 iRouter
 router isis
  net 49.1234.2035.2035.2035.00
  net 49.2001.2035.2035.2035.00
  is-type level-2-only
  metric-style wide

oRouter
 interface GigabitEthernet0/0
  ip address 192.16.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:192:16:32::3/64
  ipv6 router isis


 oRouter
 interface GigabitEthernet0/1
  ip address 192.26.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:192:26:32::3/64
  ipv6 router isis


 oRouter
 router isis
  net 49.1234.2036.2036.2036.00
  is-type level-1
  metric-style wide
```

Summary Address

```
iRouter ---------- ASA ------------ oRouter

ASA Configuration

interface GigabitEthernet0/1
 nameif inside
 security-level 100
 ip address 172.16.32.1 255.255.255.0
 ipv6 address 2001:172:16:32::1/64
 isis
 ipv6 router isis
 isis authentication key cisco#123 level-2
 isis authentication mode md5


interface GigabitEthernet0/0
 nameif outside
 security-level 80
 ip address 192.16.32.1 255.255.255.0
 ipv6 address 2001:192:16:32::1/64
 isis
 ipv6 router isis


router isis
 net 49.1234.2005.2005.2005.00
```

```
 authentication mode md5
 authentication key cisco#123 level-2
 metric-style wide
 summary-address 172.16.0.0 255.255.252.0
 redistribute static
 maximum-paths 5
address-family ipv6 unicast
 maximum-paths 6
exit-address-family
```

Passive Interfaces

```
 iRouter ---------- ASA ------------ oRouter


 ASA Configuration
 interface GigabitEthernet0/0
  nameif outside
  security-level 80
  ip address 192.16.32.1 255.255.255.0
  ipv6 address 2001:192:16:32::1/64
  isis
  ipv6 router isis


 interface GigabitEthernet0/1
  nameif inside
  security-level 100
  ip address 172.16.32.1 255.255.255.0
  ipv6 address 2001:172:16:32::1/64
  isis
  ipv6 router isis


 interface GigabitEthernet0/2
  nameif dmz
  security-level 0
  ip address 40.40.50.1 255.255.255.0
  ipv6 address 2040:95::1/64


 router isis
  net 49.1234.2005.2005.2005.00
  metric-style wide
  redistribute isis level-2 into level-1 route-map RMAP
  passive-interface default

IOS Configuration
 iRouter
  interface GigabitEthernet0/0
  ip address 172.16.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:172:16:32::3/64
  ipv6 router isis
  isis priority 120

 iRouter
 interface GigabitEthernet0/1
  ip address 172.26.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:172:26:32::3/64
  ipv6 router isis
```

```
 iRouter
 router isis
  net 49.1234.2035.2035.2035.00
  net 49.2001.2035.2035.2035.00
  is-type level-2-only
  metric-style wide

 oRouter
 interface GigabitEthernet0/0
  ip address 192.16.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:192:16:32::3/64
  ipv6 router isis

 oRouter
 interface GigabitEthernet0/1
  ip address 192.26.32.3 255.255.255.0
  ip router isis
  ipv6 address 2001:192:26:32::3/64
  ipv6 router isis

 oRouter
 router isis
  net 49.1234.2036.2036.2036.00
  is-type level-1
  metric-style wide
```

Authentication

```
ASA ------------ Router

ASA Configuration

interface GigabitEthernet0/1
 nameif inside
 security-level 100
 ip address 172.16.32.1 255.255.255.0 standby 172.16.32.2
 ipv6 address 2001:172:16:32::1/64 standby 2001:172:16:32::2
 isis
 ipv6 router isis
 isis authentication key cisco#123 level-2
 isis authentication mode md5


interface GigabitEthernet0/0.301
 nameif outside
 security-level 80
 ip address 192.16.32.1 255.255.255.0 standby 192.16.32.2
 ipv6 address 2001:192:16:32::1/64 standby 2001:192:16:32::2
 isis
 ipv6 router isis

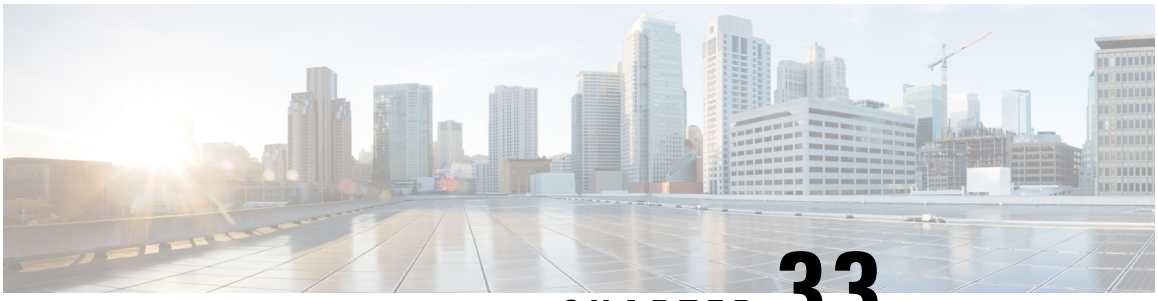

router isis
 net 49.1234.2005.2005.2005.00
 metric-style wide
 authentication mode md5
 authentication key cisco#123 level-2

IOS Configuration
iRouter
interface GigabitEthernet0/0
 ip address 172.16.32.3 255.255.255.0
```

```
 ip router isis
 ipv6 address 2001:172:16:32::3/64
 ipv6 enable
 ipv6 router isis
 isis authentication mode md5
 isis authentication key-chain KeyChain level-2
 isis priority 120
 isis ipv6 metric 600


iRouter
key chain KeyChain
 key 1
  key-string cisco#123


iRouter
router isis
 net 49.1234.2035.2035.2035.00
 net 49.2001.2035.2035.2035.00
 is-type level-2-only
 authentication mode md5
 authentication key-chain KeyChain level-2
```



CHAPTER 33

EIGRP

This chapter describes how to configure the ASA to route data, perform authentication, and redistribute routing information using the Enhanced Interior Gateway Routing Protocol (EIGRP).

- About EIGRP, on page 1049
- Guidelines for EIGRP, on page 1050
- Configure EIGRP, on page 1051
- Customize EIGRP, on page 1054
- Monitoring for EIGRP, on page 1068
- Example for EIGRP, on page 1069
- History for EIGRP, on page 1070

About EIGRP

EIGRP is an enhanced version of IGRP developed by Cisco. Unlike IGRP and RIP, EIGRP does not send out periodic route updates. EIGRP updates are sent out only when the network topology changes. Key capabilities that distinguish EIGRP from other routing protocols include fast convergence, support for variable-length subnet mask, support for partial updates, and support for multiple network layer protocols.

A router running EIGRP stores all the neighbor routing tables so that it can quickly adapt to alternate routes. If no appropriate route exists, EIGRP queries its neighbors to discover an alternate route. These queries propagate until an alternate route is found. Its support for variable-length subnet masks permits routes to be automatically summarized on a network number boundary. In addition, EIGRP can be configured to summarize on any bit boundary at any interface. EIGRP does not make periodic updates. Instead, it sends partial updates only when the metric for a route changes. Propagation of partial updates is automatically bounded so that only those routers that need the information are updated. As a result of these two capabilities, EIGRP consumes significantly less bandwidth than IGRP.

Neighbor discovery is the process that the ASA uses to dynamically learn of other routers on directly attached networks. EIGRP routers send out multicast hello packets to announce their presence on the network. When the ASA receives a hello packet from a new neighbor, it sends its topology table to the neighbor with an initialization bit set. When the neighbor receives the topology update with the initialization bit set, the neighbor sends its topology table back to the ASA.

The hello packets are sent out as multicast messages. No response is expected to a hello message. The exception to this is for statically defined neighbors. If you use the **neighbor** command, or configure the Hello Interval in ASDM, to configure a neighbor, the hello messages sent to that neighbor are sent as unicast messages. Routing updates and acknowledgements are sent out as unicast messages.

Once this neighbor relationship is established, routing updates are not exchanged unless there is a change in the network topology. The neighbor relationship is maintained through the hello packets. Each hello packet received from a neighbor includes a hold time. This is the time in which the ASA can expect to receive a hello packet from that neighbor. If the ASA does not receive a hello packet from that neighbor within the hold time advertised by that neighbor, the ASA considers that neighbor to be unavailable.

The EIGRP protocol uses four key algorithm technologies, four key technologies, including neighbor discovery/recovery, Reliable Transport Protocol (RTP), and DUAL, which is important for route computations. DUAL saves all routes to a destination in the topology table, not just the least-cost route. The least-cost route is inserted into the routing table. The other routes remain in the topology table. If the main route fails, another route is chosen from the feasible successors. A successor is a neighboring router used for packet forwarding that has a least-cost path to a destination. The feasibility calculation guarantees that the path is not part of a routing loop.

If a feasible successor is not found in the topology table, a route recomputation must occur. During route recomputation, DUAL queries the EIGRP neighbors for a route, who in turn query their neighbors. Routers that do no have a feasible successor for the route return an unreachable message.

During route recomputation, DUAL marks the route as active. By default, the ASA waits for three minutes to receive a response from its neighbors. If the ASA does not receive a response from a neighbor, the route is marked as stuck-in-active. All routes in the topology table that point to the unresponsive neighbor as a feasibility successor are removed.

Note EIGRP neighbor relationships are not supported through the IPsec tunnel without a GRE tunnel.

Null0 and EIGRP

By default, EIGRP advertises the Null0 route to the peer as summary route to prevent the router that is advertising the summary, from forwarding any packets that it does not have a route.

For example, consider the two routers, R1 and R2. The three interfaces on R1 have these networks- 192.168.0.0/24, 192.168.1.0/24, and 192.168.3.0/24. Configure R1 with summary route 192.168.0.0/22 and advertise it to R2. When R2 has an IP packet for 192.168.2.x, it would forward it to R1. R1, would drop the packet as it does not have 192.168.2.x in its routing table. However, if R1 is also connected to an ISP and it has a default route pointing to the ISP, the 192.168.2.x packet is forwarded to the ISP. To prevent this forwarding action, EIGRP generates an entry that matches the summary route, pointing to Null0. Thus, when packets for 192.168.2.x are received, R1 will drop the packet instead of using the default route.

Guidelines for EIGRP

Firewall Mode Guidelines

Supported only in routed firewall mode. Transparent firewall mode is not supported.

Cluster Guidelines

EIGRP does not form neighbor relationships with cluster peers in individual interface mode.

IPv6 Guidelines

Does not support IPv6.

Context Guidelines

- EIGRP instances cannot form adjacencies with each other across shared interfaces because, by default, inter-context exchange of multicast traffic is not supported across shared interfaces. However, you can use the static neighbor configuration under EIGRP process configuration under EIGRP process to bring up EIGRP neighbourship on a shared interface.
- Inter-context EIGRP on separate interfaces is supported.

Redistribution Guidelines

When EIGRP is configured on a device that is a part of OSPF network or vice versa, ensure that OSPF-router is configured to tag the route (EIGRP does not support route tag).

When redistributing EIGRP into OSPF and OSPF into EIGRP, a routing loop occurs when there is an outage on one of the links, interfaces, or even when the route originator is down. To prevent the redistribution of routes from one domain back into the same domain, a router can tag a route that belongs to a domain while it is redistributing, and those routes can be filtered on the remote router based on the same tag. Because the routes will not be installed into the routing table, they will not be redistributed back into the same domain.

Additional Guidelines

- A maximum of one EIGRP process is supported.
- EIGRP adjacency flap occurs whenever a configuration change is applied which results in modifying the routing information (sent or received) from neighbors especially in distribute lists, offset lists, and changes to summarization. After the routers are synchronized, EIGRP reestablishes the adjacency between neighbors. When an adjacency is torn down and reestablished, all learned routes between the neighbors are erased and the entire synchronization between the neighbors is performed newly with the new distribute list.
- There is no restriction on the maximum number of EIGRP neignbours. However, to prevent unnecessary EIGRP flap, we recommend you to limit the number to 500 per unit.

Configure EIGRP

This section describes how to enable the EIGRP process on your system. After you have enabled EIGRP, see the following sections to learn how to customize the EIGRP process on your system.

Enable EIGRP

You can only enable one EIGRP routing process on the ASA.

Procedure

Step 1 Create an EIGRP routing process and enter router configuration mode for this EIGRP process:

router eigrp as-num

Example:

```
ciscoasa(config)# router eigrp 2
```

To enable EIGRP IPv6 routing process, enter the following command:

The *as-num* argument is the autonomous system number of the EIGRP routing process.

Step 2 Configure the interfaces and networks that participate in EIGRP routing:

network ip-addr [mask]

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
```

You can configure one or more **network** statements with this command.

Directly connected and static networks that fall within the defined network are advertised by the ASA. Additionally, only interfaces with an IP address that fall within the defined network participate in the EIGRP routing process.

If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see Configure Interfaces for EIGRP, on page 1054.

Enable EIGRP Stub Routing

You can enable, and configure the ASA as an EIGRP stub router. Stub routing decreases memory and processing requirements on the ASA. As a stub router, the ASA does not need to maintain a complete EIGRP routing table because it forwards all nonlocal traffic to a distribution router. Generally, the distribution router need not send anything more than a default route to the stub router.

Only specified routes are propagated from the stub router to the distribution router. As a stub router, the ASA responds to all queries for summaries, connected routes, redistributed static routes, external routes, and internal routes with the message “inaccessible.” When the ASA is configured as a stub, it sends a special peer information packet to all neighboring routers to report its status as a stub router. Any neighbor that receives a packet informing it of the stub status will not query the stub router for any routes, and a router that has a stub peer will not query that peer. The stub router depends on the distribution router to send the correct updates to all peers.

Procedure

Step 1 Create an EIGRP routing process and enter router configuration mode for this EIGRP process:

router eigrp as-num

Example:

```
ciscoasa(config)# router eigrp 2
```

The *as-num* argument is the autonomous system number of the EIGRP routing process.

Step 2 Configure the interfaces and networks that participate in EIGRP routing:

network ip-addr [mask]

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
```

You can configure one or more **network** statements with this command.

Directly connected and static networks that fall within the defined network are advertised by the ASA. Additionally, only interfaces with an IP address that fall within the defined network participate in the EIGRP routing process.

If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see section Configure Passive Interfaces, on page 1056.

Step 3 Configure the stub routing process:

eigrp stub {**receive-only** | [**connected**] [**redistributed**] [**static**] [**summary**]}

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
ciscoasa(config-router)# eigrp stub {receive-only | [connected] [redistributed] [static]
[summary]}
```

You must specify which networks are advertised by the stub routing process to the distribution router. Static and connected networks are not automatically redistributed into the stub routing process.

Note
A stub routing process does not maintain a full topology table. At a minimum, stub routing needs a default route to a distribution router, which makes the routing decisions.

Customize EIGRP

This section describes how to customize the EIGRP routing.

Define a Network for an EIGRP Routing Process

The Network table lets you specify the networks used by the EIGRP routing process. For an interface to participate in EIGRP routing, it must fall within the range of addresses defined by the network entries. For directly connected and static networks to be advertised, they must also fall within the range of the network entries.

The Network table displays the networks configured for the EIGRP routing process. Each row of the table displays the network address and associated mask configured for the specified EIGRP routing process.

Procedure

Step 1 Create an EIGRP routing process and enter router configuration mode for this EIGRP process:

router eigrp as-num

Example:

```
ciscoasa(config)# router eigrp 2
```

The *as-num* argument is the autonomous system number of the EIGRP routing process.

Step 2 Configure the interfaces and networks that participate in EIGRP routing:

network ip-addr [mask]

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
```

You can configure one or more **network** statements with this command.

Directly connected and static networks that fall within the defined network are advertised by the ASA. Additionally, only interfaces with an IP address that fall within the defined network participate in the EIGRP routing process.

If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see Configure Passive Interfaces, on page 1056.

Configure Interfaces for EIGRP

If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, you can configure a **network** command that includes the network to which

the interface is attached, and use the **passive-interface** command to prevent that interface from sending or receiving EIGRP updates.

Procedure

Step 1 Create an EIGRP routing process and enter router configuration mode for this EIGRP process:

router eigrp as-num

Example:

```
ciscoasa(config)# router eigrp 2
```

The *as-num* argument is the autonomous system number of the EIGRP routing process.

Step 2 Configure the interfaces and networks that participate in EIGRP routing:

network ip-addr [mask]

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
```

You can configure one or more network statements with this command.

Directly connected and static networks that fall within the defined network are advertised by the ASA. Additionally, only interfaces with an IP address that fall within the defined network participate in the EIGRP routing process.

If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see Define a Network for an EIGRP Routing Process, on page 1054.

Step 3 Control the sending or receiving of candidate default route information:

no default-information {**in** | **out** | **WORD**}

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
ciscoasa(config-router)# no default-information {in | out | WORD}
```

Entering the **no default-information in** command causes the candidate default route bit to be blocked on received routes.

Entering the **no default-information out** command disables the setting of the default route bit in advertised routes.

For more information see, Configure Default Information in EIGRP, on page 1066.

Step 4 Enable MD5 authentication of EIGRP packets:

authentication mode eigrp as-num **md5**

Example:

```
ciscoasa(config)# authentication mode eigrp 2 md5
```

The *as-num* argument is the autonomous system number of the EIGRP routing process configured on the ASA. If EIGRP is not enabled or if you enter the wrong number, the ASA returns the following error message:

```
% Asystem(100) specified does not exist
```

For more information see Enable EIGRP Authentication on an Interface, on page 1059.

Step 5 Set the delay value:

delay *value*

Example:

```
ciscoasa(config-if)# delay 200
```

The *value* argument entered is in tens of microseconds. To set the delay for 2000 microseconds, enter a *value* of 200.

To view the delay value assigned to an interface, use the **show interface** command.

For more information, see Change the Interface Delay Value, on page 1058.

Step 6 Change the hello interval:

hello-interval eigrp as-num seconds

Example:

```
ciscoasa(config)# hello-interval eigrp 2 60
```

For more information see Customize the EIGRP Hello Interval and Hold Time, on page 1064.

Step 7 Change the hold time:

hold-time eigrp as-num seconds

Example:

```
ciscoasa(config)# hold-time eigrp 2 60
```

For more information see Customize the EIGRP Hello Interval and Hold Time, on page 1064.

Configure Passive Interfaces

You can configure one or more interfaces as passive interfaces. In EIGRP, a passive interface does not send or receive routing updates.

Procedure

Step 1 Create an EIGRP routing process and enter router configuration mode for this EIGRP process:

router eigrp as-num

Example:

```
ciscoasa(config)# router eigrp 2
```

The *as-num* argument is the autonomous system number of the EIGRP routing process.

Step 2 Configure the interfaces and networks that participate in EIGRP routing. You can configure one or more **network** statements with this command:

network ip-addr [mask]

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
```

Directly connected and static networks that fall within the defined network are advertised by the ASA. Additionally, only interfaces with an IP address that fall within the defined network participate in the EIGRP routing process.

If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see Define a Network for an EIGRP Routing Process, on page 1054.

Step 3 Prevent an interface from sending or receiving EIGRP routing message:

passive-interface {**default** | if-name}

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
ciscoasa(config-router)# passive-interface {default}
```

Using the **default** keyword disables EIGRP routing updates on all interfaces. Specifying an interface name, as defined by the **nameif** command, disables EIGRP routing updates on the specified interface. You can use multiple **passive-interface** commands in your EIGRP router configuration.

Configure the Summary Aggregate Addresses on Interfaces

You can configure a summary addresses on a per-interface basis. You need to manually define summary addresses if you want to create summary addresses that do not occur at a network number boundary or if you want to use summary addresses on an ASA with automatic route summarization disabled. If any more specific routes are in the routing table, EIGRP will advertise the summary address out the interface with a metric equal to the minimum of all more specific routes.

Procedure

Step 1 Enter interface configuration mode for the interface on which you are changing the delay value used by EIGRP:

interface phy_if

Example:

```
ciscoasa(config)# interface inside
```

Step 2 Create the summary address:

summary-address eigrp *as-num address mask* [*distance*]

Example:

```
ciscoasa(config-if)# summary-address eigrp 2 address mask [20]
```

By default, EIGRP summary addresses that you define have an administrative distance of 5. You can change this value by specifying the optional *distance* argument in the **summary-address** command.

Change the Interface Delay Value

The interface delay value is used in EIGRP distance calculations. You can modify this value on a per-interface basis.

Procedure

Step 1 Enter interface configuration mode for the interface on which you are changing the delay value used by EIGRP:

interface phy_if

Example:

```
ciscoasa(config)# interface inside
```

Step 2 Set a delay value:

delay *value*

Example:

```
ciscoasa(config-if)# delay 200
```

The *value* argument entered is in tens of microseconds. To set the delay for 2000 microseconds, you enter a *value* of 200.

Note

To view the delay value assigned to an interface, use the **show interface** command.

Enable EIGRP Authentication on an Interface

EIGRP route authentication provides MD5 authentication of routing updates from the EIGRP routing protocol. The MD5 keyed digest in each EIGRP packet prevents the introduction of unauthorized or false routing messages from unapproved sources.

EIGRP route authentication is configured on a per-interface basis. All EIGRP neighbors on interfaces configured for EIGRP message authentication must be configured with the same authentication mode and key for adjacencies to be established.

Note Before you can enable EIGRP route authentication, you must enable EIGRP.

Procedure

Step 1 Create an EIGRP routing process and enter router configuration mode for this EIGRP process:

router eigrp *as-num*

Example:

```
ciscoasa(config)# router eigrp 2
```

The as-num argument is the autonomous system number of the EIGRP routing process.

Step 2 Configure the interfaces and networks that participate in EIGRP routing:

network *ip-addr* [*mask*]

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
```

- You can configure one or more network statements with this command.
- Directly connected and static networks that fall within the defined network are advertised by the ASA. Additionally, only interfaces with an IP address that falls within the defined network participate in the EIGRP routing process.
- If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see Configure EIGRP, on page 1051.

Step 3 Enter interface configuration mode for the interface on which you are configuring EIGRP message authentication:

interface *phy_if*

Example:

```
ciscoasa(config)# interface inside
```

Step 4 Enable MD5 authentication of EIGRP packets:

authentication mode eigrp *as-num* md5

Example:

```
ciscoasa(config)# authentication mode eigrp 2 md5
```

The as-num argument is the autonomous system number of the EIGRP routing process configured on the ASA. If EIGRP is not enabled or if you enter the wrong number, the ASA returns the following error message:

```
% Asystem(100) specified does not exist
```

Step 5 Configure the key used by the MD5 algorithm:

authentication key eigrp *as-num key* key-id *key-id*

Example:

```
ciscoasa(config)# authentication key eigrp 2 cisco key-id 200
```

- The as-num argument is the autonomous system number of the EIGRP routing process configured on the ASA. If EIGRP is not enabled or if you enter the wrong number, the ASA returns the following error message:

  ```
  % Asystem(100) specified does not exist%
  ```

- The key argument can include up to 16 characters, including alphabets, numbers and special characters.White spaces are not allowed, in the key argument.
- The key-id argument is a number that can range from 0 to 255.

Define an EIGRP Neighbor

EIGRP hello packets are sent as multicast packets. If an EIGRP neighbor is located across a non broadcast network, such as a tunnel, you must manually define that neighbor. When you manually define an EIGRP neighbor, hello packets are sent to that neighbor as unicast messages.

Procedure

Step 1 Create an EIGRP routing process and enters router configuration mode for this EIGRP process:

router eigrp as-num

Example:

```
ciscoasa(config)# router eigrp 2
```

The *as-num* argument is the autonomous system number of the EIGRP routing process.

Step 2 Configure the interfaces and networks that participate in EIGRP routing:

network ip-addr [mask]

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
```

Step 3 Define the static neighbor:

neighbor *ip-addr* **interface** *if_name*

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# neighbor 10.0.0.0 interface interface1
```

The *ip-addr* argument is the IP address of the neighbor.

The *if-name* argument is the name of the interface, as specified by the **nameif** command, through which that neighbor is available. You can define multiple neighbors for an EIGRP routing process.

Note
You must configure the interfaces network that participate in the EIGRP routing for the neighborship configuration to be effective.

Redistribute Routes Into EIGRP

You can redistribute routes discovered by RIP and OSPF into the EIGRP routing process. You can also redistribute static and connected routes into the EIGRP routing process. You do not need to redistribute connected routes if they fall within the range of a **network** statement in the EIGRP configuration.

Note For RIP only: Before you begin this procedure, you must create a route map to further define which routes from the specified routing protocol are redistributed in to the RIP routing process.

Procedure

Step 1 Create an EIGRP routing process and enter router configuration mode for this EIGRP process:

router eigrp as-num

Example:

```
ciscoasa(config)# router eigrp 2
```

The *as-num* argument is the autonomous system number of the EIGRP routing process.

Step 2 (Optional) Specifies the default metrics that should be applied to routes redistributed into the EIGRP routing process:

default-metric bandwidth delay reliability loading mtu

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# default-metric bandwidth delay reliability loading mtu
```

If you do not specify a default metric in the EIGRP router configuration, you must specify the metric values in each **redistribute** command. If you specify the EIGRP metrics in the **redistribute** command and have the **default-metric** command in the EIGRP router configuration, the metrics in the **redistribute** command are used.

Step 3 Redistribute connected routes into the EIGRP routing process:

redistribute connected [**metric** bandwidth delay reliability loading mtu] [**route-map** map_name]

Example:

```
ciscoasa(config-router)# redistribute connected [metric bandwidth delay reliability loading
 mtu] [route-map map_name]
```

You must specify the EIGRP metric values in the **redistribute** command if you do not have a **default-metric** command in the EIGRP router configuration.

Step 4 Redistribute static routes into the EIGRP routing process:

redistribute static [**route-map** map_name]

Example:

```
ciscoasa(config-router)# redistribute static [route-map map_name]
```

This command will pass all the static routes to EIGRP. To redistribute selective static routes, ensure to create an access-list with the static route and then include it in a route-map:

Example:

```
ciscoasa(config)# ip access-list extended R1_Loopback
ciscoasa(config-ext-nacl)#permit ip host 1.1.1.1 any
ciscoasa(config-ext-nacl)#exit

ciscoasa(config)#route-map Permit_to_Distribute
ciscoasa(config-route-map)#match ip address R1_Loopback
ciscoasa(config-route-map)#exit
```

After creating the route-map, include it in the redistribute command as follows:

Example:

```
ciscoasa(config)#router eigrp 2
ciscoasa(config-router)#redistribute static subnets route-map Permit_to_Distribute
```

Step 5 Redistribute routes from an OSPF routing process into the EIGRP routing process:

redistribute ospf pid [**match** {**internal | external** [**1** | **2**] | **nssa-external** [**1** | **2**]}] [**metric** bandwidth delay reliability loading mtu] [**route-map** map_name]

Example:

```
ciscoasa(config-router): redistribute ospf pid [match {internal | external [1 | 2] |
nssa-external [1 | 2]}] [metric bandwidth delay reliability loading mtu] [route-map map_name]
```

Step 6 Redistribute routes from a RIP routing process into the EIGRP routing process:

redistribute rip [**metric** bandwidth delay reliability load mtu] [**route-map map_name**]

Example:

```
ciscoasa(config-router): redistribute rip [metric bandwidth delay
reliability load mtu] [route-map map_name]
```

Filter Networks in EIGRP



Note Before you begin this process, you must create a standard ACL that defines the routes that you want to advertise. That is, create a standard ACL that defines the routes that you want to filter from sending or receiving updates.

Procedure

Step 1 Create an EIGRP routing process and enter router configuration mode for this EIGRP process:

router eigrp as-num

Example:

```
ciscoasa(config)# router eigrp 2
```

The *as-num* argument is the autonomous system number of the EIGRP routing process.

Step 2 Configure the interfaces and networks that participate in EIGRP routing:

ciscoasa(config-router)# **network** ip-addr [mask]

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
```

You can configure one or more network statements with this command.

Directly connected and static networks that fall within the defined network are advertised by the ASA. Additionally, only interfaces with an IP address that fall within the defined network participate in the EIGRP routing process.

If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see Configure Interfaces for EIGRP, on page 1054.

Step 3 Filter networks sent in EIGRP routing updates:

distribute-list acl **out** [**connected** | **ospf** | **rip** | **static** | **interface** if_name]

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
ciscoasa(config-router): distribute-list acl out [connected]
```

You can specify an interface to apply the filter to only those updates that are sent by that specific interface.

You can enter multiple **distribute-list** commands in your EIGRP router configuration.

Step 4 Filter networks received in EIGRP routing updates:

distribute-list acl **in** [**interface** if_name]

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
ciscoasa(config-router): distribute-list acl in [interface interface1]
```

You can specify an interface to apply the filter to only those updates that are received by that interface.

Customize the EIGRP Hello Interval and Hold Time

The ASA periodically sends hello packets to discover neighbors and to learn when neighbors become unreachable or inoperative. By default, hello packets are sent every 5 seconds.

The hello packet advertises the ASA hold time. The hold time indicates to EIGRP neighbors the length of time the neighbor should consider the ASA reachable. If the neighbor does not receive a hello packet within the advertised hold time, then the ASA is considered unreachable. By default, the advertised hold time is 15 seconds (three times the hello interval).

Both the hello interval and the advertised hold time are configured on a per-interface basis. We recommend setting the hold time to be at minimum three times the hello interval.

Procedure

Step 1 Enter interface configuration mode for the interface on which you are configuring the hello interval or advertised hold time:

interface phy_if

Example:

```
ciscoasa(config)# interface inside
```

Step 2 Change the hello interval:

hello-interval eigrp as-num seconds

Example:

```
ciscoasa(config)# hello-interval eigrp 2 60
```

Step 3 Change the hold time:

hold-time eigrp as-num seconds

Example:

```
ciscoasa(config)# hold-time eigrp 2 60
```

Disable Automatic Route Summarization

Automatic route summarization is enabled by default. The EIGRP routing process summarizes on network number boundaries. This can cause routing problems if you have noncontiguous networks.

For example, if you have a router with the networks 192.168.1.0, 192.168.2.0, and 192.168.3.0 connected to it, and those networks all participate in EIGRP, the EIGRP routing process creates the summary address 192.168.0.0 for those routes. If an additional router is added to the network with the networks 192.168.10.0 and 192.168.11.0, and those networks participate in EIGRP, they will also be summarized as 192.168.0.0. To prevent the possibility of traffic being routed to the wrong location, you should disable automatic route summarization on the routers creating the conflicting summary addresses.

Procedure

Step 1 Create an EIGRP routing process and enter router configuration mode for this EIGRP process:

router eigrp as-num

Example:

```
ciscoasa(config)# router eigrp 2
```

The *as-num* argument is the autonomous system number of the EIGRP routing process.

Step 2 Disable automatic route summarization:

no auto-summary

Example:

```
ciscoasa(config-router)# no auto-summary
```

Automatic summary addresses have a default administrative distance of 5.

Configure Default Information in EIGRP

You can control the sending and receiving of default route information in EIGRP updates. By default, default routes are sent and accepted. Configuring the ASA to disallow default information to be received causes the candidate default route bit to be blocked on received routes. Configuring the ASA to disallow default information to be sent disables the setting of the default route bit in advertised routes.

Procedure

Step 1 Create an EIGRP routing process and enter router configuration mode for this EIGRP process:

router eigrp as-num

Example:

```
ciscoasa(config)# router eigrp 2
```

The *as-num* argument is the autonomous system number of the EIGRP routing process.

Step 2 Configure the interfaces and networks that participate in EIGRP routing:

network ip-addr [mask]

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
```

You can configure one or more network statements with this command.

Directly connected and static networks that fall within the defined network are advertised by the ASA. Additionally, only interfaces with an IP address that fall within the defined network participate in the EIGRP routing process.

If you have an interface that you do not want to have participate in EIGRP routing, but that is attached to a network that you want advertised, see Configure Interfaces for EIGRP, on page 1054.

Step 3 Control the sending or receiving of candidate default route information:

no default-information {**in** | **out** | **WORD**}

Example:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
ciscoasa(config-router)# no default-information {in | out | WORD}
```

Note
Entering the **no default-information in** command causes the candidate default route bit to be blocked on received routes. Entering the **no default-information out** command disables the setting of the default route bit in advertised routes.

Disable EIGRP Split Horizon

Split horizon controls the sending of EIGRP update and query packets. When split horizon is enabled on an interface, update and query packets are not sent for destinations for which this interface is the next hop. Controlling update and query packets in this manner reduces the possibility of routing loops.

By default, split horizon is enabled on all interfaces.

Split horizon blocks route information from being advertised by a router out of any interface from which that information originated. This behavior usually optimizes communications among multiple routing devices, particularly when links are broken. However, with nonbroadcast networks, there may be situations where this behavior is not desired. For these situations, including networks in which you have EIGRP configured, you may want to disable split horizon.

If you disable split horizon on an interface, you must disable it for all routers and access servers on that interface.

To disable EIGRP split horizon, perform the following steps:

Procedure

Step 1 Enter interface configuration mode for the interface on which you are changing the delay value used by EIGRP:

interface phy_if

Example:

```
ciscoasa(config)# interface phy_if
```

Step 2 Disable the split horizon:

no split-horizon eigrp *as-number*

Example:

```
ciscoasa(config-if)# no split-horizon eigrp 2
```

Restart the EIGRP Process

You can restart an EIGRP process or clear redistribution or clear counters.

Procedure

Restart an EIGRP process or clear redistribution or clear counters:

clear eigrp pid {**1-65535** | **neighbors** | **topology** | **events)**}

Example:

```
ciscoasa(config)# clear eigrp pid 10 neighbors
```

Monitoring for EIGRP

You can use the following commands to monitor the EIGRP routing process. For examples and descriptions of the command output, see the command reference. Additionally, you can disable the logging of neighbor change messages and neighbor warning messages.

To monitor or disable various EIGRP routing statistics, enter one of the following commands:

- **router-id**

  Displays the router-id for this EIGRP process.

- **show eigrp** [*as-number*] **events** [{*start end*} | **type**]

  Displays the EIGRP event log.

- **show eigrp** [*as-number*] **interfaces** [*if-name*] [**detail**]

  Displays the interfaces participating in EIGRP routing.

- **show eigrp** [as-number] **neighbors** [**detail** | **static**] [*if-name*]

  Displays the EIGRP neighbor table.

- **show eigrp** [*as-number*] **topology** [*ip-addr* [**mask**] | **active** | **all-links** | **pending** | **summary** | **zero-successors**]

  Displays the EIGRP topology table.

- **show eigrp** [*as-number*] **traffic**

  Displays EIGRP traffic statistics.

- **show mfib cluster**

  Displays MFIB information in terms of forwarding entries and interfaces.

- **show route cluster**

  Displays additional route synchronization details for clustering.

- **no eigrp log-neighbor-changes**

  Disables the logging of neighbor change messages. Enter this command in router configuration mode for the EIGRP routing process.

- **no eigrp log-neighbor-warnings**

  Disables the logging of neighbor warning messages.

- **show ipv6 eigrp** *as-number* **interface** *interface*

  Displays the EIGRP IPv6 topology table.

- **show ipv6 eigrp** [*as-number*] **traffic**

  Displays EIGRP IPv6 traffic statistics.

- **show ipv6 eigrp** [as-number] **neighbors** [*if-name*]

  Displays the EIGRP IPv6 neighbor table.

- **show ipv6 eigrp interfaces** [*if-name*]

  Displays neighbor related information with respect to given interface.

- **show ipv6 eigrp** [*as-number*] **topology** [*ipv6-address* [**mask**] | **active** | **all-links** | **pending** | **summary** | **zero-successors**]

  Displays the EIGRP IPv6 topology table.

- **show ipv6 eigrp** [*as-number*] **events** [{*start* - *end*} | **type**]

  Displays the EIGRP IPv6 event log.

- **show ipv6 eigrp timers**

  Displays the configured hello timer and hold timer.

Example for EIGRP

The following example shows how to enable and configure EIGRP with various optional processes:

Procedure

Step 1 To enable EIGRP, enter the following commands:

```
ciscoasa(config)# router eigrp 2
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
```

Step 2 To configure an interface from sending or receiving EIGRP routing messages, enter the following command:

```
ciscoasa(config-router)# passive-interface {default}
```

Step 3 To define an EIGRP neighbor, enter the following command:

```
ciscoasa(config-router)# neighbor 10.0.0.0 interface interface1
```

Step 4 To configure the interfaces and networks that participate in EIGRP routing, enter the following command:

```
ciscoasa(config-router)# network 10.0.0.0 255.0.0.0
```

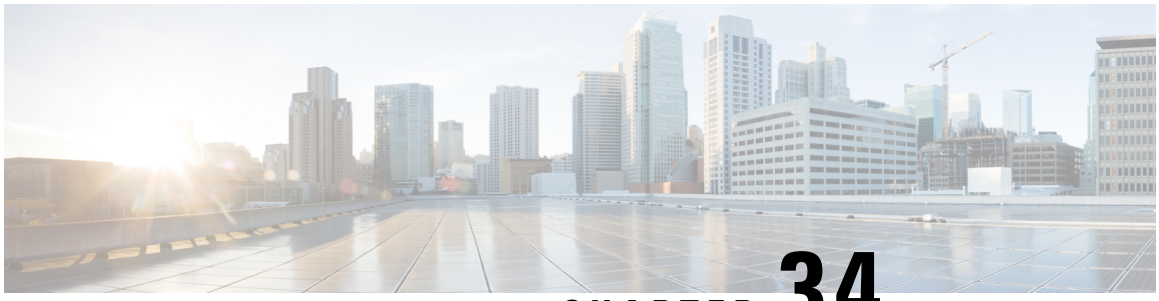
Step 5 To change the interface delay value used in EIGRP distance calculations, enter the following commands:

```
ciscoasa(config-router)# exit
ciscoasa(config)# interface phy_if
ciscoasa(config-if)# delay 200
```

History for EIGRP

Table 40: Feature History for EIGRP

Feature Name	Platform Releases	Feature Information
EIGRP Support	7.0(1)	Support was added for routing data, performing authentication, and redistributing and monitoring routing information using the Enhanced Interior Gateway Routing Protocol (EIGRP). We introduced the following command: **route eigrp**.
Dynamic Routing in Multiple Context Mode	9.0(1)	EIGRP routing is supported in multiple context mode.
Clustering	9.0(1)	For EIGRP, bulk synchronization, route synchronization, and layer 2 load balancing are supported in the clustering environment. We introduced or modified the following commands: **show route cluster**, **debug route cluster, show mfib cluster, debug mfib cluster**.
EIGRP Auto-Summary	9.2(1)	For EIGRP, the Auto-Summary field is now disabled by default.



CHAPTER 34

Multicast Routing

This chapter describes how to configure the ASA to use the multicast routing protocol.

- Multicast routing, on page 1071
- Guidelines for multicast routing, on page 1075
- Enable Multicast Routing, on page 1075
- Customize Multicast Routing, on page 1076
- Monitoring for PIM, on page 1088
- Example for Multicast Routing, on page 1089
- History for Multicast Routing, on page 1090

Multicast routing

Multicast routing is a bandwidth-conserving technology that reduces traffic by simultaneously delivering a single stream of information to thousands of corporate recipients and homes. Applications include videoconferencing, corporate communications, distance learning, and distributing software, stock quotes, and news.

Multicast routing applications and protocols

Multicast routing protocols send source traffic to multiple receivers efficiently, without adding extra load on the source or receivers, and use less network bandwidth than other technologies. Multicast packets are replicated in the network by ASA enabled with Protocol Independent Multicast (PIM) and other supporting multicast protocols, which results in the most efficient delivery of data to multiple receivers possible.

The ASA supports both stub multicast routing and PIM multicast routing. However, you cannot configure both concurrently on a single ASA.

Note The UDP and non-UDP transports are both supported for multicast routing. However, the non-UDP transport has no FastPath optimization.

Stub multicast routing

Stub multicast routing provides dynamic host registration and facilitates multicast routing.

When configured for stub multicast routing, the ASA acts as an IGMP proxy agent. Instead of fully participating in multicast routing, the ASA forwards IGMP messages to an upstream multicast router, which sets up delivery of the multicast data. When configured for stub multicast routing, the ASA cannot be configured for PIM sparse or bidirectional mode. Enable PIM on the interfaces that participate in IGMP stub multicast routing.

The ASA supports both PIM-SM and bidirectional PIM. PIM-SM is a multicast routing protocol that uses the underlying unicast routing information base or a separate multicast-capable routing information base. It builds unidirectional shared trees, each rooted at a single Rendezvous Point (RP) per multicast group. Optionally, it creates shortest-path trees for each multicast source.

PIM multicast routing

Bidirectional PIM is a variant of PIM-SM that builds bidirectional shared trees connecting multicast sources and receivers.

DF election process

Bidirectional trees are built using a Designated Forwarder (DF) election process operating on each link of the multicast topology. With the assistance of the DF, multicast data is forwarded from sources to the Rendezvous Point (RP) and then along the shared tree to receivers. This process does not require source-specific state. The DF election takes place during RP discovery and provides a default route to the RP.



Note If the ASA is the PIM RP, use the untranslated outside address of the ASA as the RP address.

PIM sparse mode

PIM Sparse Mode (PIM-SM) is a PIM mode that forwards traffic only when specifically requested by receivers. It uses a central Rendezvous Point (RP) to coordinate the multicast traffic forwarding between the multicast source and the destination. RP is a router that serves as the meeting point for the source and the receivers. Multicast sources register with the RP through their first hop router. They send a copy of the multicast traffic to the receivers through the RP.

PIM source specific multicast support

ASA can forward source specific multicast traffic (e.g., for groups in the 232.x.x.x range) even though they do not support PIM-SSM configuration or Internet Group Management Protocol version 3 (IGMPv3).

SSM is classified as a data delivery mechanism for one-to-many applications such as IPTV. The SSM model uses a concept of "channels" denoted by an (S,G) pair, where S is a source address and G is an SSM destination address. Subscribing to a channel is achieved by using a group management protocol such as IGMPv3. SSM enables a receiving client, once it has learned about a particular multicast source, to receive multicast streams directly from the source rather than receiving it from a shared Rendezvous Point (RP). Access control mechanisms are introduced within SSM, providing a security enhancement that is not available with current sparse or sparse-dense mode implementations.

Limitation:

- ASA cannot act as the last-hop router for SSM multicast because it does not support IGMPv3 (which receivers use to join SSM groups).

If the ASA is the last-hop, it will ignore IGMPv3 join messages from receivers for SSM groups, and SSM forwarding will not work.

- Static multicast routes on ASA do not work for SSM range (232.x.x.x).

Workaround:

For Cisco ASA firewall to forward SSM-related multicast traffic, you must add a multicast-capable layer 3 device (such as a router or switch) that supports PIM and IGMPv3 on the same network segment as the receivers.

How it Works:

- Receivers register their SSM group joins (IGMPv3) with this layer 3 device.
- This layer 3 device sends PIM join messages towards the ASA.
- The ASA receives these PIM messages and dynamically learns multicast routes.
- SSM multicast traffic is then properly forwarded by the ASA, since it is no longer the last-hop device.

PIM bootstrap router (BSR)

PIM Bootstrap Router (BSR) is a dynamic Rendezvous Point (RP) selection model that uses candidate routers for RP function and for relaying the RP information for a group.

The RP function includes RP discovery and provides a default route to the RP. It does this by configuring a set of devices as candidate BSRs (C-BSR) which participate in a BSR election process to choose a BSR amongst themselves. Once the BSR is chosen, devices that are configured as candidate Rendezvous Points (C-RP) start sending their group mapping to the elected BSR. The BSR then distributes the group-to-RP mapping information to all the other devices down the multicast tree through BSR messages that travel from PIM router to PIM router on a per-hop basis.

This feature allows networks to learn RPs dynamically. Dynamic RP discovery is essential in large, complex networks where an RP may occasionally go down and come up.

PIM bootstrap router (BSR) terminology

This reference provides definitions for key terminology used in PIM Bootstrap Router (BSR) configurations to help understand the roles and functions of different components in the BSR election and RP advertisement process.

These terms are frequently referenced in the PIM BSR configuration:

- Bootstrap Router (BSR) — A BSR advertises Rendezvous Point (RP) information to other routers with PIM on a hop-by-hop basis. Among multiple Candidate-BSRs, a single BSR is chosen after an election process. The primary purpose of this Bootstrap router is to collect all Candidate-RP (C-RP) announcements in to a database called the RP-set and to periodically send this out to all other routers in the network as BSR messages (every 60 seconds).
- Bootstrap Router (BSR) messages — BSR messages are multicast to the All-PIM-Routers group with a TTL of 1. All PIM neighbors that receive these messages retransmit them (again with a TTL of 1) out of all interfaces except the one in which the messages were received. BSR messages contain the RP-set and the IP address of the currently active BSR. This is how C-RPs know where to unicast their C-RP messages.

- Candidate Bootstrap Router (C-BSR) — A device that is configured as a candidate-BSR participates in the BSR election mechanism. A C-BSR with highest priority is elected as the BSR. The highest IP address of the C-BSR is used as a tiebreaker. The BSR election process is preemptive, for example if a new C-BSR with a higher priority comes up, it triggers a new election process.
- Candidate Rendezvous Point (C-RP) — An RP acts as a meeting place for sources and receivers of multicast data. A device that is configured as a C-RP periodically advertises the multicast group mapping information directly to the elected BSR through unicast. These messages contain the Group-range, C-RP address, and a hold time. The IP address of the current BSR is learned from the periodic BSR messages that are received by all routers in the network. In this way, the BSR learns about possible RPs that are currently up and reachable.



Note The ASA does not act as a C-RP, even though the C-RP is a mandatory requirement for BSR traffic. Only routers can act as a C-RP. So, for BSR testing functionality, you must add routers to the topology.

- BSR Election Mechanism — Each C-BSR originates Bootstrap messages (BSMs) that contain a BSR Priority field. Routers within the domain flood the BSMs throughout the domain. A C-BSR that hears about a higher-priority C-BSR than itself suppresses its sending of further BSMs for some period of time. The single remaining C-BSR becomes the elected BSR, and its BSMs inform all the other routers in the domain that it is the elected BSR.

Multicast group concept

Multicast is based on the concept of a group. An arbitrary group of receivers expresses an interest in receiving a particular data stream. This group does not have any physical or geographical boundaries. Hosts can be located anywhere on the Internet. Hosts that want to receive data sent to a group must join the group using IGMP.

Group membership requirements

Hosts must be a member of the group to receive the data stream.

Multicast Addresses

Multicast addresses specify an arbitrary group of IP hosts that have joined the group and want to receive traffic sent to this group.

Clustering

Multicast routing supports clustering. In Spanned EtherChannel clustering, the control unit sends all multicast routing packets and data packets until fast-path forwarding is established.

After fast-path forwarding is established, data units may forward multicast data packets. All data flows are full flows. Stub forwarding flows are also supported. Because only one unit receives multicast packets in Spanned EtherChannel clustering, redirection to the control unit is common. In Individual Interface clustering, units do not act independently. All data and routing packets are processed and forwarded by the control unit. Data units drop all packets that have been sent.

Guidelines for multicast routing

Table 41: System requirements

Item	Description
Context mode	Supported in single context mode.
Firewall mode	Supported only in routed firewall mode. Transparent firewall mode is not supported.
IPv6	IPv6 is not supported.
Multicast group	The range of addresses from 224.0.0.0 to 224.0.0.255 is reserved for routing protocols and other topology discovery or maintenance protocols, such as gateway discovery and group membership reporting. Internet multicast routing from the 224.0.0/24 address range is not supported. When multicast routing is enabled for reserved addresses, an IGMP group is not created.
Clustering	In clustering, for IGMP and PIM, this feature is only supported on the primary unit.

Access control configuration

You must configure an access control rule on the inbound interface to allow traffic to the multicast host, such as 224.1.2.3. However, you cannot specify a destination interface for the rule, or it cannot be applied to multicast connections during initial connection validation.

Interface and protocol management

Ensure proper configuration management when working with PIM-enabled interfaces and multicast routing protocols.

- Multicast routing is not supported in ECMP.
- Do not configure ASA to simultaneously be a Rendezvous Point (RP) and a First Hop Router.
- The HSRP standby IP address does not participate in PIM neighborship. If the RP router IP is routed through the HSRP standby IP address, multicast routing does not work in ASA. To ensure multicast traffic passes through successfully, configure the route for the RP address to use an interface IP address rather than the HSRP standby IP address.

Enable Multicast Routing

Enabling multicast routing on the ASA, enables IGMP and PIM on all data interfaces by default, but not on the management interface for most models (see Management Slot/Port Interface, on page 594 for interfaces that do not allow through traffic). IGMP is used to learn whether members of a group are present on directly

attached subnets. Hosts join multicast groups by sending IGMP report messages. PIM is used to maintain forwarding tables to forward multicast datagrams.

To enable multicast routing on the management interface, you must explicitly set a multicast boundary on the management interface.



Note Only the UDP transport layer is supported for multicast routing.

The following list shows the maximum number of entries for specific multicast tables. Once these limits are reached, any new entries are discarded.

- MFIB—30,000
- IGMP Groups—30,000
- PIM Routes—72,000

Procedure

Enable multicast routing:

multicast-routing

Example:

```
ciscoasa(config)# multicast-routing
```

The number of entries in the multicast routing tables are limited by the amount of RAM on the ASA.

Customize Multicast Routing

This section describes how to customize multicast routing.

Configure Stub Multicast Routing and Forward IGMP Messages



Note Stub multicast routing is not supported concurrently with PIM sparse and bidirectional modes.

An ASA acting as the gateway to the stub area does not need to participate in PIM sparse mode or bidirectional mode. Instead, you can configure it to act as an IGMP proxy agent and forward IGMP messages from hosts connected on one interface to an upstream multicast router on another interface. To configure the ASA as an IGMP proxy agent, forward the host join and leave messages from the stub area interface to an upstream interface. You must also enable PIM on the interfaces participating in stub mode multicast routing.

Procedure

Configure stub multicast routing and forward IGMP messages:

igmp forward interface if_name

Example:

```
ciscoasa(config-if)# igmp forward interface interface1
```

Configure a Static Multicast Route

Configuring static multicast routes lets you separate multicast traffic from unicast traffic. For example, when a path between a source and destination does not support multicast routing, the solution is to configure two multicast devices with a GRE tunnel between them and to send the multicast packets over the tunnel.

When using PIM, the ASA expects to receive packets on the same interface where it sends unicast packets back to the source. In some cases, such as bypassing a route that does not support multicast routing, you may want unicast packets to take one path and multicast packets to take another.

Static multicast routes are not advertised or redistributed.

Procedure

Step 1 Configure a static multicast route:

mroute src_ip src_mask {input_if_name | *rpf_neighbor*} [distance]

Example:

```
ciscoasa(config)# mroute src_ip src_mask {input_if_name | rpf_neighbor} [distance]
```

Step 2 Configure a static multicast route for a stub area:

mroute src_ip src_mask input_if_name [**dense** output_if_name] [distance]

Example:

```
ciscoasa(config)# mroute src_ip src_mask input_if_name [dense output_if_name] [distance]
```

The **dense** *output_if_name* keyword and argument pair is only supported for stub multicast routing.

Configure IGMP Features

IP hosts use IGMP to report their group memberships to directly-connected multicast routers. IGMP is used to dynamically register individual hosts in a multicast group on a particular LAN. Hosts identify group memberships by sending IGMP messages to their local multicast router. Under IGMP, routers listen to IGMP messages and periodically send out queries to discover which groups are active or inactive on a particular subnet.

This section describes how to configure optional IGMP setting on a per-interface basis.

Disable IGMP on an Interface

You can disable IGMP on specific interfaces. This information is useful if you know that there are no multicast hosts on a specific interface and you want to prevent the ASA from sending host query messages on that interface.

Procedure

Disable IGMP on an interface:

no igmp

Example:

```
ciscoasa(config-if)# no igmp
```

To reenable IGMP on an interface, use the **igmp** command.

Note
Only the **no igmp** command appears in the interface configuration.

Configure IGMP Group Membership

You can configure the ASA to be a member of a multicast group. Configuring the ASA to join a multicast group causes upstream routers to maintain multicast routing table information for that group and keep the paths for that group active.

Note If you want to forward multicast packets for a specific group to an interface without the ASA accepting those packets as part of the group, see Configure a Statically Joined IGMP Group, on page 1079.

Procedure

Configure the ASA to be a member of a multicast group:

igmp join-group group-address

Example:

```
ciscoasa(config-if)# igmp join-group mcast-group
```

The *group-address* argument is the IP address of the group.

Note
The **igmp join-group** command enables PIM to send *Join* requests towards the sources or towards the Rendezvous Point (RP), provided, the firewall with this command is the PIM Designated Router (DR) on that interface where the command is applied.

Configure a Statically Joined IGMP Group

Sometimes a group member cannot report its membership in the group because of some configuration, or there may be no members of a group on the network segment. However, you still want multicast traffic for that group to be sent to that network segment. You can have multicast traffic for that group sent to the segment by configuring a statically joined IGMP group.

Enter the **igmp static-group** command. The ASA does not accept the multicast packets, but instead forwards them to the specified interface.

Procedure

Configure the ASA statically to join a multicast group on an interface:

igmp static-group

Example:

```
ciscoasa(config-if)# igmp static-group group-address
```

The *group-address* argument is the IP address of the group.

Note
The **igmp static-group** command enables PIM to send *Join* requests towards the sources or towards the Rendezvous Point (RP), provided, the firewall with this command is the PIM Designated Router (DR) on that interface where the command is applied.

Control Access to Multicast Groups

You can control access to multicast groups by using access control lists.

Procedure

Step 1 Create a standard ACL for the multicast traffic:

access-list name **standard** [**permit | deny]** ip_addr mask

Example:

```
ciscoasa(config)# access-list acl1 standard permit 192.52.662.25
```

You can create more than one entry for a single ACL. You can use extended or standard ACLs.

The *ip_addr mask* argument is the IP address of the multicast group being permitted or denied.

Step 2 Create an extended ACL:'

access-list name **extended** [**permit | deny]** protocol src_ip_addr src_mask dst_ip_addr dst_mask

Example:

```
ciscoasa(config)# access-list acl2 extended permit protocol
src_ip_addr src_mask dst_ip_addr dst_mask
```

The *dst_ip_addr* argument is the IP address of the multicast group being permitted or denied.

Step 3 Apply the ACL to an interface:

igmp access-group acl

Example:

```
ciscoasa(config-if)# igmp access-group acl
```

The *acl* argument is the name of a standard or extended IP ACL.

Limit the Number of IGMP States on an Interface

You can limit the number of IGMP states resulting from IGMP membership reports on a per-interface basis. Membership reports exceeding the configured limits are not entered in the IGMP cache, and traffic for the excess membership reports is not forwarded.

Procedure

Limit the number of IGMP states on an interface:

igmp limit number

Example:

```
ciscoasa(config-if)# igmp limit 50
```

Valid values range from 0 to 5000, with being the default value.

Setting this value to 0 prevents learned groups from being added, but manually defined memberships (using the **igmp join-group** and **igmp static-group** commands) are still permitted. The **no** form of this command restores the default value.

Note When you change the IGMP limit on the interface with active joins on it, the new limit is not applicable to the existing groups. ASA validates the limit only when a new group is added to the interface or when the IGMP join timers expire. To apply the new limit with immediate effect, you must disenable and re-enable IGMP on the interface.

Modify the Query Messages to Multicast Groups

The ASA sends query messages to discover which multicast groups have members on the networks attached to the interfaces. Members respond with IGMP report messages indicating that they want to receive multicast packets for specific groups. Query messages are addressed to the all-systems multicast group, which has an address of 224.0.0.1, with a time-to-live value of 1.

These messages are sent periodically to refresh the membership information stored on the ASA. If the ASA discovers that there are no local members of a multicast group still attached to an interface, it stops forwarding multicast packet for that group to the attached network, and it sends a prune message back to the source of the packets.

By default, the PIM designated router on the subnet is responsible for sending the query messages. By default, they are sent once every 125 seconds.

When changing the query response time, by default, the maximum query response time advertised in IGMP queries is 10 seconds. If the ASA does not receive a response to a host query within this amount of time, it deletes the group.

Note The **igmp query-timeout** and **igmp query-interval** commands require IGMP Version 2.

To change the query interval, query response time, and query timeout value, perform the following steps:

Procedure

Step 1 Set the query interval time in seconds:

igmp query-interval *seconds*

Example:

```
ciscoasa(config-if)# igmp query-interval 30
```

Valid values range from 1 to 3600; 125 is the default value.

If the ASA does not hear a query message on an interface for the specified timeout value (by default, 255 seconds), then the ASA becomes the designated router and starts sending the query messages.

Step 2 Change the timeout value of the query:

igmp query-timeout *seconds*

Example:

```
ciscoasa(config-if)# igmp query-timeout 30
```

Valid values range from 60 to 300; 225 is the default value.

Step 3 Change the maximum query response time:

igmp query-max-response-time *seconds*

Valid values range from 1 to 25; 10 is the default value.

Example:

```
ciscoasa(config-if)# igmp query-max-response-time 20
```

Change the IGMP Version

By default, the ASA runs IGMP Version 2, which enables several additional features such as the **igmp query-timeout** and **igmp query-interval** commands.

All multicast routers on a subnet must support the same version of IGMP. The ASA does not automatically detect Version 1 routers and switch to Version 1. However, a mix of IGMP Version 1 and 2 hosts on the subnet works; the ASA running IGMP Version 2 works correctly when IGMP Version 1 hosts are present.

Procedure

Control the version of IGMP that you want to run on the interface:

igmp version {**1** | **2**}

Example:

```
ciscoasa(config-if)# igmp version 2
```

Configure PIM Features

Routers use PIM to maintain forwarding tables to use for forwarding multicast diagrams. When you enable multicast routing on the ASA, PIM and IGMP are automatically enabled on all interfaces.

Note PIM is not supported with PAT. The PIM protocol does not use ports, and PAT only works with protocols that use ports.

This section describes how to configure optional PIM settings.

Enable and Disable PIM on an Interface

You can enable or disable PIM on specific interfaces.

Procedure

Step 1 Enable or reenable PIM on a specific interface:

pim

Example:

```
ciscoasa(config-if)# pim
```

Step 2 Disable PIM on a specific interface:

no pim

Example:

```
ciscoasa(config-if)# no pim
```

Note
Only the **no pim** command appears in the interface configuration.

Configure a Static Rendezvous Point Address

All routers within a common PIM sparse mode or bidir domain require knowledge of the PIM RP address. The address is statically configured using the **pim rp-address** command.

Note The ASA does not support Auto-RP. You must use the **pim rp-address** command to specify the RP address.

You can configure the ASA to serve as RP to more than one group. The group range specified in the ACL determines the PIM RP group mapping. If an ACL is not specified, then the RP for the group is applied to the entire multicast group range (224.0.0.0/4).

Procedure

Enable or reenable PIM on a specific interface:

pim rp-address ip_address [acl] [**bidir**]

The *ip_address* argument is the unicast IP address of the router assigned to be a PIM RP.

The *acl* argument is the name or number of a standard ACL that defines with which multicast groups the RP should be used. Do not use a host ACL with this command.

Excluding the **bidir** keyword causes the groups to operate in PIM sparse mode.

Note
The ASA always advertises the bidirectional capability in the PIM hello messages, regardless of the actual bidirectional configuration.

Example:

```
ciscoasa(config)# pim rp-address 10.86.75.23 [acl1] [bidir]
```

Configure the Designated Router Priority

The DR is responsible for sending PIM register, join, and prune messages to the RP. When there is more than one multicast router on a network segment, selecting the DR is based on the DR priority. If multiple devices have the same DR priority, then the device with the highest IP address becomes the DR.

By default, the ASA has a DR priority of 1. You can change this value.

Procedure

Change the designated router priority:

pim dr-priority num

Example:

```
ciscoasa(config-if)# pim dr-priority 500
```

The *num* argument can be any number ranging from 1 to 4294967294.

Configure and Filter PIM Register Messages

When the ASA is acting as an RP, you can restrict specific multicast sources from registering with it to prevent unauthorized sources from registering with the RP. The Request Filter pane lets you define the multicast sources from which the ASA will accept PIM register messages.

Procedure

Configure the ASA to filter PIM register messages:

pim accept-register {**list** acl | **route-map** map-name}

Example:

```
ciscoasa(config)# pim accept-register {list acl1 | route-map map2}
```

In the example, the ASA filters PIM register messages *acl1* and route map *map2*.

Configure PIM Message Intervals

Router query messages are used to select the PIM DR. The PIM DR is responsible for sending router query messages. By default, router query messages are sent every 30 seconds. Additionally, every 60 seconds, the ASA sends PIM join or prune messages.

Procedure

Step 1 Send router query messages:

pim hello-interval seconds

Example:

```
ciscoasa(config-if)# pim hello-interval 60
```

Valid values for the *seconds* argument range from 1 to 3600 seconds.

Step 2 Change the amount of time (in seconds) that the ASA sends PIM join or prune messages:

pim join-prune-interval seconds

Example:

```
ciscoasa(config-if)# pim join-prune-interval 60
```

Valid values for the *seconds* argument range from 10 to 600 seconds.

Filter PIM Neighbors

You can define the routers that can become PIM neighbors. By filtering the routers that can become PIM neighbors, you can do the following:

- Prevent unauthorized routers from becoming PIM neighbors.

- Prevent attached stub routers from participating in PIM.

Procedure

Step 1 Use a standard ACL to define the routers that you want to have participate in PIM:

access-list pim_nbr deny *router-IP_addr PIM neighbor*

Example:

```
ciscoasa(config)# access-list pim_nbr deny 10.1.1.1 255.255.255.255
```

In the example, the following ACL, when used with the **pim neighbor-filter** command, prevents the 10.1.1.1 router from becoming a PIM neighbor.

Step 2 Filter neighbor routers:

pim neighbor-filter pim_nbr

Example:

```
ciscoasa(config)# interface GigabitEthernet0/3
ciscoasa(config-if)# pim neighbor-filter pim_nbr
```

In the example, the 10.1.1.1 router is prevented from becoming a PIM neighbor on interface GigabitEthernet0/3.

Configure a Bidirectional Neighbor Filter

The Bidirectional Neighbor Filter pane shows the PIM bidirectional neighbor filters, if any, that are configured on the ASA. A PIM bidirectional neighbor filter is an ACL that defines the neighbor devices that can participate in the DF election. If a PIM bidirectional neighbor filter is not configured for an interface, then there are no restrictions. If a PIM bidirectional neighbor filter is configured, only those neighbors permitted by the ACL can participate in the DF election process.

When a PIM bidirectional neighbor filter configuration is applied to the ASA, an ACL appears in the running configuration with the name *interface-name*_multicast, in which the *interface-name* is the name of the interface to which the multicast boundary filter is applied. If an ACL with that name already exists, a number is appended to the name (for example, inside_multicast_1). This ACL defines which devices can become PIM neighbors of the ASA.

Bidirectional PIM allows multicast routers to keep reduced state information. All of the multicast routers in a segment must be bidirectionally enabled for bidir to elect a DF.

The PIM bidirectional neighbor filters enable the transition from a sparse-mode-only network to a bidir network by letting you specify the routers that should participate in the DF election, while still allowing all routers to participate in the sparse-mode domain. The bidir-enabled routers can elect a DF from among themselves, even when there are non-bidir routers on the segment. Multicast boundaries on the non-bidir routers prevent PIM messages and data from the bidir groups from leaking in or out of the bidir subset cloud.

When a PIM bidirectional neighbor filter is enabled, the routers that are permitted by the ACL are considered to be bidirectionally capable. Therefore, the following is true:

- If a permitted neighbor does not support bidir, then the DF election does not occur.
- If a denied neighbor supports bidir, then the DF election does not occur.
- If a denied neighbor does not support bidir, the DF election can occur.

Procedure

Step 1 Use a standard ACL to define the routers that you want to have participate in PIM:

access-list pim_nbr deny *router-IP_addr PIM neighbor*

Example:

```
ciscoasa(config)# access-list pim_nbr deny 10.1.1.1 255.255.255.255
```

In the example, the following ACL, when used with the **pim neighbor-filter** command, prevents the 10.1.1.1 router from becoming a PIM neighbor.

Step 2 Filter neighbor routers:

pim bidirectional-neighbor-filter pim_nbr

Example:

```
ciscoasa(config)# interface GigabitEthernet0/3
ciscoasa(config-if)# pim bidirectional neighbor-filter pim_nbr
```

In the example, the 10.1.1.1 router is prevented from becoming a PIM bidirectional neighbor on interface GigabitEthernet0/3.

Configure the ASA as a Candidate BSR

You can configure the ASA as a candidate BSR. It also helps in the auto discovery of the RP.

Procedure

Step 1 Configure the router to announce its candidacy as a bootstrap router (BSR):

pim bsr-candidate *interface_name* [*hash_mask_length* [*priority*]]

Example:

```
ciscoasa(config)# pim bsr-candidate inside 12 3
```

Step 2 (Optional) Configure the ASA as a Border Bootstrap Router:

interface *interface_name*

pim bsr-border

Example:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# pim bsr-border
```

When this command is configured on an interface, no bootstrap router (BSR) messages will be sent or received through an interface.

Configure a Multicast Boundary

Address scoping defines domain boundaries so that domains with RPs that have the same IP address do not leak into each other. Scoping is performed on the subnet boundaries within large domains and on the boundaries between the domain and the Internet.

You can set up an administratively scoped boundary on an interface for multicast group addresses. IANA has designated the multicast address range from 239.0.0.0 to 239.255.255.255 as the administratively scoped addresses. This range of addresses can be reused in domains administered by different organizations. The addresses would be considered local, not globally unique.

A standard ACL defines the range of affected addresses. When a boundary is set up, no multicast data packets are allowed to flow across the boundary from either direction. The boundary allows the same multicast group address to be reused in different administrative domains.

You can configure, examine, and filter Auto-RP discovery and announcement messages at the administratively scoped boundary by entering the **filter-autorp** keyword. Any Auto-RP group range announcements from the Auto-RP packets that are denied by the boundary ACL are removed. An Auto-RP group range announcement is permitted and passed by the boundary only if all addresses in the Auto-RP group range are permitted by the boundary ACL. If any address is not permitted, the entire group range is filtered and removed from the Auto-RP message before the Auto-RP message is forwarded.

Procedure

Configure a multicast boundary:

multicast boundary acl [**filter-autorp**]

Example:

```
ciscoasa(config-if)# multicast boundary acl1 [filter-autorp]
```

Monitoring for PIM

You can use the following commands to monitor the PIM routing process. For examples and descriptions of the command output, see the command reference.

To monitor or disable various PIM routing statistics, enter one of the following commands:

- **show pim bsr-router**

  Displays the bootstrap router information.

- **show mroute**

  Displays the contents of the multicast routing table.

- **show mfib summary**

  Displays summary information about the number of IPv4 PIM multicast forwarding information base entries and interfaces.

- **show mfib active**

  Displays information from the Multicast Forwarding Information Base (MFIB) about the rate at which active multicast sources are sending to multicast groups.

- **show pim group-map**

  Displays the group-to-PIM mode mapping. To display the elected RP for a group, specify the group address or name.

- **show pim group-map rp-timers**

  Displays the timer expiry and uptime for each group to PIM mode mapping entry.

- **show pim neighbor**

  Displays the Protocol Independent Multicast (PIM) neighbors.

Example for Multicast Routing

The following example shows how to enable and configure multicast routing with various optional processes:

1. Enable multicast routing:

```
ciscoasa(config)# multicast-routing
```

2. Configure a static multicast route:

```
ciscoasa(config)# mroute src_ip src_mask {input_if_name | rpf_neighbor} [distance]
ciscoasa(config)# exit
```

3. Configure the ASA to be a member of a multicast group:

```
ciscoasa(config)# interface
ciscoasa(config-if)# igmp join-group group-address
```

History for Multicast Routing

Table 42: Feature History for Multicast Routing

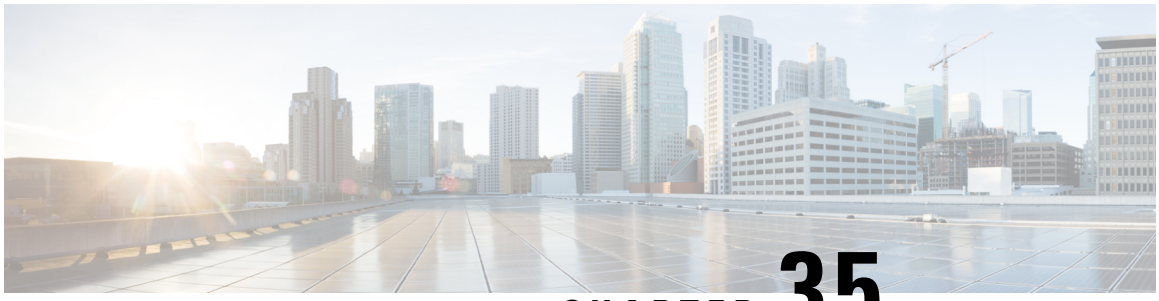
Feature Name	Platform Releases	Feature Information
Multicast routing over an Static Virtual Tunnel Interface tunnel	9.24(10)	You can securely deliver real-time multicast applications, such as video streaming, financial data feeds, and data replication over your VPN infrastructure by using IPsec tunnels configured with static VTI.
Multicast routing support	7.0(1)	Support was added for multicast routing data, authentication, and redistribution and monitoring of routing information using the multicast routing protocol. We introduced the **multicast-routing** command.
Clustering support	9.0(1)	Support was added for clustering. We introduced the following commands: **debug mfib cluster**, **show mfib cluster**.
Protocol Independent Multicast Source-Specific Multicast (PIM-SSM) pass-through support	9.5(1)	Support was added to allow PIM-SSM packets to pass through when multicast routing is enabled, unless the ASA is the Last-Hop Router. This allows greater flexibility in choosing a multicast group while also protecting against different attacks; hosts only receive traffic from explicitly-requested sources. We did not change any commands.
Protocol Independent Multicast Bootstrap Router(BSR)	9.5(2)	Support was added for a new dynamic Rendezvous Point (RP) selection model that uses candidate routers for Rendezvous Point function and for relaying the Rendezvous Point information for a group. This feature provides a means of dynamically learning Rendezvous Points (RPs), which is very essential in large complex networks where an RP can periodically go down and come up. We introduced the following commands: clear pim group-map, debug pim bsr, pim bsr-border, pim bsr-candidate, show pim bsr-router, show pim group-map rp-timers
igmp limit increased	9.15(1) *Also in 9.12(4)*	igmp limit increased from 500 to 5000. New/Modified commands: **igmp limit**.



PART VI

AAA Servers and the Local Database

- AAA and the Local Database, on page 1093
- RADIUS Servers for AAA, on page 1105
- TACACS+ Servers for AAA, on page 1125
- LDAP Servers for AAA, on page 1133
- Kerberos Servers for AAA, on page 1145
- RSA SecurID Servers for AAA, on page 1153



CHAPTER 35

AAA and the Local Database

This chapter describes authentication, authorization, and accounting (AAA, pronounced “triple A”). AAA is a a set of services for controlling access to computer resources, enforcing policies, assessing usage, and providing the information necessary to bill for services. These processes are considered important for effective network management and security.

This chapter also describes how to configure the local database for AAA functionality. For external AAA servers, see the chapter for your server type.

- About AAA and the Local Database, on page 1093
- Guidelines for the Local Database, on page 1098
- Add a User Account to the Local Database, on page 1098
- Monitoring the Local Database, on page 1100
- History for the Local Database, on page 1101

About AAA and the Local Database

This section describes AAA and the local database.

Authentication

Authentication provides a way to identify a user, typically by having the user enter a valid username and valid password before access is granted. The AAA server compares a user's authentication credentials with other user credentials stored in a database. If the credentials match, the user is permitted access to the network. If the credentials do not match, authentication fails and network access is denied.

You can configure the ASA to authenticate the following items:

- All administrative connections to the ASA, including the following sessions:
  - Telnet
  - SSH
  - Serial console
  - ASDM using HTTPS
  - VPN management access

- The **enable** command
- Network access
- VPN access

Authorization

Authorization is the process of enforcing policies: determining what types of activities, resources, or services a user is permitted to access. After a user is authenticated, that user may be authorized for different types of access or activity.

You can configure the ASA to authorize the following items:

- Management commands
- Network access
- VPN access

Accounting

Accounting measures the resources a user consumes during access, which may include the amount of system time or the amount of data that a user has sent or received during a session. Accounting is carried out through the logging of session statistics and usage information, which is used for authorization control, billing, trend analysis, resource utilization, and capacity planning activities.

Interaction Between Authentication, Authorization, and Accounting

You can use authentication alone or with authorization and accounting. Authorization always requires a user to be authenticated first. You can use accounting alone, or with authentication and authorization.

AAA Servers and Server Groups

The AAA server is a network server that is used for access control. Authentication identifies the user. Authorization implements policies that determine which resources and services an authenticated user may access. Accounting keeps track of time and data resources that are used for billing and analysis.

If you want to use an external AAA server, you must first create a AAA server group for the protocol that the external server uses, and add the server to the group. You can create more than one group per protocol, and separate groups for all protocols that you want to use. Each server group is specific to one type of server or service.

See the following topics for details on how to create the groups:

- Configure RADIUS Server Groups, on page 1117
- Configure TACACS+ Server Groups, on page 1127
- Configure LDAP Server Groups, on page 1139
- Configure Kerberos AAA Server Groups, on page 1145

- Configure RSA SecurID AAA Server Groups, on page 1154

See the VPN configuration guide for more information on using Kerberos Constrained Delegation and HTTP Form.

The following table summarizes the supported types of server and their uses, including the local database.

Table 43: Supported Services for AAA Servers

Server Type and Service	Authentication	Authorization	Accounting
Local Database			
Administrators	Yes	Yes	No
VPN Users	Yes	No	No
Firewall Sessions (AAA rules)	Yes	Yes	No
RADIUS			
Administrators	Yes	Yes	Yes
VPN Users	Yes	Yes	Yes
Firewall Sessions (AAA rules)	Yes	Yes	Yes
TACACS+			
Administrators	Yes	Yes	Yes
VPN Users	Yes	No	Yes
Firewall Sessions (AAA rules)	Yes	Yes	Yes
LDAP			
Administrators	Yes	No	No
VPN Users	Yes	Yes	No
Firewall Sessions (AAA rules)	Yes	No	No
Kerberos			
Administrators	Yes	No	No
VPN Users	Yes	No	No
Firewall Sessions (AAA rules)	Yes	No	No
SDI (RSA SecurID)			
Administrators	Yes	No	No
VPN Users	Yes	No	No

Server Type and Service	Authentication	Authorization	Accounting
Firewall Sessions (AAA rules)	Yes	No	No
HTTP Form			
Administrators	No	No	No
VPN Users	Yes	No	No
Firewall Sessions (AAA rules)	No	No	No

Notes

- RADIUS—Accounting for administrators does not include command accounting.
- RADIUS—Authorization for firewall sessions is supported with user-specific access lists only, which are received or specified in a RADIUS authentication response.
- TACACS+—Accounting for administrators includes command accounting.
- HTTP Form—Authentication and SSO operations for clientless SSL VPN user sessions only.

About the Local Database

The ASA maintains a local database that you can populate with user profiles. You can use a local database instead of AAA servers to provide user authentication, authorization, and accounting.

You can use the local database for the following functions:

- ASDM per-user access
- Console authentication
- Telnet and SSH authentication
- **enable** command authentication

  This setting is for CLI-access only and does not affect the Cisco ASDM login.
- Command authorization

  If you turn on command authorization using the local database, then the ASA refers to the user privilege level to determine which commands are available. Otherwise, the privilege level is not generally used. By default, all commands are either privilege level 0 or level 15.
- Network access authentication
- VPN client authentication

For multiple context mode, you can configure usernames in the system execution space to provide individual logins at the CLI using the **login** command; however, you cannot configure any AAA rules that use the local database in the system execution space.

Note You cannot use the local database for network access authorization.

Fallback Support

The local database can act as a fallback method for several functions. This behavior is designed to help you prevent accidental lockout from the ASA.

When a user logs in, the servers in the group are accessed one at a time, starting with the first server that you specify in the configuration, until a server responds. If all servers in the group are unavailable, the ASA tries the local database if you have configured it as a fallback method (for management authentication and authorization only). If you do not have a fallback method, the ASA continues to try the AAA servers.

For users who need fallback support, we recommend that their usernames and passwords in the local database match their usernames and passwords on the AAA servers. This practice provides transparent fallback support. Because the user cannot determine whether a AAA server or the local database is providing the service, using usernames and passwords on AAA servers that are different than the usernames and passwords in the local database means that the user cannot be certain which username and password should be given.

The local database supports the following fallback functions:

- Console and enable password authentication—If the servers in the group are all unavailable, the ASA uses the local database to authenticate administrative access, which can also include enable password authentication.
- Command authorization—If the TACACS+ servers in the group are all unavailable, the local database is used to authorize commands based on privilege levels.
- VPN authentication and authorization—VPN authentication and authorization are supported to enable remote access to the ASA if AAA servers that normally support these VPN services are unavailable. When a VPN client of an administrator specifies a tunnel group configured to fallback to the local database, the VPN tunnel can be established even if the AAA server group is unavailable, provided that the local database is configured with the necessary attributes.

How Fallback Works with Multiple Servers in a Group

If you configure multiple servers in a server group and you enable fallback to the local database for the server group, fallback occurs when no server in the group responds to the authentication request from the ASA. To illustrate, consider this scenario:

You configure an LDAP server group with two Active Directory servers, server 1 and server 2, in that order. When the remote user logs in, the ASA attempts to authenticate to server 1.

If server 1 responds with an authentication failure (such as user not found), the ASA does not attempt to authenticate to server 2.

If server 1 does not respond within the timeout period (or the number of authentication attempts exceeds the configured maximum), the ASA tries server 2.

If both servers in the group do not respond, and the ASA is configured to fall back to the local database, the ASA tries to authenticate to the local database.

Note A RADIUS client, by default, sends three requests before timing out. When the client attempts to SSH, the request is sent to one server at a time. As the requests time out, the server is marked as FAILED. The next RADIUS request goes to the next ACTIVE server, subsequently marking all the servers as FAILED, and eventually falling back to LOCAL authentication (if configured). The servers marked as FAILED eventually recover and are made ACTIVE. In this case, if you have 3-4 servers in the group, you might observe intermittent login issues.

Guidelines for the Local Database

Make sure that you prevent a lockout from the ASA when using the local database for authentication or authorization.

Add a User Account to the Local Database

To add a user to the local database, perform the following steps:

Procedure

Step 1 Create the user account.

username *username* [**password** *password*] [**privilege** *priv_level*]

Example:

```
ciscoasa(config)# username exampleuser1 password madmaxfuryroadrules privilege 1
```

The**username** *username* keyword is a string from 3 to 64 characters long, using any combination of ASCII printable characters (character codes 32-126), with the exception of spaces and the question mark. The **password** *password* keyword is a string from 8 to 127 characters long, and can be any combination of ASCII printable characters (character codes 32-126), with the following exceptions:

- No spaces
- No question marks
- You cannot use three or more consecutive sequential or repetitive ASCII characters. For example, the following passwords will be rejected:
  - **abc**user1
  - user**543**
  - user**aaaa**
  - user2**666**

You might want to create a username without a password if you are using SSH public key authentication, for example. The **privilege** *priv_level* keyword sets the privilege level, which ranges from 0 to 15. The default is 2. This privilege level is used with command authorization.

Caution
If you do not use command authorization (the **aaa authorization console LOCAL** command), then the default level 2 allows management access to privileged EXEC mode. If you want to limit access to privileged EXEC mode, either set the privilege level to 0 or 1, or use the **service-type** command.

These less-used options are not shown in the above syntax: The **nopassword** keyword creates a user account without a password; this option is insecure, is not recommended, and is not supported with all types of connections.

The **encrypted** keyword (for passwords 32 characters and fewer in 9.6 and earlier) or the **pbkdf2** keyword (for passwords longer than 32 characters in 9.6 and later, and passwords of all lengths in 9.7 and later) indicates that the password is encrypted (using an MD5-based hash or a PBKDF2 (Password-Based Key Derivation Function 2) hash). Note that already existing passwords continue to use the MD5-based hash unless you enter a new password. When you define a password in the **username** command, the ASA encrypts it when it saves it to the configuration for security purposes. When you enter the **show running-config** command, the **username** command does not show the actual password; it shows the encrypted password followed by the **encrypted** or **pbkdf2** keyword. For example, if you enter the password "test," the **show running-config** command output would appear as something similar to the following:

```
username user1 password DLaUiAX3l78qgoB5c7iVNw== encrypted
```

The only time you would actually enter the **encrypted** or **pbkdf2** keyword at the CLI is if you are cutting and pasting a configuration file for use in another ASA, and you are using the same password.

Step 2 (Optional) Configure username attributes.

username *username* **attributes**

Example:

```
ciscoasa(config)# username exampleuser1 attributes
```

The *username* argument is the username that you created in the first step.

By default, VPN users that you add with this command have no attributes or group policy association. You must configure all values explicitly using the **username attributes** command. See the VPN configuration guide for more information.

Step 3 (Optional) Configure the user level if you configured management authorization using the **aaa authorization exec** command.

service-type {**admin** | **nas-prompt** | **remote-access**}

Example:

```
ciscoasa(config-username)# service-type admin
```

The **admin** keyword allows full access to any services specified by the **aaa authentication console LOCAL** commands. The **admin** keyword is the default.

The **nas-prompt** keyword allows access to the CLI when you configure the **aaa authentication** {**telnet** | **ssh** | **serial**} **console** command, but denies ASDM configuration access if you configure the **aaa authentication http console** command. ASDM monitoring access is allowed. If you enable authentication with the **aaa authentication enable console** command, the user cannot access privileged EXEC mode using the **enable** command (or the **login** command).

The **remote-access** keyword denies management access. You cannot use any services specified by the **aaa authentication console** commands (excluding the **serial** keyword; serial access is allowed).

Step 4 (Optional) For public key authentication for SSH connections to the ASA on a per-user basis, see Configure SSH for Public Key Access, on page 1166.

Step 5 (Optional) If you are using this username for VPN authentication, you can configure many VPN attributes for the user. See the VPN configuration guide for more information.

Examples

The following example assigns a privilege level of 15 to the admin user account:

```
ciscoasa(config)# username admin password farscape1 privilege 15
```

The following example enables management authorization, creates a user account with a password, enters username configuration mode, and specifies a **service-type** of **nas-prompt**:

```
ciscoasa(config)# aaa authorization exec authentication-server
ciscoasa(config)# username user1 password gOrgeOus
ciscoasa(config)# username user1 attributes
ciscoasa(config-username)# service-type nas-prompt
```

Monitoring the Local Database

See the following commands for monitoring the local database:

- **show aaa-server**

  This command shows the configured database statistics. To clear the AAA server statistics, enter the **clear aaa-server statistics** command.

- **show running-config aaa-server**

  This command shows the AAA server running configuration. To clear AAA server configuration, enter the **clear configure aaa-server** command.

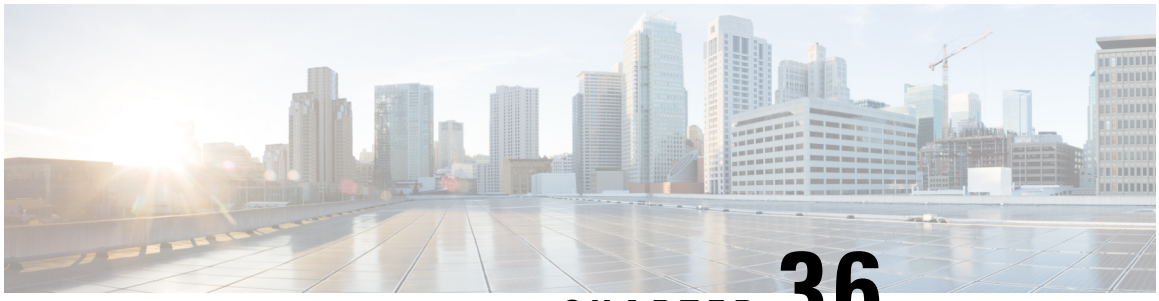
History for the Local Database

Table 44: History for the Local Database

Feature Name	Platform Releases	Description
Local database configuration for AAA	7.0(1)	Describes how to configure the local database for AAA use. We introduced the following commands: **username**, **aaa authorization exec authentication-server**, **aaa authentication console LOCAL**, **aaa authorization exec LOCAL**, **service-type, aaa authentication** {**telnet | ssh | serial**} **console LOCAL**, **aaa authentication http console LOCAL**, **aaa authentication enable console LOCAL**, **show running-config aaa-server**, **show aaa-server**, **clear configure aaa-server**, **clear aaa-server statistics**.
Support for SSH public key authentication	9.1(2)	You can now enable public key authentication for SSH connections to the ASA on a per-user basis. You can specify a public key file (PKF) formatted key or a Base64 key. The PKF key can be up to 4096 bits. Use PKF format for keys that are too large to for the ASA support of the Base64 format (up to 2048 bits). We introduced the following commands: **ssh authentication**. *Also available in 8.4(4.1); PKF key format support is only in 9.1(2).*
Longer password support for local **username** and **enable** passwords (up to 127 characters)	9.6(1)	You can now create local **username** and **enable** passwords up to 127 characters (the former limit was 32). When you create a password longer than 32 characters, it is stored in the configuration using a PBKDF2 (Password-Based Key Derivation Function 2) hash. Shorter passwords continue to use the MD5-based hashing method. We modified the following commands: **enable, username**
SSH public key authentication improvements	9.6(2)	In earlier releases, you could enable SSH public key authentication (**ssh authentication**) without also enabling AAA SSH authentication with the Local user database (**aaa authentication ssh console LOCAL**). The configuration is now fixed so that you must explicitly enable AAA SSH authentication. To disallow users from using a password instead of the private key, you can now create a username without any password defined. We modified the following commands: **ssh authentication, username**

Feature Name	Platform Releases	Description
PBKDF2 hashing for all local **username** and **enable** passwords	9.7(1)	Local **username** and **enable** passwords of all lengths are stored in the configuration using a PBKDF2 (Password-Based Key Derivation Function 2) hash. Previously, passwords 32 characters and shorter used the MD5-based hashing method. Already existing passwords continue to use the MD5-based hash unless you enter a new password. See the "Software and Configurations" chapter in the General Operations Configuration Guide for downgrading guidelines. We modified the following commands: **enable, username**
Separate authentication for users with SSH public key authentication and users with passwords	9.6(3)/9.8(1)	In releases prior to 9.6(2), you could enable SSH public key authentication (**ssh authentication**) without also explicitly enabling AAA SSH authentication with the Local user database (**aaa authentication ssh console LOCAL**). In 9.6(2), the ASA required you to explicitly enable AAA SSH authentication. In this release, you no longer have to explicitly enable AAA SSH authentication; when you configure the **ssh authentication** command for a user, local authentication is enabled by default for users with this type of authentication. Moreover, when you explicitly configure AAA SSH authentication, this configuration only applies for for usernames with *passwords*, and you can use any AAA server type (**aaa authentication ssh console radius_1**, for example). For example, some users can use public key authentication using the local database, and other users can use passwords with RADIUS. We did not modify any commands.
Stronger local user and enable password requirements	9.17(1)	For local users and the enable password, the following password requirements were added: • Password length—Minimum 8 characters. Formerly, the minimum was 3 characters. • Repetitive and sequential characters—Three or more consecutive sequential or repetitive ASCII characters are disallowed. For example, the following passwords will be rejected: • **abc**user1 • user**543** • user**aaaa** • user2**666** New/Modified commands: **enable password**, **username**

Feature Name	Platform Releases	Description
Local user lockout changes	9.17(1)	The ASA can lock out local users after a configurable number of failed login attempts. This feature did not apply to users with privilege level 15. Also, a user would be locked out indefinitely until an admin unlocked their account. Now, users will be unlocked after 10 minutes unless an admin uses the **clear aaa local user lockout** command before then. Privilege level 15 users are also now affected by the lockout setting. New/Modified commands: **aaa local authentication attempts max-fail** , **show aaa local user**
SSH and Telnet password change prompt	9.17(1)	The first time a local user logs into the ASA using SSH or Telnet, they are prompted to change their password. They will also be prompted for the first login after an admin changes their password. If the ASA reloads, however, users will not be prompted even if it is their first login. New/Modified commands: **show aaa local user**



CHAPTER 36

RADIUS Servers for AAA

This chapter describes how to configure RADIUS servers for AAA.

- About RADIUS Servers for AAA, on page 1105
- Guidelines for RADIUS Servers for AAA, on page 1116
- Configure RADIUS Servers for AAA, on page 1116
- Monitoring RADIUS Servers for AAA, on page 1123
- History for RADIUS Servers for AAA, on page 1124

About RADIUS Servers for AAA

The ASA supports the following RFC-compliant RADIUS servers for AAA:

- Cisco Secure ACS 3.2, 4.0, 4.1, 4.2, and 5.x
- Cisco Identity Services Engine (ISE)
- RSA RADIUS in RSA Authentication Manager 5.2, 6.1, 7.x, 8.x
- Microsoft

Supported Authentication Methods

The ASA supports the following authentication methods with RADIUS servers:

- PAP—For all connection types.
- CHAP and MS-CHAPv1—For L2TP-over-IPsec connections.
- MS-CHAPv2—For L2TP-over-IPsec connections, and for regular IPsec remote access connections when the password management feature is enabled. You can also use MS-CHAPv2 with clientless connections.
- Authentication Proxy modes—For RADIUS-to Active-Directory, RADIUS-to-RSA/SDI, RADIUS-to-Token server, and RSA/SDI-to-RADIUS connections,

Note To enable MS-CHAPv2 as the protocol used between the ASA and the RADIUS server for a VPN connection, password management must be enabled in the tunnel group general attributes. Enabling password management generates an MS-CHAPv2 authentication request from the ASA to the RADIUS server. See the description of the **password-management** command for details.

If you use double authentication and enable password management in the tunnel group, then the primary and secondary authentication requests include MS-CHAPv2 request attributes. If a RADIUS server does not support MS-CHAPv2, then you can configure that server to send a non-MS-CHAPv2 authentication request by using the **no mschapv2-capable** command.

User Authorization of VPN Connections

The ASA can use RADIUS servers for user authorization of VPN remote access and firewall cut-through-proxy sessions using dynamic ACLs or ACL names per user. To implement dynamic ACLs, you must configure the RADIUS server to support them. When the user authenticates, the RADIUS server sends a downloadable ACL or ACL name to the ASA. Access to a given service is either permitted or denied by the ACL. The ASA deletes the ACL when the authentication session expires.

In addition to ACLs, the ASA supports many other attributes for authorization and setting of permissions for VPN remote access and firewall cut-through proxy sessions.

Supported Sets of RADIUS Attributes

The ASA supports the following sets of RADIUS attributes:

- Authentication attributes defined in RFC 2138 and 2865.
- Accounting attributes defined in RFC 2139 and 2866.
- RADIUS attributes for tunneled protocol support, defined in RFC 2868 and 6929.
- Cisco IOS Vendor-Specific Attributes (VSAs), identified by RADIUS vendor ID 9.
- Cisco VPN-related VSAs, identified by RADIUS vendor ID 3076.
- Microsoft VSAs, defined in RFC 2548.

Supported RADIUS Authorization Attributes

Authorization refers to the process of enforcing permissions or attributes. A RADIUS server defined as an authentication server enforces permissions or attributes if they are configured. These attributes have vendor ID 3076.

The following table lists the supported RADIUS attributes that can be used for user authorization.

Note RADIUS attribute names do not contain the cVPN3000 prefix. Cisco Secure ACS 4.x supports this new nomenclature, but attribute names in pre-4.0 ACS releases still include the cVPN3000 prefix. The ASAs enforce the RADIUS attributes based on attribute numeric ID, not attribute name.

All attributes listed in the following table are downstream attributes that are sent from the RADIUS server to the ASA except for the following attribute numbers: 146, 150, 151, and 152. These attribute numbers are upstream attributes that are sent from the ASA to the RADIUS server. RADIUS attributes 146 and 150 are sent from the ASA to the RADIUS server for authentication and authorization requests. All four previously listed attributes are sent from the ASA to the RADIUS server for accounting start, interim-update, and stop requests. Upstream RADIUS attributes 146, 150, 151, and 152 were introduced in Version 8.4(3).

Table 45: Supported RADIUS Authorization Attributes

Attribute Name	ASA	Attr. No.	Syntax/Type	Single or Multi- Valued	Description or Value
Access-Hours	Y	1	String	Single	Name of the time range, for example, Busine
Access-List-Inbound	Y	86	String	Single	ACL ID
Access-List-Outbound	Y	87	String	Single	ACL ID
Address-Pools	Y	217	String	Single	Name of IP local pool
Allow-Network-Extension-Mode	Y	64	Boolean	Single	0 = Disabled 1 = Enabled
Authenticated-User-Idle-Timeout	Y	50	Integer	Single	1-35791394 minutes
Authorization-DN-Field	Y	67	String	Single	Possible values: UID, OU, O, CN, L, SP, C, GN, SN, I, GENQ, DNQ, SER, use-entire-na
Authorization-Required		66	Integer	Single	0 = No 1 = Yes
Authorization-Type	Y	65	Integer	Single	0 = None 1 = RADIUS 2 = LDAP
Banner1	Y	15	String	Single	Banner string to display for Cisco VPN remo sessions: IPsec IKEv1, AnyConnect Client SSL-TLS/DTLS/IKEv2, and Clientless SSL
Banner2	Y	36	String	Single	Banner string to display for Cisco VPN remo sessions: IPsec IKEv1, AnyConnect Client SSL-TLS/DTLS/IKEv2, and Clientless SSL. T string is concatenated to the Banner1 string , if
Cisco-IP-Phone-Bypass	Y	51	Integer	Single	0 = Disabled 1 = Enabled
Cisco-LEAP-Bypass	Y	75	Integer	Single	0 = Disabled 1 = Enabled
Client Type	Y	150	Integer	Single	1 = Cisco VPN Client (IKEv1) 2 = AnyConn SSL VPN 3 = Clientless SSL VPN 4 = Cut-Through-Proxy 5 = L2TP/IPsec SSL VP AnyConnect Client IPsec VPN (IKEv2)

Attribute Name	ASA	Attr. No.	Syntax/Type	Single or Multi- Valued	Description or Value
Client-Type-Version-Limiting	Y	77	String	Single	IPsec VPN version number string
DHCP-Network-Scope	Y	61	String	Single	IP Address
Extended-Authentication-On-Rekey	Y	122	Integer	Single	0 = Disabled 1 = Enabled
Framed-Interface-Id	Y	96	String	Single	Assigned IPv6 interface ID. Combines with Framed-IPv6-Prefix to create a complete assigne address. For example: Framed-Interface-ID=1:1 combined with Framed-IPv6-Prefix=2001:0db8::/ the assigned IP address 2001:0db8::1:1:1:1.
Framed-IPv6-Prefix	Y	97	String	Single	Assigned IPv6 prefix and length. Combines with Framed-Interface-Id to create a complete assigne address. For example: prefix 2001:0db8::/64 cor with Framed-Interface-Id=1:1:1:1 gives the IP a 2001:0db8::1:1:1:1. You can use this attribute to an IP address without using Framed-Interface-Id assigning the full IPv6 address with prefix lengt for example, Framed-IPv6-Prefix=2001:0db8::1
Group-Policy	Y	25	String	Single	Sets the group policy for the remote access VPN For Versions 8.2.x and later, use this attribute in IETF-Radius-Class. You can use one of the follo formats: • *group policy name* • OU=*group policy name* • OU=*group policy name*;
IE-Proxy-Bypass-Local		83	Integer	Single	0 = None 1 = Local
IE-Proxy-Exception-List		82	String	Single	New line (\n) separated list of DNS domains
IE-Proxy-PAC-URL	Y	133	String	Single	PAC address string
IE-Proxy-Server		80	String	Single	IP address
IE-Proxy-Server-Policy		81	Integer	Single	1 = No Modify 2 = No Proxy 3 = Auto detect 4 = Concentrator Setting
IKE-KeepAlive-Confidence-Interval	Y	68	Integer	Single	10-300 seconds
IKE-Keepalive-Retry-Interval	Y	84	Integer	Single	2-10 seconds
IKE-Keep-Alives	Y	41	Boolean	Single	0 = Disabled 1 = Enabled
Intercept-DHCP-Configure-Msg	Y	62	Boolean	Single	0 = Disabled 1 = Enabled
IPsec-Allow-Passwd-Store	Y	16	Boolean	Single	0 = Disabled 1 = Enabled

Attribute Name	ASA	Attr. No.	Syntax/Type	Single or Multi- Valued	Description or Value
IPsec-Authentication		13	Integer	Single	0 = None 1 = RADIUS 2 = LDAP (authoriza = NT Domain 4 = SDI 5 = Internal 6 = RAD Expiry 7 = Kerberos/Active Directory
IPsec-Auth-On-Rekey	Y	42	Boolean	Single	0 = Disabled 1 = Enabled
IPsec-Backup-Server-List	Y	60	String	Single	Server Addresses (space delimited)
IPsec-Backup-Servers	Y	59	String	Single	1 = Use Client-Configured list 2 = Disable and list 3 = Use Backup Server list
IPsec-Client-Firewall-Filter-Name		57	String	Single	Specifies the name of the filter to be pushed as firewall policy
IPsec-Client-Firewall-Filter-Optional	Y	58	Integer	Single	0 = Required 1 = Optional
IPsec-Default-Domain	Y	28	String	Single	Specifies the single default domain name to client (1-255 characters).
IPsec-IKE-Peer-ID-Check	Y	40	Integer	Single	1 = Required 2 = If supported by peer certifi not check
IPsec-IP-Compression	Y	39	Integer	Single	0 = Disabled 1 = Enabled
IPsec-Mode-Config	Y	31	Boolean	Single	0 = Disabled 1 = Enabled
IPsec-Over-UDP	Y	34	Boolean	Single	0 = Disabled 1 = Enabled
IPsec-Over-UDP-Port	Y	35	Integer	Single	4001- 49151. The default is 10000.
IPsec-Required-Client-Firewall-Capability	Y	56	Integer	Single	0 = None 1 = Policy defined by remote FW Are-You-There (AYT) 2 = Policy pushed CP from server
IPsec-Sec-Association		12	String	Single	Name of the security association
IPsec-Split-DNS-Names	Y	29	String	Single	Specifies the list of secondary domain name the client (1-255 characters).
IPsec-Split-Tunneling-Policy	Y	55	Integer	Single	0 = No split tunneling 1 = Split tunneling 2 = permitted
IPsec-Split-Tunnel-List	Y	27	String	Single	Specifies the name of the network or ACL th the split tunnel inclusion list.
IPsec-Tunnel-Type	Y	30	Integer	Single	1 = LAN-to-LAN 2 = Remote access
IPsec-User-Group-Lock		33	Boolean	Single	0 = Disabled 1 = Enabled
IPv6-Address-Pools	Y	218	String	Single	Name of IP local pool-IPv6
IPv6-VPN-Filter	Y	219	String	Single	ACL value

Attribute Name	ASA	Attr. No.	Syntax/Type	Single or Multi- Valued	Description or Value
L2TP-Encryption		21	Integer	Single	Bitmap: 1 = Encryption required 2 = 40 bits 4 = 8 = Stateless-Req 15= 40/128-Encr/Stateless-Re
L2TP-MPPC-Compression		38	Integer	Single	0 = Disabled 1 = Enabled
Member-Of	Y	145	String	Single	Comma-delimited string, for example: `Engineering, Sales` An administrative attribute that can be used in d access policies. It does not set a group policy.
MS-Client-Subnet-Mask	Y	63	Boolean	Single	An IP address
NAC-Default-ACL		92	String		ACL
NAC-Enable		89	Integer	Single	0 = No 1 = Yes
NAC-Revalidation-Timer		91	Integer	Single	300-86400 seconds
NAC-Settings	Y	141	String	Single	Name of the NAC policy
NAC-Status-Query-Timer		90	Integer	Single	30-1800 seconds
Perfect-Forward-Secrecy-Enable	Y	88	Boolean	Single	0 = No 1 = Yes
PPTP-Encryption		20	Integer	Single	Bitmap: 1 = Encryption required 2 = 40 bits 4 = 8 = Stateless-Required 15= 40/128-Encr/Stateles
PPTP-MPPC-Compression		37	Integer	Single	0 = Disabled 1 = Enabled
Primary-DNS	Y	5	String	Single	An IP address
Primary-WINS	Y	7	String	Single	An IP address
Privilege-Level	Y	220	Integer	Single	An integer between 0 and 15.
Required-Client- Firewall-Vendor-Code	Y	45	Integer	Single	1 = Cisco Systems (with Cisco Integrated Client Zone Labs 3 = NetworkICE 4 = Sygate 5 = Cisco (with Cisco Intrusion Prevention Security Agent
Required-Client-Firewall-Description	Y	47	String	Single	String

Attribute Name	ASA	Attr. No.	Syntax/Type	Single or Multi- Valued	Description or Value
Required-Client-Firewall-Product-Code	Y	46	Integer	Single	Cisco Systems Products: 1 = Cisco Intrusion Prevention Security Age Integrated Client (CIC) Zone Labs Products: 1 = Zone Alarm 2 = Zon 3 = Zone Labs Integrity NetworkICE Product: 1 = BlackIce Defende Sygate Products: 1 = Personal Firewall 2 = P Firewall Pro 3 = Security Agent
Required-Individual-User-Auth	Y	49	Integer	Single	0 = Disabled 1 = Enabled
Require-HW-Client-Auth	Y	48	Boolean	Single	0 = Disabled 1 = Enabled
Secondary-DNS	Y	6	String	Single	An IP address
Secondary-WINS	Y	8	String	Single	An IP address
SEP-Card-Assignment		9	Integer	Single	Not used
Session Subtype	Y	152	Integer	Single	0 = None 1 = Clientless 2 = Client 3 = Clien Session Subtype applies only when the Sessi (151) attribute has the following values: 1, 2
Session Type	Y	151	Integer	Single	0 = None 1 = AnyConnect Client SSL VPN AnyConnect Client IPSec VPN (IKEv2) 3 = SSL VPN 4 = Clientless Email Proxy 5 = Cl Client (IKEv1) 6 = IKEv1 LAN-LAN 7 = IK LAN-LAN 8 = VPN Load Balancing
Simultaneous-Logins	Y	2	Integer	Single	0-2147483647
Smart-Tunnel	Y	136	String	Single	Name of a Smart Tunnel
Smart-Tunnel-Auto	Y	138	Integer	Single	0 = Disabled 1 = Enabled 2 = AutoStart
Smart-Tunnel-Auto-Signon-Enable	Y	139	String	Single	Name of a Smart Tunnel Auto Signon list ap the domain name
Strip-Realm	Y	135	Boolean	Single	0 = Disabled 1 = Enabled
SVC-Ask	Y	131	String	Single	0 = Disabled 1 = Enabled 3 = Enable default Enable default clientless (2 and 4 not used)
SVC-Ask-Timeout	Y	132	Integer	Single	5-120 seconds
SVC-DPD-Interval-Client	Y	108	Integer	Single	0 = Off 5-3600 seconds
SVC-DPD-Interval-Gateway	Y	109	Integer	Single	0 = Off) 5-3600 seconds

Attribute Name	ASA	Attr. No.	Syntax/Type	Single or Multi- Valued	Description or Value
SVC-DTLS	Y	123	Integer	Single	0 = False 1 = True
SVC-Keepalive	Y	107	Integer	Single	0 = Off 15-600 seconds
SVC-Modules	Y	127	String	Single	String (name of a module)
SVC-MTU	Y	125	Integer	Single	MTU value 256-1406 in bytes
SVC-Profiles	Y	128	String	Single	String (name of a profile)
SVC-Rekey-Time	Y	110	Integer	Single	0 = Disabled 1-10080 minutes
Tunnel Group Name	Y	146	String	Single	1-253 characters
Tunnel-Group-Lock	Y	85	String	Single	Name of the tunnel group or “none”
Tunneling-Protocols	Y	11	Integer	Single	1 = PPTP 2 = L2TP 4 = IPSec (IKEv1) 8 = L2T 16 = WebVPN 32 = SVC 64 = IPsec (IKEv2) 8 a mutually exclusive. 0 - 11, 16 - 27, 32 - 43, 48 - legal values.
Use-Client-Address		17	Boolean	Single	0 = Disabled 1 = Enabled
VLAN	Y	140	Integer	Single	0-4094
WebVPN-Access-List	Y	73	String	Single	Access-List name
WebVPN ACL	Y	73	String	Single	Name of a WebVPN ACL on the device
WebVPN-ActiveX-Relay	Y	137	Integer	Single	0 = Disabled Otherwise = Enabled
WebVPN-Apply-ACL	Y	102	Integer	Single	0 = Disabled 1 = Enabled
WebVPN-Auto-HTTP-Signon	Y	124	String	Single	Reserved
WebVPN-Citrix-Metaframe-Enable	Y	101	Integer	Single	0 = Disabled 1 = Enabled
WebVPN-Content-Filter-Parameters	Y	69	Integer	Single	1 = Java ActiveX 2 = Java Script 4 = Image 8 = in images
WebVPN-Customization	Y	113	String	Single	Name of the customization
WebVPN-Default-Homepage	Y	76	String	Single	A URL such as http://example-example.com
WebVPN-Deny-Message	Y	116	String	Single	Valid string (up to 500 characters)
WebVPN-Download_Max-Size	Y	157	Integer	Single	0x7fffffff
WebVPN-File-Access-Enable	Y	94	Integer	Single	0 = Disabled 1 = Enabled
WebVPN-File-Server-Browsing-Enable	Y	96	Integer	Single	0 = Disabled 1 = Enabled
WebVPN-File-Server-Entry-Enable	Y	95	Integer	Single	0 = Disabled 1 = Enabled

Attribute Name	ASA	Attr. No.	Syntax/Type	Single or Multi- Valued	Description or Value
WebVPN-Group-based-HTTP/HTTPS-Proxy-Exception-List	Y	78	String	Single	Comma-separated DNS/IP with an optional v (for example *.cisco.com, 192.168.1.*, wwwin
WebVPN-Hidden-Shares	Y	126	Integer	Single	0 = None 1 = Visible
WebVPN-Home-Page-Use-Smart-Tunnel	Y	228	Boolean	Single	Enabled if clientless home page is to be rende Smart Tunnel.
WebVPN-HTML-Filter	Y	69	Bitmap	Single	1 = Java ActiveX 2 = Scripts 4 = Image 8 =
WebVPN-HTTP-Compression	Y	120	Integer	Single	0 = Off 1 = Deflate Compression
WebVPN-HTTP-Proxy-IP-Address	Y	74	String	Single	Comma-separated DNS/IP:port, with http= c prefix (for example http=10.10.10.10:80, https=11.11.11.11:443)
WebVPN-Idle-Timeout-Alert-Interval	Y	148	Integer	Single	0-30. 0 = Disabled.
WebVPN-Keepalive-Ignore	Y	121	Integer	Single	0-900
WebVPN-Macro-Substitution	Y	223	String	Single	Unbounded.
WebVPN-Macro-Substitution	Y	224	String	Single	Unbounded.
WebVPN-Port-Forwarding-Enable	Y	97	Integer	Single	0 = Disabled 1 = Enabled
WebVPN-Port-Forwarding-Exchange-Proxy-Enable	Y	98	Integer	Single	0 = Disabled 1 = Enabled
WebVPN-Port-Forwarding-HTTP-Proxy	Y	99	Integer	Single	0 = Disabled 1 = Enabled
WebVPN-Port-Forwarding-List	Y	72	String	Single	Port forwarding list name
WebVPN-Port-Forwarding-Name	Y	79	String	Single	String name (example, “Corporate-Apps”). This text replaces the default string, “Applicati on the clientless portal home page.
WebVPN-Post-Max-Size	Y	159	Integer	Single	0x7fffffff
WebVPN-Session-Timeout-Alert-Interval	Y	149	Integer	Single	0-30. 0 = Disabled.
WebVPN Smart-Card-Removal-Disconnect	Y	225	Boolean	Single	0 = Disabled 1 = Enabled
WebVPN-Smart-Tunnel	Y	136	String	Single	Name of a Smart Tunnel
WebVPN-Smart-Tunnel-Auto-Sign-On	Y	139	String	Single	Name of a Smart Tunnel auto sign-on list ap the domain name
WebVPN-Smart-Tunnel-Auto-Start	Y	138	Integer	Single	0 = Disabled 1 = Enabled 2 = Auto Start

Attribute Name	ASA	Attr. No.	Syntax/Type	Single or Multi- Valued	Description or Value
WebVPN-Smart-Tunnel-Tunnel-Policy	Y	227	String	Single	One of "e networkname," "i networkname," or "a networkname is the name of a Smart Tunnel netw e indicates the tunnel excluded, i indicates the tu specified, and a indicates all tunnels.
WebVPN-SSL-VPN-Client-Enable	Y	103	Integer	Single	0 = Disabled 1 = Enabled
WebVPN-SSL-VPN-Client-Keep- Installation	Y	105	Integer	Single	0 = Disabled 1 = Enabled
WebVPN-SSL-VPN-Client-Required	Y	104	Integer	Single	0 = Disabled 1 = Enabled
WebVPN-SSO-Server-Name	Y	114	String	Single	Valid string
WebVPN-Storage-Key	Y	162	String	Single	
WebVPN-Storage-Objects	Y	161	String	Single	
WebVPN-SVC-Keepalive-Frequency	Y	107	Integer	Single	15-600 seconds, 0=Off
WebVPN-SVC-Client-DPD-Frequency	Y	108	Integer	Single	5-3600 seconds, 0=Off
WebVPN-SVC-DTLS-Enable	Y	123	Integer	Single	0 = Disabled 1 = Enabled
WebVPN-SVC-DTLS-MTU	Y	125	Integer	Single	MTU value is from 256-1406 bytes.
WebVPN-SVC-Gateway-DPD-Frequency	Y	109	Integer	Single	5-3600 seconds, 0=Off
WebVPN-SVC-Rekey-Time	Y	110	Integer	Single	4-10080 minutes, 0=Off
WebVPN-SVC-Rekey-Method	Y	111	Integer	Single	0 (Off), 1 (SSL), 2 (New Tunnel)
WebVPN-SVC-Compression	Y	112	Integer	Single	0 (Off), 1 (Deflate Compression)
WebVPN-UNIX-Group-ID (GID)	Y	222	Integer	Single	Valid UNIX group IDs
WebVPN-UNIX-User-ID (UIDs)	Y	221	Integer	Single	Valid UNIX user IDs
WebVPN-Upload-Max-Size	Y	158	Integer	Single	0x7fffffff
WebVPN-URL-Entry-Enable	Y	93	Integer	Single	0 = Disabled 1 = Enabled
WebVPN-URL-List	Y	71	String	Single	URL list name
WebVPN-User-Storage	Y	160	String	Single	
WebVPN-VDI	Y	163	String	Single	List of settings

Supported IETF RADIUS Authorization Attributes

The following table lists the supported IETF RADIUS attributes.

Table 46: Supported IETF RADIUS Attributes

Attribute Name	ASA	Attr. No.	Syntax/Type	Single or Multi-Valued	Description or Value
IETF-Radius-Class	Y	25		Single	For Versions 8.2.x and later, we recommend that you use the Group-Policy attribute (VSA 3076, #25): • *group policy name* • OU=*group policy name* • OU=*group policy name*
IETF-Radius-Filter-Id	Y	11	String	Single	ACL name that is defined on the ASA, which applie only to full tunnel IPsec and SSL VPN clients.
IETF-Radius-Framed-IP-Address	Y	n/a	String	Single	An IP address
IETF-Radius-Framed-IP-Netmask	Y	n/a	String	Single	An IP address mask
IETF-Radius-Idle-Timeout	Y	28	Integer	Single	Seconds
IETF-Radius-Service-Type	Y	6	Integer	Single	Seconds. Possible Service Type values: • .Administrative—User is allowed access to the configure prompt. • .NAS-Prompt—User is allowed access to the ex prompt. • .remote-access—User is allowed network acces
IETF-Radius-Session-Timeout	Y	27	Integer	Single	Seconds

RADIUS Accounting Disconnect Reason Codes

These codes are returned if the ASA encounters a disconnect when sending packets:

Disconnect Reason Code
ACCT_DISC_USER_REQ = 1
ACCT_DISC_LOST_CARRIER = 2
ACCT_DISC_LOST_SERVICE = 3
ACCT_DISC_IDLE_TIMEOUT = 4
ACCT_DISC_SESS_TIMEOUT = 5
ACCT_DISC_ADMIN_RESET = 6
ACCT_DISC_ADMIN_REBOOT = 7

Disconnect Reason Code
ACCT_DISC_PORT_ERROR = 8
ACCT_DISC_NAS_ERROR = 9
ACCT_DISC_NAS_REQUEST = 10
ACCT_DISC_NAS_REBOOT = 11
ACCT_DISC_PORT_UNNEEDED = 12
ACCT_DISC_PORT_PREEMPTED = 13
ACCT_DISC_PORT_SUSPENDED = 14
ACCT_DISC_SERV_UNAVAIL = 15
ACCT_DISC_CALLBACK = 16
ACCT_DISC_USER_ERROR = 17
ACCT_DISC_HOST_REQUEST = 18
ACCT_DISC_ADMIN_SHUTDOWN = 19
ACCT_DISC_SA_EXPIRED = 21
ACCT_DISC_MAX_REASONS = 22

Guidelines for RADIUS Servers for AAA

This section describes the guidelines and limitations that you should check before configuring RADIUS servers for AAA.

- You can have up to 200 server groups in single mode or 4 server groups per context in multiple mode.
- Each group can have up to 16 servers in single mode or 8 servers in multiple mode.
- The maximum length of the RADIUS payload is 4096 bytes.

Configure RADIUS Servers for AAA

This section describes how to configure RADIUS servers for AAA.

Procedure

Step 1 Load the ASA attributes into the RADIUS server. The method that you use to load the attributes depends on which type of RADIUS server that you are using:

- If you are using Cisco ACS: the server already has these attributes integrated. You can skip this step.
- For RADIUS servers from other vendors (for example, Microsoft Internet Authentication Service): you must manually define each ASA attribute. To define an attribute, use the attribute name or number, type, value, and vendor code (3076).

Step 2 Configure RADIUS Server Groups, on page 1117.

Step 3 Add a RADIUS Server to a Group, on page 1120.

Configure RADIUS Server Groups

If you want to use an external RADIUS server for authentication, authorization, or accounting, you must first create at least one RADIUS server group per AAA protocol and add one or more servers to each group.

Procedure

Step 1 Create the RADIUS AAA server group.

aaa-server *group_name* **protocol radius**

Example:

```
ciscoasa(config)# aaa-server servergroup1 protocol radius
ciscoasa(config-aaa-server-group)#
```

When you enter the **aaa-server protocol** command, you enter aaa-server group configuration mode.

Step 2 (Optional.) Specify the maximum number of failed AAA transactions with a RADIUS server in the group before trying the next server.

max-failed-attempts *number*

The range is from 1 and 5. The default is 3.

If you configured a fallback method using the local database (for management access only), and all the servers in the group fail to respond, or their responses are invalid, then the group is considered to be unresponsive, and the fallback method is tried. The server group remains marked as unresponsive for a period of 10 minutes (by default), so that additional AAA requests within that period do not attempt to contact the server group, and the fallback method is used immediately. To change the unresponsive period from the default, see the **reactivation-mode** command in the next step.

If you do not have a fallback method, the ASA continues to retry the servers in the group.

Example:

```
ciscoasa(config-aaa-server-group)# max-failed-attempts 2
```

Step 3 (Optional.) Specify the method (reactivation policy) by which failed servers in a group are reactivated.

reactivation-mode {**depletion** [**deadtime** *minutes*] | **timed**}

Where:

- **depletion** [**deadtime** *minutes*] reactivates failed servers only after all of the servers in the group are inactive. This is the default reactivation mode. You can specify the amount of time, between 0 and 1440 minutes, that elapses between the disabling of the last server in the group and the subsequent reenabling of all servers. Deadtime applies only if you configure fallback to the local database; authentication is attempted locally until the deadtime elapses. The default is 10 minutes.
- **timed** reactivates failed servers after 30 seconds of down time.

Example:

```
ciscoasa(config-aaa-server-group)# reactivation-mode deadtime 20
```

Step 4 (Optional.) Send accounting messages to all servers in the group.

accounting-mode simultaneous

To restore the default of sending messages only to the active server, enter the **accounting-mode single** command.

Example:

```
ciscoasa(config-aaa-server-group)# accounting-mode simultaneous
```

Step 5 (Optional.) Enable the periodic generation of RADIUS interim-accounting-update messages.

interim-accounting-update [**periodic** [*hours*]]

ISE maintains a directory of active sessions based on the accounting records that it receives from NAS devices like the ASA. However, if ISE does not receive any indication that the session is still active (accounting message or posture transactions) for a period of 5 days, it will remove the session record from its database. To ensure that long-lived VPN connections are not removed, configure the group to send periodic interim-accounting-update messages to ISE for all active sessions.

- **periodic** [*hours*] enables the periodic generation and transmission of accounting records for every VPN session that is configured to send accounting records to the server group in question. You can optionally include the interval, in hours, for sending these updates. The default is 24 hours, the range is 1 to 120.
- (No parameters.) If you use this command without the **periodic** keyword, the ASA sends interim-accounting-update messages only when a VPN tunnel connection is added to a clientless VPN session. When this happens the accounting update is generated in order to inform the RADIUS server of the newly assigned IP address.

Example:

```
hostname(config-aaa-server-group)# interim-accounting-update periodic 12
```

Step 6 (Optional.) Enable the RADIUS Dynamic Authorization (ISE Change of Authorization, CoA) services for the AAA server group.

dynamic-authorization [**port** *number*]

Specifying a port is optional. The default is 1700, the range is 1024 to 65535.

When you use the server group in a VPN tunnel, the RADIUS server group will be registered for CoA notification and the ASA will listen to the port for the CoA policy updates from ISE. Enable dynamic authorization only if you are using this server group in a remote access VPN in conjunction with ISE.

Example:

```
ciscoasa(config-aaa-server-group)# dynamic-authorization
```

Step 7 (Optional.) If you do not want to use ISE for authentication, enable authorize-only mode for the RADIUS server group. (Enable authorize-only mode only if you are using this server group in a remote access VPN in conjunction with ISE.)

authorize-only

This indicates that when this server group is used for authorization, the RADIUS Access Request message will be built as an "Authorize Only" request as opposed to the configured password methods defined for the AAA server. If you do configure a common password using **radius-common-pw** command for the RADIUS server, it will be ignored.

For example, you would use authorize-only mode if you want to use certificates for authentication rather than this server group. You would still use this server group for authorization and accounting in the VPN tunnel.

Example:

```
ciscoasa(config-aaa-server-group)# authorize-only
```

Step 8 (Optional.) Merge a downloadable ACL with the ACL received in the Cisco AV pair from a RADIUS packet.

merge-dacl {**before-avpair** | **after-avpair**}

Example:

```
ciscoasa(config-aaa-server-group)# merge-dacl before-avpair
```

This option applies only to VPN connections. For VPN users, ACLs can be in the form of Cisco AV pair ACLs, downloadable ACLs, and an ACL that is configured on the ASA. This option determines whether or not the downloadable ACL and the AV pair ACL are merged, and does not apply to any ACLs configured on the ASA.

The default setting is **no merge dacl**, which specifies that downloadable ACLs will not be merged with Cisco AV pair ACLs. If both an AV pair and a downloadable ACL are received, the AV pair has priority and is used.

The **before-avpair** option specifies that the downloadable ACL entries should be placed before the Cisco AV pair entries.

The **after-avpair** option specifies that the downloadable ACL entries should be placed after the Cisco AV pair entries.

Examples

The following example shows how to add one RADIUS group with a single server:

```
ciscoasa(config)# aaa-server AuthOutbound protocol radius
ciscoasa(config-aaa-server-group)# exit
ciscoasa(config)# aaa-server AuthOutbound (inside) host 10.1.1.3
ciscoasa(config-aaa-server-host)# key RadUauthKey
ciscoasa(config-aaa-server-host)# exit
```

The following example shows how to configure an ISE server group for dynamic authorization (CoA) updates and hourly periodic accounting. Included is the tunnel group configuration that configures password authentication with ISE.

```
ciscoasa(config)# aaa-server ise protocol radius
ciscoasa(config-aaa-server-group)# interim-accounting-update periodic 1
ciscoasa(config-aaa-server-group)# dynamic-authorization
ciscoasa(config-aaa-server-group)# exit
ciscoasa(config)# aaa-server ise (inside) host 10.1.1.3
ciscoasa(config-aaa-server-host)# key sharedsecret
ciscoasa(config-aaa-server-host)# exit
ciscoasa(config)# tunnel-group aaa-coa general-attributes
ciscoasa(config-tunnel-general)# address-pool vpn
ciscoasa(config-tunnel-general)# authentication-server-group ise
ciscoasa(config-tunnel-general)# accounting-server-group ise
ciscoasa(config-tunnel-general)# exit
```

The following example shows how to configure a tunnel group for local certificate validation and authorization with ISE. Include the authorize-only command in the server group configuration, because the server group will not be used for authentication.

```
ciscoasa(config)# aaa-server ise protocol radius
ciscoasa(config-aaa-server-group)# authorize-only
ciscoasa(config-aaa-server-group)# interim-accounting-update periodic 1
ciscoasa(config-aaa-server-group)# dynamic-authorization
ciscoasa(config-aaa-server-group)# exit
ciscoasa(config)# aaa-server ise (inside) host 10.1.1.3
ciscoasa(config-aaa-server-host)# key sharedsecret
ciscoasa(config-aaa-server-host)# exit
ciscoasa(config)# tunnel-group aaa-coa general-attributes
ciscoasa(config-tunnel-general)# address-pool vpn
ciscoasa(config-tunnel-general)# authentication certificate
ciscoasa(config-tunnel-general)# authorization-server-group ise
ciscoasa(config-tunnel-general)# accounting-server-group ise
ciscoasa(config-tunnel-general)# exit
```

Add a RADIUS Server to a Group

To add a RADIUS server to a group, perform the following steps:

Procedure

Step 1 Identify the RADIUS server and the AAA server group to which it belongs.

aaa-server *server_group* [**(***interface_name***)**] **host** *server_ip*

Example:

```
ciscoasa(config-aaa-server-group)# aaa-server servergroup1 outside host 10.10.1.1
```

If you do not specify an **(***interface_name***)**, then the ASA uses the **inside** interface by default.

Step 2 Specify how the ASA treats netmasks received in a downloadable ACL from a RADIUS server.

acl-netmask-convert {**auto-detect** | **standard** | **wildcard**}

Example:

```
ciscoasa(config-aaa-server-host)# acl-netmask-convert standard
```

The**auto-detect** keyword specifies that the ASA should attempt to determine the type of netmask expression used. If the ASA detects a wildcard netmask expression, it converts it to a standard netmask expression.

The **standard** keyword specifies that the ASA assumes downloadable ACLs received from the RADIUS server contain only standard netmask expressions. No translation from wildcard netmask expressions is performed.

The **wildcard** keyword specifies that the ASA assumes downloadable ACLs received from the RADIUS server contain only wildcard netmask expressions and converts them all to standard netmask expressions when the ACLs are downloaded.

Step 3 Specify a common password to be used for all users who are accessing a RADIUS authorization server through the ASA.

radius-common-pw *string*

Example:

```
ciscoasa(config-aaa-server-host)# radius-common-pw examplepassword123abc
```

The *string* argument is a case-sensitive, alphanumeric keyword of up to 127 characters to be used as a common password for all authorization transactions with the RADIUS server.

Step 4 Enable MS-CHAPv2 authentication requests to the RADIUS server.

mschapv2-capable

Example:

```
ciscoasa(config-aaa-server-host)# mschapv2-capable
```

Step 5 Specify the timeout value for connection attempts to the server.

timeout *seconds*

Specify the timeout interval (1-300 seconds) for the server; the default is 10 seconds. For each AAA transaction the ASA retries connection attempts (based on the interval defined on the **retry-interval** command) until the timeout is reached. If the number of consecutive failed transactions reaches the limit specified on the max-failed-attempts command in the AAA server group, the AAA server is deactivated and the ASA starts sending requests to another AAA server if it is configured.

Example:

```
ciscoasa(config-aaa-server-host)# timeout 15
```

Step 6 Configure the amount of time between retry attempts for a particular AAA server designated in a previous command.

retry-interval *seconds*

Example:

```
ciscoasa(config-aaa-server-host)# retry-interval 8
```

The *seconds* argument specifies the retry interval (1-10 seconds) for the request. This is the time that the ASA waits before retrying a connection request.

Note
For the RADIUS protocol, if the server responds with an ICMP Port Unreachable message, the retry-interval setting is ignored and the AAA server is immediately moved to the failed state. If this is the only server in the AAA group, it is reactivated and another request is sent to it. This is the intended behavior.

Step 7 Send accounting messages to all servers in the group.

accounting-mode simultaneous

Example:

```
ciscoasa(config-aaa-server-group)# accounting-mode simultaneous
```

Enter the **accounting-mode single** command to restore the default of sending messages only to the active server.

Step 8 Specify the authentication port as port number1645, or the server port to be used for authentication of users.

authentication-port *port*

Example:

```
ciscoasa(config-aaa-server-host)# authentication-port 1646
```

Step 9 Specify the accounting port as port number 1646, or the server port to be used for accounting for this host.

accounting-port *port*

Example:

```
ciscoasa(config-aaa-server-host)# accounting-port 1646
```

Step 10 Specify the server secret value used to authenticate the RADIUS server to the ASA. The server secret that you configure should match the one configured on the RADIUS server. If you do not know the server secret value, ask the RADIUS server administrator. The maximum length is 64 characters.

key

Example:

```
ciscoasa(config-aaa-host)# key myexamplekey1
```

The server secret that you configure should match the one configured on the RADIUS server. If you do not know the server secret value, ask the RADIUS server administrator. The maximum length is 64 characters.

Example

The following example shows how to add a RADIUS server to an existing RADIUS server group:

```
ciscoasa(config)# aaa-server svrgrp1 protocol radius
ciscoasa(config-aaa-server-group)# aaa-server svrgrp1 host 192.168.3.4
ciscoasa(config-aaa-server-host)# acl-netmask-convert wildcard
ciscoasa(config-aaa-server-host)# radius-common-pw myexaplepasswordabc123
ciscoasa(config-aaa-server-host)# mschapv2-capable
ciscoasa(config-aaa-server-host)# timeout 9
ciscoasa(config-aaa-server-host)# retry-interval 7
ciscoasa(config-aaa-server-host)# accounting-mode simultaneous
ciscoasa(config-aaa-server-host)# authentication-port 1650
ciscoasa(config-aaa-server-host)# authorization-port 1645
ciscoasa(config-aaa-server-host)# key mysecretkeyexampleiceage2
ciscoasa(config-aaa-server-host)# exit
ciscoasa(config)#
```

Monitoring RADIUS Servers for AAA

See the following commands for monitoring the status of RADIUS servers for AAA:

- **show aaa-server**

  This command shows the configured RADIUS server statistics. You can use the **clear aaa-server statistics** command to reset the counters to zero.

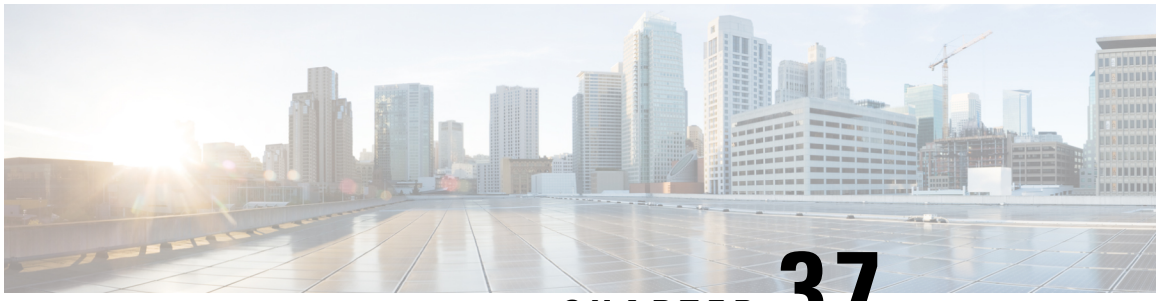
- **show running-config aaa-server**

  This command shows the RADIUS server running configuration.

History for RADIUS Servers for AAA

Table 47: History for RADIUS Servers for AAA

Feature Name	Platform Releases	Description
RADIUS Servers for AAA	7.0(1)	Describes how to configure RADIUS servers for AAA. We introduced the following commands: **aaa-server protocol**, **max-failed-attempts**, **reactivation-mode**, **accounting-mode simultaneous**, **aaa-server host**, **show aaa-server**, **show running-config aaa-server**, **clear aaa-server statistics**, **authentication-port**, **accounting-port**, **retry-interval**, **acl-netmask-convert**, **clear configure aaa-server**, **merge-dacl**, **radius-common-pw, key**.
Key vendor-specific attributes (VSAs) sent in RADIUS access request and accounting request packets from the ASA	8.4(3)	Four New VSAs—Tunnel Group Name (146) and Client Type (150) are sent in RADIUS access request packets from the ASA. Session Type (151) and Session Subtype (152) are sent in RADIUS accounting request packets from the ASA. All four attributes are sent for all accounting request packet types: Start, Interim-Update, and Stop. The RADIUS server (for example, ACS and ISE) can then enforce authorization and policy attributes or use them for accounting and billing purposes.
Increased limits for AAA server groups and servers per group.	9.13(1)	You can configure more AAA server groups. In single context mode, you can configure 200 AAA server groups (the former limit was 100). In multiple context mode, you can configure 8 (the former limit was 4). In addition, in multiple context mode, you can configure 8 servers per group (the former limit was 4 servers per group). The single context mode per-group limit of 16 remains unchanged. We modified the following commands to accept these new limits: **aaa-server**, **aaa-server host**.



CHAPTER 37

TACACS+ Servers for AAA

This chapter describes how to configure TACACS+ servers used in AAA.

- About TACACS+ Servers for AAA, on page 1125
- Guidelines for TACACS+ Servers for AAA, on page 1126
- Configure TACACS+ Servers, on page 1127
- Monitoring TACACS+ Servers for AAA, on page 1130
- History for TACACS+ Servers for AAA, on page 1130

About TACACS+ Servers for AAA

The ASA supports TACACS+ server authentication with the following protocols: ASCII, PAP, CHAP, and MS-CHAPv1.

TACACS+ Attributes

The ASA provides support for TACACS+ attributes. TACACS+ attributes separate the functions of authentication, authorization, and accounting. The protocol supports two types of attributes: mandatory and optional. Both the server and client must understand a mandatory attribute, and the mandatory attribute must be applied to the user. An optional attribute may or may not be understood or used.

Note To use TACACS+ attributes, make sure that you have enabled AAA services on the NAS.

The following table lists supported TACACS+ authorization response attributes for cut-through-proxy connections.

Table 48: Supported TACACS+ Authorization Response Attributes

Attribute	Description
acl	Identifies a locally configured ACL to be applied to the connection.
idletime	Indicates the amount of inactivity in minutes that is allowed before the authenticated user session is terminated.

Attribute	Description
timeout	Specifies the absolute amount of time in minutes that authentication credentials remain active before the authenticated user session is terminated.

The following table lists supported TACACS+ accounting attributes.

.

Table 49: Supported TACACS+ Accounting Attributes

Attribute	Description
bytes_in	Specifies the number of input bytes transferred during this connection (stop records only).
bytes_out	Specifies the number of output bytes transferred during this connection (stop records only).
cmd	Defines the command executed (command accounting only).
disc-cause	Indicates the numeric code that identifies the reason for disconnecting (stop records only).
elapsed_time	Defines the elapsed time in seconds for the connection (stop records only).
foreign_ip	Specifies the IP address of the client for tunnel connections. Defines the address on the lowest security interface for cut-through-proxy connections.
local_ip	Specifies the IP address that the client connected to for tunnel connections. Defines the address on the highest security interface for cut-through-proxy connections.
NAS port	Contains a session ID for the connection.
packs_in	Specifies the number of input packets transferred during this connection.
packs_out	Specifies the number of output packets transferred during this connection.
priv-level	Set to the user privilege level for command accounting requests or to 1 otherwise.
rem_iddr	Indicates the IP address of the client.
service	Specifies the service used. Always set to “shell” for command accounting only.
task_id	Specifies a unique task ID for the accounting transaction.
username	Indicates the name of the user.

Guidelines for TACACS+ Servers for AAA

This section describes the guidelines and limitation that you should check before configuring TACACS+ servers for AAA.

IPv6

The AAA server can use either an IPv4 or IPv6 address.

Additional Guidelines

- You can have up to 200 server groups in single mode or 4 server groups per context in multiple mode.
- Each group can have up to 16 servers in single mode or 8 servers in multiple mode.
- If you configure multi-factor authentication with ISE TACACS+ and Duo, you might need to re-enter your password when making SSH connections after acknowledging the login with Duo.
- For FPR1000, FPR2100, or FPR3100 Series that are running in ASA appliance mode, you must comply with these username conventions:
  - Must be Linux-valid usernames.
  - Must be lower-case only.
  - May include alphanumeric characters, period (.), or hyphen (-).
  - Must not include other special characters such as at sign (@) and slash (/).

Configure TACACS+ Servers

This section describes how to configure TACACS+ servers.

Procedure

Step 1 Configure TACACS+ Server Groups, on page 1127.

Step 2 Add a TACACS+ Server to a Group, on page 1129.

Configure TACACS+ Server Groups

If you want to use a TACACS+ server for authentication, authorization, or accounting, you must first create at least one TACACS+ server group and add one or more servers to each group. You identify TACACS+ server groups by name.

To add a TACACS+ server group, perform the following steps:

Procedure

Step 1 Identify the server group name and the protocol.

aaa-server *server_tag* **protocol tacacs+**

Example:

```
ciscoasa(config)# aaa-server servergroup1 protocol tacacs+
```

When you enter the **aaa-server protocol** command, you enter aaa-server group configuration mode.

Step 2 Specify the maximum number of failed AAA transactions with a AAA server in the group before trying the next server.

max-failed-attempts *number*

Example:

```
ciscoasa(config-aaa-server-group)# max-failed-attempts 2
```

The *number* argument can range from 1 and 5. The default is 3.

If you configured a fallback method using the local database (for management access only), and all the servers in the group fail to respond, or their responses are invalid, then the group is considered to be unresponsive, and the fallback method is tried. The server group remains marked as unresponsive for a period of 10 minutes (by default), so that additional AAA requests within that period do not attempt to contact the server group, and the fallback method is used immediately. To change the unresponsive period from the default, see the **reactivation-mode** command in the next step.

If you do not have a fallback method, the ASA continues to retry the servers in the group.

Step 3 Specify the method (reactivation policy) by which failed servers in a group are reactivated.

reactivation-mode {**depletion** [**deadtime** *minutes*] | **timed**}

Example:

```
ciscoasa(config-aaa-server-group)# reactivation-mode depletion deadtime 20
```

The **depletion** keyword reactivates failed servers only after all of the servers in the group are inactive.

The **deadtime** *minutes* keyword-argument pair specifies the amount of time in minutes, between 0 and 1440, that elapses between the disabling of the last server in the group and the subsequent reenabling of all servers. Deadtime applies only if you configure fallback to the local database; authentication is attempted locally until the deadtime elapses. The default is 10 minutes.

The **timed** keyword reactivates failed servers after 30 seconds of down time.

Step 4 Send accounting messages to all servers in the group.

accounting-mode simultaneous

Example:

```
ciscoasa(config-aaa-server-group)# accounting-mode simultaneous
```

To restore the default of sending messages only to the active server, enter the **accounting-mode single** command.

Example

The following example shows how to add one TACACS+ group with one primary and one backup server:

```
ciscoasa(config)# aaa-server AuthInbound protocol tacacs+
ciscoasa(config-aaa-server-group)# max-failed-attempts 2
ciscoasa(config-aaa-server-group)# reactivation-mode depletion deadtime 20
ciscoasa(config-aaa-server-group)# exit
ciscoasa(config)# aaa-server AuthInbound (inside) host 10.1.1.1
ciscoasa(config-aaa-server-host)# key TACPlusUauthKey
ciscoasa(config-aaa-server-host)# exit
ciscoasa(config)# aaa-server AuthInbound (inside) host 10.1.1.2
ciscoasa(config-aaa-server-host)# key TACPlusUauthKey2
ciscoasa(config-aaa-server-host)# exit
```

Add a TACACS+ Server to a Group

To add a TACACS+ server to a group, perform the following steps:

Procedure

Step 1 Identify the TACACS+ server and the server group to which it belongs.

aaa-server *server_group* [**(***interface_name***)**] **host** *server_ip*

Example:

```
ciscoasa(config-aaa-server-group)# aaa-server servergroup1 outside host 10.10.1.1
```

If you do not specify an **(***interface_name***)**, then the ASA uses the **inside** interface by default.

The server can use either an IPv4 or an IPv6 address.

Step 2 Specify the timeout value for connection attempts to the server.

timeout *seconds*

Specify the timeout interval (1-300 seconds) for the server; the default is 10 seconds. For each AAA transaction the ASA retries connection attempts (based on the interval defined on the **retry-interval** command) until the timeout is reached. If the number of consecutive failed transactions reaches the limit specified on the max-failed-attempts command in the AAA server group, the AAA server is deactivated and the ASA starts sending requests to another AAA server if it is configured.

Example:

```
ciscoasa(config-aaa-server-host)# timeout 15
```

Step 3 Specify the server port as port number 49, or the TCP port number used by the ASA to communicate with the TACACS+ server.

server-port *port_number*

Example:

```
ciscoasa(config-aaa-server-host)# server-port 49
```

Step 4 Specify the server secret value used to authenticate the NAS to the TACACS+ server.

key

Example:

```
ciscoasa(config-aaa-host)# key myexamplekey1
```

This value is a case-sensitive, alphanumeric keyword of up to 127 characters, which is the same value as the key on the TACACS+ server. Any characters over 127 are ignored. The key is used between the client and the server to encrypt data between them and must be the same on both the client and server systems. The key cannot contain spaces, but other special characters are allowed.

Monitoring TACACS+ Servers for AAA

See the following commands for monitoring TACACS+ servers for AAA:

- **show aaa-server**

  This command shows the configured TACACS+ server statistics. Enter the **clear aaa-server statistics** command to clear the TACACS+ server statistics.

- **show running-config aaa-server**

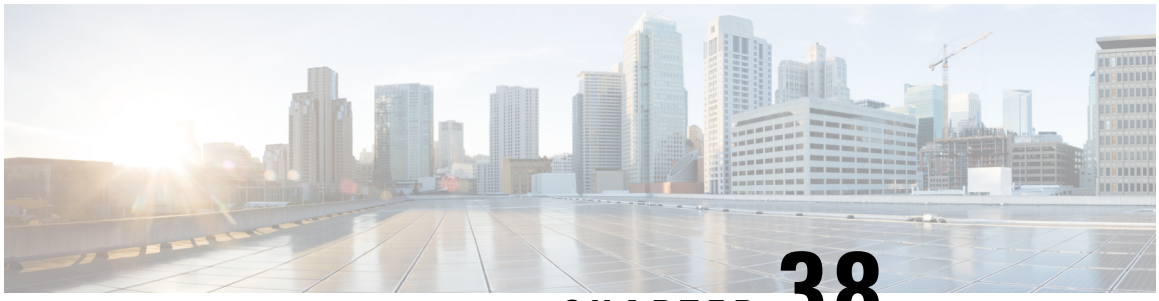
  This command shows the TACACS+ server running configuration. Enter the **clear configure aaa-server**command to clear the TACACS+ server configuration.

History for TACACS+ Servers for AAA

Table 50: History for TACACS+ Servers for AAA

Feature Name	Platform Releases	Description
TACACS+ Servers	7.0(1)	Describes how to configure TACACS+ servers for AAA. We introduced the following commands: **aaa-server protocol**, **max-failed-attempts, reactivation-mode, accounting-mode simultaneous, aaa-server host, aaa authorization exec authentication-server**, **server-port, key**, **clear aaa-server statistics**, **clear configure aaa-server**, **show aaa-server**, **show running-config aaa-server**, **username**, **service-type**, **timeout**.

Feature Name	Platform Releases	Description
TACACS+ servers with IPv6 addresses for AAA	9.7(1)	You can now use either an IPv4 or IPv6 address for the AAA server.
Increased limits for AAA server groups and servers per group.	9.13(1)	You can configure more AAA server groups. In single context mode, you can configure 200 AAA server groups (the former limit was 100). In multiple context mode, you can configure 8 (the former limit was 4). In addition, in multiple context mode, you can configure 8 servers per group (the former limit was 4 servers per group). The single context mode per-group limit of 16 remains unchanged. We modified the following commands to accept these new limits: **aaa-server**, **aaa-server host**.



CHAPTER 38

LDAP Servers for AAA

This chapter describes how to configure LDAP servers used in AAA.

- About LDAP and the ASA, on page 1133
- Guidelines for LDAP Servers for AAA, on page 1137
- Configure LDAP Servers for AAA, on page 1137
- Monitoring LDAP Servers for AAA, on page 1143
- History for LDAP Servers for AAA, on page 1144

About LDAP and the ASA

The ASA is compatible with the most LDAPv3 directory servers, including:

- Sun Microsystems JAVA System Directory Server, now part of Oracle Directory Server Enterprise Edition, and formerly named the Sun ONE Directory Server
- Microsoft Active Directory
- Novell
- OpenLDAP

By default, the ASA autodetects whether it is connected to Microsoft Active Directory, Sun LDAP, Novell, OpenLDAP, or a generic LDAPv3 directory server. However, if autodetection fails to determine the LDAP server type, you can manually configure it.

How Authentication Works with LDAP

During authentication, the ASA acts as a client proxy to the LDAP server for the user, and authenticates to the LDAP server in either plain text or by using the SASL protocol. By default, the ASA passes authentication parameters, usually a username and password, to the LDAP server in plain text.

The ASA supports the following SASL mechanisms, listed in order of increasing strength:

- Digest-MD5—The ASA responds to the LDAP server with an MD5 value computed from the username and password.
- Kerberos—The ASA responds to the LDAP server by sending the username and realm using the GSSAPI Kerberos mechanism.

The ASA and LDAP server supports any combination of these SASL mechanisms. If you configure multiple mechanisms, the ASA retrieves the list of SASL mechanisms that are configured on the server, and sets the authentication mechanism to the strongest one configured on both the ASA and the server. For example, if both the LDAP server and the ASA support both mechanisms, the ASA selects Kerberos, the stronger of the two.

When user LDAP authentication has succeeded, the LDAP server returns the attributes for the authenticated user. For VPN authentication, these attributes generally include authorization data that is applied to the VPN session. In this case, using LDAP accomplishes authentication and authorization in a single step.

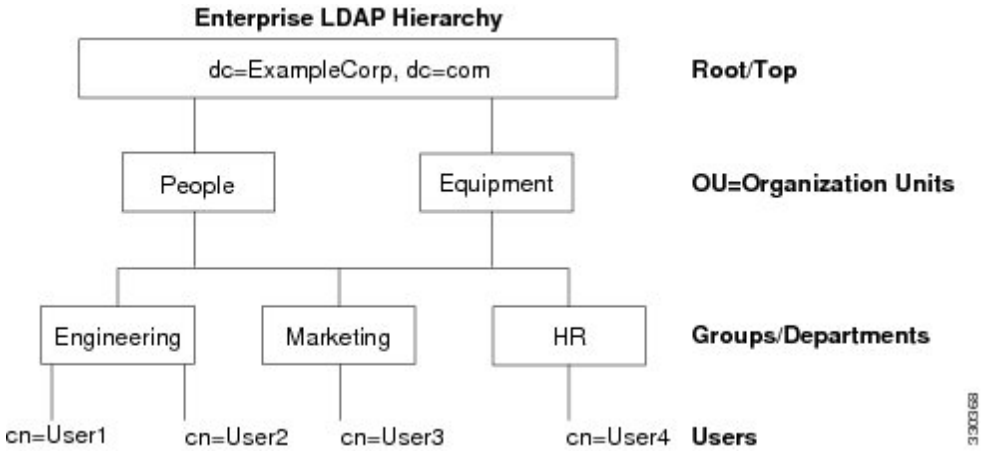
Note For more information about LDAP, see RFCs 1777, 2251, and 2849.

LDAP Hierarchy

Your LDAP configuration should reflect the logical hierarchy of your organization. For example, suppose an employee at your company, Example Corporation, is named Employee1. Employee1 works in the Engineering group. Your LDAP hierarchy could have one or many levels. You might decide to set up a single-level hierarchy in which Employee1 is considered a member of Example Corporation. Or you could set up a multi-level hierarchy in which Employee1 is considered to be a member of the department Engineering, which is a member of an organizational unit called People, which is itself a member of Example Corporation. See the following figure for an example of a multi-level hierarchy.

A multi-level hierarchy has more detail, but searches return results more quickly in a single-level hierarchy.

Figure 64: A Multi-Level LDAP Hierarchy



Search the LDAP Hierarchy

The ASA lets you tailor the search within the LDAP hierarchy. You configure the following three fields on the ASA to define where in the LDAP hierarchy that your search begins, the extent, and the type of information you are looking for. Together, these fields limit the search of the hierarchy to only the part that includes the user permissions.

- LDAP Base DN defines where in the LDAP hierarchy that the server should begin searching for user information when it receives an authorization request from the ASA.

- Search Scope defines the extent of the search in the LDAP hierarchy. The search proceeds this many levels in the hierarchy below the LDAP Base DN. You can choose to have the server search only the level immediately below it, or it can search the entire subtree. A single level search is quicker, but a subtree search is more extensive.
- Naming Attribute(s) defines the RDN that uniquely identifies an entry in the LDAP server. Common naming attributes can include cn (Common Name), sAMAccountName, and userPrincipalName.

The figure shows a sample LDAP hierarchy for Example Corporation. Given this hierarchy, you could define your search in different ways. The following table shows two sample search configurations.

In the first example configuration, when Employee1 establishes the IPsec tunnel with LDAP authorization required, the ASA sends a search request to the LDAP server, indicating it should search for Employee1 in the Engineering group. This search is quick.

In the second example configuration, the ASA sends a search request indicating that the server should search for Employee1 within Example Corporation. This search takes longer.

Table 51: Example Search Configurations

No.	LDAP Base DN	Search Scope	Naming Attribute	Result
1	group= Engineering,ou=People,dc=ExampleCorporation, dc=com	One Level	cn=Employee1	Quicker search
2	dc=ExampleCorporation,dc=com	Subtree	cn=Employee1	Longer search

Bind to an LDAP Server

The ASA uses the login DN and login password to establish trust (bind) with an LDAP server. When performing a Microsoft Active Directory read-only operation (such as authentication, authorization, or group search), the ASA can bind using a login DN with fewer privileges. For example, the login DN can be a user whose AD "Member Of" designation is part of Domain Users. For VPN password management operations, the login DN needs elevated privileges, and must be part of the Account Operators AD group.

The following is an example of a login DN:

```
cn=Binduser1,ou=Admins,ou=Users,dc=company_A,dc=com
```

The ASA supports the following authentication methods:

- Simple LDAP authentication with an unencrypted password on port 389
- Secure LDAP (LDAP-S) on port 636
- Simple Authentication and Security Layer (SASL) MD5
- SASL Kerberos

The ASA does not support anonymous authentication.

Note As an LDAP client, the ASA does not support the transmission of anonymous binds or requests.

LDAP Attribute Maps

The ASA can use an LDAP directory for authenticating users for:

- VPN remote access users
- Firewall network access/cut-through-proxy sessions
- Setting policy permissions (also called authorization attributes), such as ACLs, bookmark lists, DNS or WINS settings, and session timers.
- Setting the key attributes in a local group policy

The ASA uses LDAP attribute maps to translate native LDAP user attributes to ASA attributes. You can bind these attribute maps to LDAP servers or remove them. You can also show or clear attribute maps.

The LDAP attribute map does not support multi-valued attributes. For example, if a user is a member of several AD groups, and the LDAP attribute map matches more than one group, the value chosen is based on the alphabetization of the matched entries.

To use the attribute mapping features correctly, you need to understand LDAP attribute names and values, as well as the user-defined attribute names and values.

The names of frequently mapped LDAP attributes and the type of user-defined attributes that they would commonly be mapped to include the following:

- IETF-Radius-Class (Group_Policy in ASA version 8.2 and later)—Sets the group policy based on the directory department or user group (for example, Microsoft Active Directory memberOf) attribute value. The group policy attribute replaced the IETF-Radius-Class attribute with ASDM version 6.2/ASA version 8.2 or later.
- IETF-Radius-Filter-Id—Applies an access control list or ACL to VPN clients, IPsec, and SSL.
- IETF-Radius-Framed-IP-Address—Assigns a static IP address assigned to a VPN remote access client, IPsec, and SSL.
- Banner1—Displays a text banner when the VPN remote access user logs in.
- Tunneling-Protocols—Allows or denies the VPN remote access session based on the access type.

Note A single LDAP attribute map may contain one or many attributes. You can only map one LDAP attribute from a specific LDAP server.

Guidelines for LDAP Servers for AAA

This section includes the guidelines and limitations that you should check before configuring LDAP servers for AAA.

IPv6

The AAA server can use either an IPv4 or IPv6 address.

Additional Guidelines

- The DN configured on the ASA to access a Sun directory server must be able to access the default password policy on that server. We recommend using the directory administrator, or a user with directory administrator privileges, as the DN. Alternatively, you can place an ACL on the default password policy.
- You must configure LDAP over SSL to enable password management with Microsoft Active Directory and Sun servers.
- The ASA does not support password management with Novell, OpenLDAP, and other LDAPv3 directory servers.
- Beginning with Version 7.1(x), the ASA performs authentication and authorization using the native LDAP schema, and the Cisco schema is no longer needed.
- You can have up to 200 server groups in single mode or 4 server groups per context in multiple mode.
- Each group can have up to 16 servers in single mode or 8 servers in multiple mode.
- When a user logs in, the LDAP servers are accessed one at a time, starting with the first server that you specify in the configuration, until a server responds. If all servers in the group are unavailable, the ASA tries the local database if you configured it as a fallback method (management authentication and authorization only). If you do not have a fallback method, the ASA continues to try the LDAP servers.

Configure LDAP Servers for AAA

This section describes how to configure LDAP servers for AAA.

Procedure

Step 1 Configure LDAP attribute maps. See Configure LDAP Attribute Maps, on page 1138.

Step 2 Add an LDAP server group. See Configure LDAP Server Groups, on page 1139.

Step 3 (Optional) Configure authorization from an LDAP server that is separate and distinct from the authentication mechanism. See Configure Authorization with LDAP for VPN, on page 1142.

Configure LDAP Attribute Maps

To configure LDAP attribute maps, perform the following steps:

Procedure

Step 1 Create an unpopulated LDAP attribute map table.

ldap-attribute-map *map-name*

Example:

```
ciscoasa(config)# ldap-attribute-map att_map_1
```

Step 2 Map the user-defined attribute name department to the Cisco attribute.

map-name *user-attribute-name Cisco-attribute-name*

Example:

```
ciscoasa(config-ldap-attribute-map)# map-name department IETF-Radius-Class
```

Step 3 Map the user-defined map value department to the user-defined attribute value and the Cisco attribute value.

map-value *user-attribute-name Cisco-attribute-name*

Example:

```
ciscoasa(config-ldap-attribute-map)# map-value department Engineering group1
```

Step 4 Identify the server and the AAA server group to which it belongs.

aaa-server *server_group* [*interface_name*] **host** *server_ip*

Example:

```
ciscoasa(config)# aaa-server ldap_dir_1 host 10.1.1.4
```

Step 5 Bind the attribute map to the LDAP server.

ldap-attribute-map *map-name*

Example:

```
ciscoasa(config-aaa-server-host)# ldap-attribute-map att_map_1
```

Examples

The following example shows how to limit management sessions to the ASA based on an LDAP attribute called accessType. The accessType attribute may have one of these values:

- VPN
- admin
- helpdesk

The following example shows how each value is mapped to one of the valid IETF-Radius-Service-Type attributes that the ASA supports: remote-access (Service-Type 5) Outbound, admin (Service-Type 6) Administrative, and nas-prompt (Service-Type 7) NAS Prompt.

```
ciscoasa(config)# ldap attribute-map MGMT
ciscoasa(config-ldap-attribute-map)# map-name accessType IETF-Radius-Service-Type
ciscoasa(config-ldap-attribute-map)# map-value accessType VPN 5
ciscoasa(config-ldap-attribute-map)# map-value accessType admin 6
ciscoasa(config-ldap-attribute-map)# map-value accessType helpdesk 7

ciscoasa(config-ldap-attribute-map)# aaa-server LDAP protocol ldap
ciscoasa(config-aaa-server-group)# aaa-server LDAP (inside) host 10.1.254.91
ciscoasa(config-aaa-server-host)# ldap-base-dn CN=Users,DC=cisco,DC=local
ciscoasa(config-aaa-server-host)# ldap-scope subtree
ciscoasa(config-aaa-server-host)# ldap-login-password test
ciscoasa(config-aaa-server-host)# ldap-login-dn CN=Administrator,CN=Users,DC=cisco,DC=local
ciscoasa(config-aaa-server-host)# server-type auto-detect
ciscoasa(config-aaa-server-host)# ldap-attribute-map MGMT
```

The following example shows how to display the complete list of Cisco LDAP attribute names:

```
ciscoasa(config)# ldap attribute-map att_map_1
ciscoasa(config-ldap-attribute-map)# map-name att_map_1?

ldap mode commands/options:
cisco-attribute-names:
  Access-Hours
  Allow-Network-Extension-Mode
  Auth-Service-Type
  Authenticated-User-Idle-Timeout
  Authorization-Required
  Authorization-Type
 :
 :
  X509-Cert-Data
ciscoasa(config-ldap-attribute-map)#
```

Configure LDAP Server Groups

To create and configure an LDAP server group, then add an LDAP server to that group, perform the following steps:

Before you begin

You must add an attribute map before you may add an LDAP server to an LDAP server group.

Procedure

Step 1 Identify the server group name and the protocol.

aaa-server *server_tag* **protocol ldap**

Example:

```
ciscoasa(config)# aaa-server servergroup1 protocol ldap
ciscoasa(config-aaa-server-group)#
```

When you enter the **aaa-server protocol** command, you enter aaa-server group configuration mode.

Step 2 Specify the maximum number of failed AAA transactions with an LDAP server in the group before trying the next server.

max-failed-attempts *number*

Example:

```
ciscoasa(config-aaa-server-group)# max-failed-attempts 2
```

The *number* argument can range from 1 and 5. The default is 3.

If you configured a fallback method using the local database (for management access only) to configure the fallback mechanism, and all the servers in the group fail to respond, or their responses are invalid, then the group is considered to be unresponsive, and the fallback method is tried. The server group remains marked as unresponsive for a period of 10 minutes (by default), so that additional AAA requests within that period do not attempt to contact the server group, and the fallback method is used immediately. To change the unresponsive period from the default, see the **reactivation-mode** command in the next step.

If you do not have a fallback method, the ASA continues to retry the servers in the group.

Step 3 Specify the method (reactivation policy) by which failed servers in a group are reactivated.

reactivation-mode {**depletion** [**deadtime** *minutes*] | **timed**}

Example:

```
ciscoasa(config-aaa-server-group)# reactivation-mode deadtime 20
```

The **depletion** keyword reactivates failed servers only after all of the servers in the group are inactive.

The **deadtime** *minutes* keyword-argument pair specifies the amount of time in minutes, between 0 and 1440, that elapses between the disabling of the last server in the group and the subsequent reenabling of all servers. Deadtime applies only if you configure fallback to the local database; authentication is attempted locally until the deadtime elapses. The default is 10 minutes.

The **timed** keyword reactivates failed servers after 30 seconds of down time.

Step 4 Identify the LDAP server and AAA server group to which it belongs.

aaa-server *server_group* [**(***interface_name***)**] **host** *server_ip*

Example:

```
ciscoasa(config)# aaa-server servergroup1 outside host 10.10.1.1
```

If you do not specify an **(***interface_name***)**, then the ASA uses the **inside** interface by default.

When you enter the **aaa-server host** command, you enter aaa-server host configuration mode. As needed, use host configuration mode commands to further configure the AAA server.

The following table lists the available commands for LDAP servers, and whether or not a new LDAP server definition has a default value for that command. If no default value is provided (indicated by "—"), use the command to specify the value.

Table 52: Host Mode Commands and Defaults

Command	Default Value	Description
ldap-attribute-map	—	—
ldap-base-dn	—	—
ldap-login-dn	—	—
ldap-login-password	—	—
ldap-naming-attribute	—	—
ldap-over-ssl	636	If not set, the ASA uses sAMAccountName for LDAP requests. Whether using SASL or plain text, you can secure communications between the ASA and the LDAP server with SSL. If you do not configure SASL, we strongly recommend that you secure LDAP communications with SSL.
ldap-scope	—	—
sasl-mechanism	—	—
server-port	389	—
server-type	autodiscovery	If autodetection fails to determine the LDAP server type, and you know the server is either a Microsoft, Sun or generic LDAP server, you can manually configure the server type.
timeout	10 seconds	—

Example

The following example shows how to configure an LDAP server group named watchdogs and add an LDAP server to the group. Because the example does not define a retry interval or the port that the LDAP server listens to, the ASA uses the default values for these two server-specific parameters.

```
ciscoasa(config)# aaa-server watchdogs protocol ldap
ciscoasa(config-aaa-server-group)# aaa-server watchdogs host 192.168.3.4
ciscoasa(config-aaa-server-host)# exit
ciscoasa(config)#
```

Configure Authorization with LDAP for VPN

When LDAP user authentication for VPN access has succeeded, the ASA queries the LDAP server, which returns LDAP attributes. These attributes generally include authorization data that applies to the VPN session. Using LDAP in this way accomplishes authentication and authorization in a single step.

There may be cases, however, where you require authorization from an LDAP directory server that is separate and distinct from the authentication mechanism. For example, if you use an SDI or certificate server for authentication, no authorization information is returned. For user authorizations in this case, you can query an LDAP directory after successful authentication, accomplishing authentication and authorization in two steps.

To set up VPN user authorization using LDAP, perform the following steps.

Procedure

Step 1 Create an IPsec remote access tunnel group named remotegrp.

tunnel-group *groupname*

Example:

```
ciscoasa(config)# tunnel-group remotegrp
```

Step 2 Associate the server group and the tunnel group.

tunnel-group *groupname* **general-attributes**

Example:

```
ciscoasa(config)# tunnel-group remotegrp general-attributes
```

Step 3 Assign a new tunnel group to a previously created AAA server group for authorization.

authorization-server-group *group-tag*

Example:

```
ciscoasa(config-general)# authorization-server-group ldap_dir_1
```

Example

While there are other authorization-related commands and options available for specific requirements, the following example shows commands for enabling user authorization with LDAP. The example then creates an IPsec remote access tunnel group named remote-1, and assigns that new tunnel group to the previously created ldap_dir_1 AAA server group for authorization:

```
ciscoasa(config)# tunnel-group remote-1 type ipsec-ra
ciscoasa(config)# tunnel-group remote-1 general-attributes
ciscoasa(config-general)# authorization-server-group ldap_dir_1
ciscoasa(config-general)#
```

After you complete this configuration work, you can then configure additional LDAP authorization parameters such as a directory password, a starting point for searching a directory, and the scope of a directory search by entering the following commands:

```
ciscoasa(config)# aaa-server ldap_dir_1 protocol ldap
ciscoasa(config-aaa-server-group)# aaa-server ldap_dir_1 host 10.1.1.4
ciscoasa(config-aaa-server-host)# ldap-login-dn obscurepassword
ciscoasa(config-aaa-server-host)# ldap-base-dn starthere
ciscoasa(config-aaa-server-host)# ldap-scope subtree
ciscoasa(config-aaa-server-host)#
```

Monitoring LDAP Servers for AAA

See the following commands for monitoring LDAP servers for AAA:

- **show aaa-server**

  This command shows the configured AAA server statistics. Use the **clear aaa-server statistics** command to clear the AAA server statistics.

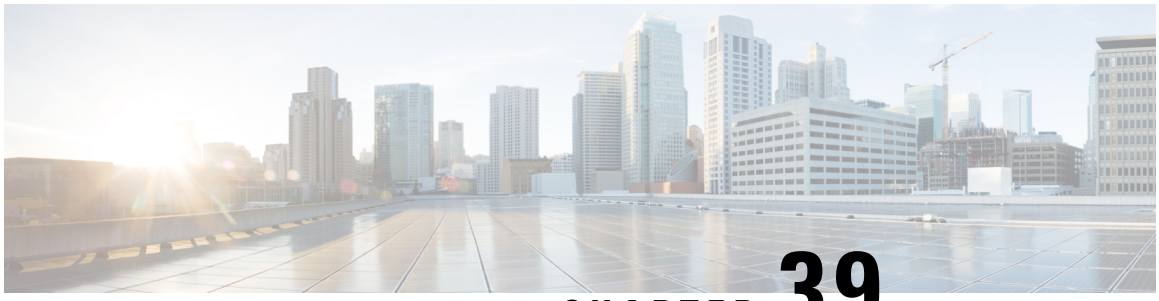
- **show running-config aaa-server**

  This command shows the AAA server running configuration. Use the **clear configure aaa-server** command to clear AAA server configuration.

History for LDAP Servers for AAA

Table 53: History for AAA Servers

Feature Name	Platform Releases	Description
LDAP Servers for AAA	7.0(1)	LDAP Servers describe support for AAA and how to configure LDAP servers. We introduced the following commands: **username**, **aaa authorization exec authentication-server**, **aaa authentication console LOCAL**, **aaa authorization exec LOCAL**, **service-type, ldap attribute-map**, **aaa-server protocol**, **aaa authentication telnet** | **ssh** | **serial**} **console LOCAL**, **aaa authentication http console LOCAL**, **aaa authentication enable console LOCAL**, **max-failed-attempts**, **reactivation-mode**, **accounting-mode simultaneous**, **aaa-server host**, **authorization-server-group**, **tunnel-group**, **tunnel-group general-attributes**, **map-name**, **map-value**, **ldap-attribute-map**.
LDAP servers with IPv6 addresses for AAA	9.7(1)	You can now use either an IPv4 or IPv6 address for the AAA server.
Increased limits for AAA server groups and servers per group.	9.13(1)	You can configure more AAA server groups. In single context mode, you can configure 200 AAA server groups (the former limit was 100). In multiple context mode, you can configure 8 (the former limit was 4). In addition, in multiple context mode, you can configure 8 servers per group (the former limit was 4 servers per group). The single context mode per-group limit of 16 remains unchanged. We modified the following commands to accept these new limits: **aaa-server**, **aaa-server host**.



CHAPTER 39

Kerberos Servers for AAA

The following topics explain how to configure Kerberos servers used in AAA. You can use Kerberos servers for the authentication of management connections, network access, and VPN user access.

- Guidelines for Kerberos Servers for AAA, on page 1145
- Configure Kerberos Servers for AAA, on page 1145
- Monitor Kerberos Servers for AAA, on page 1149
- History for Kerberos Servers for AAA, on page 1150

Guidelines for Kerberos Servers for AAA

- You can have up to 200 server groups in single mode or 8 server groups per context in multiple mode.
- Each group can have up to 16 servers in single mode or 8 servers in multiple mode. When a user logs in, the servers are accessed one at a time starting with the first server you specify in the configuration, until a server responds.

Configure Kerberos Servers for AAA

The following topics explain how to configure Kerberos server groups. You can then use these groups when configuring management access or VPNs.

Configure Kerberos AAA Server Groups

If you want to use a Kerberos server for authentication, you must first create at least one Kerberos server group and add one or more servers to each group.

Procedure

Step 1 Create the Kerberos AAA server group and enter aaa-server-group configuration mode.

aaa-server *server_group_name* **protocol kerberos**

Example:

```
ciscoasa(config)# aaa-server watchdog protocol kerberos
```

Step 2 (Optional.) Specify the maximum number of failed AAA transactions with a AAA server in the group before trying the next server.

max-failed-attempts *number*

Example:

```
ciscoasa(config-aaa-server-group)# max-failed-attempts 2
```

The *number* argument can range from 1 and 5. The default is 3.

If you configured a fallback method using the local database (for management access only), and all the servers in the group fail to respond, or their responses are invalid, then the group is considered to be unresponsive, and the fallback method is tried. The server group remains marked as unresponsive for a period of 10 minutes (by default), so that additional AAA requests within that period do not attempt to contact the server group, and the fallback method is used immediately. To change the unresponsive period from the default, see the **reactivation-mode** command in the next step.

If you do not have a fallback method, the ASA continues to retry the servers in the group.

Step 3 (Optional.) Specify the method (reactivation policy) by which failed servers in a group are reactivated.

reactivation-mode {**depletion** [**deadtime** *minutes*] | **timed**}

Example:

```
ciscoasa(config-aaa-server-group)# reactivation-mode depletion deadtime 20
```

The **depletion** keyword reactivates failed servers only after all of the servers in the group are inactive. This is the default mode.

The **deadtime** *minutes* keyword-argument pair specifies the amount of time in minutes, between 0 and 1440, that elapses between the disabling of the last server in the group and the subsequent reenabling of all servers. Deadtime applies only if you configure fallback to the local database; authentication is attempted locally until the deadtime elapses. The default is 10 minutes.

The **timed** keyword reactivates failed servers after 30 seconds of down time.

Step 4 (Optional.) Enable Kerberos Key Distribution Center (KDC) validation

validate-kdc

Example:

```
ciscoasa(config-aaa-server-group)# validate-kdc
```

To accomplish the authentication, you must also import a keytab file that you exported from the Kerberos Key Distribution Center (KDC). By validating the KDC, you can prevent an attack where the attacker spoofs the KDC so that user credentials are authenticated against the attacker's Kerberos server.

For information about how to upload the keytab file, see Configure Kerberos Key Distribution Center Validation, on page 1148.

Example

The following example creates a Kerberos server group named watchdogs, adds a server, and sets the realm to EXAMPLE.COM.

```
hostname(config)# aaa-server watchdogs protocol kerberos
hostname(config-aaa-server-group)# aaa-server watchdogs host 192.168.3.4
hostname(config-aaa-server-host)# kerberos-realm EXAMPLE.COM
hostname(config-aaa-server-host)# exit
hostname(config)#
```

Add Kerberos Servers to a Kerberos Server Group

Before you can use a Kerberos server group, you must add at least one Kerberos server to the group.

Procedure

Step 1 Add the Kerberos server to the Kerberos server group.

aaa-server *server_group* [**(***interface_name***)**] **host** *server_ip*

Example:

```
ciscoasa(config-aaa-server-group)# aaa-server servergroup1 outside host 10.10.1.1
```

If you do not specify an interface, then the ASA uses the **inside** interface by default.

You can use an IPv4 or IPv6 address.

Step 2 Specify the timeout value for connection attempts to the server.

timeout *seconds*

Specify the timeout interval (1-300 seconds) for the server; the default is 10 seconds. For each AAA transaction the ASA retries connection attempts (based on the interval defined on the **retry-interval** command) until the timeout is reached. If the number of consecutive failed transactions reaches the limit specified on the max-failed-attempts command in the AAA server group, the AAA server is deactivated and the ASA starts sending requests to another AAA server if it is configured.

Example:

```
ciscoasa(config-aaa-server-host)# timeout 15
```

Step 3 Specify the retry interval, which is the time the system waits before retrying a connection request.

retry-interval *seconds*

You can specify 1-10 seconds. The default is 10.

Example:

```
ciscoasa(config-aaa-server-host)# retry-interval 6
```

Step 4 Specify the server port if it is different from the default Kerberos port, which is TCP/88. The ASA contacts the Kerberos server on this port.

server-port *port_number*

Example:

```
ciscoasa(config-aaa-server-host)# server-port 8888
```

Step 5 Configure the Kerberos realm.

kerberos-realm *name*

Kerberos realm names use numbers and upper case letters only, and can be up to 64 characters. The name should match the output of the Microsoft Windows **set USERDNSDOMAIN** command when it is run on the Active Directory server for the Kerberos realm. In the following example, EXAMPLE.COM is the Kerberos realm name:

```
C:\>set USERDNSDOMAIN
USERDNSDOMAIN=EXAMPLE.COM
```

Although the ASA accepts lower case letters in the name, it does not translate lower case letters to upper case letters. Be sure to use upper case letters only.

Example:

```
ciscoasa(config-asa-server-group)# kerberos-realm EXAMPLE.COM
```

Example

```
hostname(config)# aaa-server watchdogs protocol kerberos
hostname(config-aaa-server-group)# aaa-server watchdogs host 192.168.3.4
ciscoasa(config-aaa-server-host)# timeout 9
ciscoasa(config-aaa-server-host)# retry 7
ciscoasa(config-aaa-server-host)# kerberos-realm EXAMPLE.COM
ciscoasa(config-aaa-server-host)# exit
ciscoasa(config)#
```

Configure Kerberos Key Distribution Center Validation

You can configure a Kerberos AAA server group to authenticate the servers in the group. To accomplish the authentication, you must import a keytab file that you exported from the Kerberos Key Distribution Center (KDC). By validating the KDC, you can prevent an attack where the attacker spoofs the KDC so that user credentials are authenticated against the attacker's Kerberos server.

When you enable KDC validation, after obtaining the ticket-granting ticket (TGT) and validating the user, the system also requests a service ticket on behalf of the user for host/ASA_hostname. The system then validates the returned service ticket against the secret key for the KDC, which is stored in a keytab file that you generated from the KDC and then uploaded to the ASA. If KDC authentication fails, the server is considered untrusted and the user is not authenticated.

The following procedure explains how to accomplish KDC authentication.

Before you begin

You cannot use KDC validation in conjunction with Kerberos Constrained Delegation (KCD). The **validate-kdc** command will be ignored if the server group is used for KCD.

Procedure

Step 1 (On the KDC.) Create a user account in the Microsoft Active Directory for the ASA (go to **Start** > **Programs** > **Administrative Tools** > **Active Directory Users and Computers**). For example, if the fully-qualified domain name (FQDN) of the ASA is asahost.example.com, create a user named asahost.

Step 2 (On the KDC.) Create a host service principal name (SPN) for the ASA using the FQDN and user account:

```
C:> setspn -A HOST/asahost.example.com asahost
```

Step 3 (On the KDC.) Create a keytab file for the ASA (line feeds added for clarity):

```
C:\Users\Administrator> ktpass /out new.keytab +rndPass
/princ host/asahost@EXAMPLE.COM
/mapuser asahost@example.com
/ptype KRB5_NT_SRV_HST
/mapop set
```

Step 4 (On the ASA.) Import the keytab (in this example, new.keytab) to the ASA using the **aaa kerberos import-keytab** command.

```
ciscoasa(config)# aaa kerberos import-keytab ftp://ftpserver.example.com/new.keytab
ftp://ftpserver.example.com/new.keytab imported successfully
```

Step 5 (On the ASA.) Add the **validate-kdc** command to the Kerberos AAA server group configuration. The keytab file is used only by server groups that contain this command.

```
ciscoasa(config)# aaa-server svrgrp1 protocol kerberos
ciscoasa(config-aaa-server-group)# validate-kdc
```

Monitor Kerberos Servers for AAA

You can use the following commands to monitor and clear Kerberos-related information.

- **show aaa-server**

  Shows the AAA server statistics. Use the **clear aaa-server statistics** command to clear the server statistics.

- **show running-config aaa-server**

  Shows the AAA servers that are configured for the system. Use the **clear configure aaa-server** command to remove the AAA server configuration.

- **show aaa kerberos** [**username** *user*]

  Shows all Kerberos tickets, or tickets for a given username.

- **clear aaa kerberos tickets** [**username** *user*]

  Clears all Kerberos tickets, or tickets for a given username.

- **show aaa kerberos keytab**

  Shows information about the Kerberos keytab file.

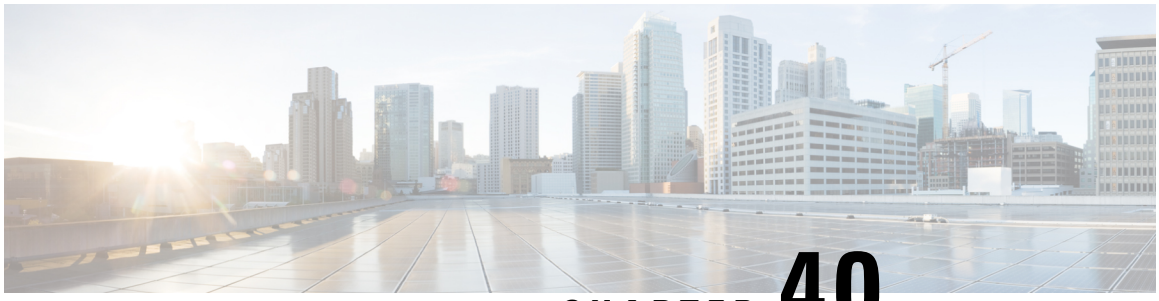
- **clear aaa kerberos keytab**

  Clears the Kerberos keytab file.

History for Kerberos Servers for AAA

Feature Name	Platform Releases	Description
Kerberos Servers	7.0(1)	Support for Kerberos servers for AAA. We introduced the following commands: **aaa-server protocol**, **max-failed-attempts**, **reactivation-mode**, **aaa-server host**, **kerberos-realm**, **server-port**, **clear aaa-server statistics**, **clear configure aaa-server**, **show aaa-server**, **show running-config aaa-server**, **timeout**.
IPv6 addresses for AAA	9.7(1)	You can now use either an IPv4 or IPv6 address for the AAA server.
Increased limits for AAA server groups and servers per group.	9.13(1)	You can configure more AAA server groups. In single context mode, you can configure 200 AAA server groups (the former limit was 100). In multiple context mode, you can configure 8 (the former limit was 4). In addition, in multiple context mode, you can configure 8 servers per group (the former limit was 4 servers per group). The single context mode per-group limit of 16 remains unchanged. We modified the following commands to accept these new limits: **aaa-server**, **aaa-server host**.

Feature Name	Platform Releases	Description
Kerberos Key Distribution Center (KDC) authentication.	9.8(4) and subsequent interim releases until 9.14(1)	You can import a keytab file from a Kerberos Key Distribution Center (KDC), and the system can authenticate that the Kerberos server is not being spoofed before using it to authenticate users. To accomplish KDC authentication, you must set up a **host**/*ASA_hostname* service principal name (SPN) on the Kerberos KDC, then export a keytab for that SPN. You then must upload the keytab to the ASA, and configure the Kerberos AAA server group to validate the KDC. We added the following commands: **aaa kerberos import-keytab**, **clear aaa kerberos keytab**, **show aaa kerberos keytab**, **validate-kdc**.



CHAPTER 40

RSA SecurID Servers for AAA

The following topics explain how to configure RSA SecurID servers used in AAA. The RSA SecureID servers are also known as SDI servers, because SDI is the protocol used to communicate with them. You can use RSA SecurID servers for the authentication of management connections, network access, and VPN user access.

- About RSA SecurID Servers, on page 1153
- Guidelines for RSA SecurID Servers for AAA, on page 1153
- Configure RSA SecurID Servers for AAA, on page 1154
- Monitor RSA SecurID Servers for AAA, on page 1157
- History for RSA SecurID Servers for AAA, on page 1157

About RSA SecurID Servers

You can use RSA SecurID servers either directly for authentication, or indirectly, as a second factor for authentication. In the latter case, you would configure the relationship to the SecurID server between the SecurID server and your RADIUS server, and configure the ASA to use the RADIUS server.

But, if you want to directly authenticate against the SecurID server, you would create a AAA server group for the SDI protocol, which is the protocol used to communicate with these servers.

When you use SDI, you need only specify the primary SecurID server when you create the AAA server group. The ASA will retrieve the sdiconf.rec file, which lists all of the SecurID server replicas, when it first connects to the server. The ASA can then use these replicas for authentication if the primary server does not respond.

In addition, you must register the ASA as an authentication agent in the RSA Authentication Manager. Authentication attempts will fail until you register the ASA.

Guidelines for RSA SecurID Servers for AAA

- You can have up to 200 server groups in single mode or 8 server groups per context in multiple mode.
- Each group can have up to 16 servers in single mode or 8 servers in multiple mode. When a user logs in, the servers are accessed one at a time starting with the first server you specify in the configuration, until a server responds.

Configure RSA SecurID Servers for AAA

The following topics explain how to configure RSA SecurID server groups. You can then use these groups when configuring management access or VPNs.

Configure RSA SecurID AAA Server Groups

If you want to use direct communication with an RSA SecurID server for authentication, you must first create at least one SDI server group and add one or more servers to each group. If you are using the SecurID server in a proxy relationship with a RADIUS server, you do not need to configure an SDI AAA server group on the ASA.

Procedure

Step 1 Create the SDI AAA server group and enter aaa-server-group configuration mode.

aaa-server *server_group_name* **protocol sdi**

Example:

```
ciscoasa(config)# aaa-server watchdog protocol sdi
```

Step 2 (Optional.) Specify the maximum number of failed AAA transactions with a AAA server in the group before trying the next server.

max-failed-attempts *number*

Example:

```
ciscoasa(config-aaa-server-group)# max-failed-attempts 2
```

The *number* argument can range from 1 and 5. The default is 3.

If you configured a fallback method using the local database (for management access only), and all the servers in the group fail to respond, or their responses are invalid, then the group is considered to be unresponsive, and the fallback method is tried. The server group remains marked as unresponsive for a period of 10 minutes (by default), so that additional AAA requests within that period do not attempt to contact the server group, and the fallback method is used immediately. To change the unresponsive period from the default, see the **reactivation-mode** command in the next step.

If you do not have a fallback method, the ASA continues to retry the servers in the group.

Step 3 (Optional.) Specify the method (reactivation policy) by which failed servers in a group are reactivated.

reactivation-mode {**depletion** [**deadtime** *minutes*] | **timed**}

Example:

```
ciscoasa(config-aaa-server-group)# reactivation-mode depletion deadtime 20
```

The **depletion** keyword reactivates failed servers only after all of the servers in the group are inactive. This is the default mode.

The **deadtime** *minutes* keyword-argument pair specifies the amount of time in minutes, between 0 and 1440, that elapses between the disabling of the last server in the group and the subsequent reenabling of all servers. Deadtime applies only if you configure fallback to the local database; authentication is attempted locally until the deadtime elapses. The default is 10 minutes.

The **timed** keyword reactivates failed servers after 30 seconds of down time.

Add RSA SecurID Servers to an SDI Server Group

Before you can use an SDI server group, you must add at least one RSA SecurID server to the group.

Servers in an SDI server group use the authentication and server management protocol (ACE) to communicate with the ASA.

Procedure

Step 1 Add the RSA SecurID server to the SDI server group.

aaa-server *server_group* [**(***interface_name***)**] **host** *server_ip*

Example:

```
ciscoasa(config-aaa-server-group)# aaa-server servergroup1 outside host 10.10.1.1
```

If you do not specify an interface, then the ASA uses the **inside** interface by default.

You can use an IPv4 or IPv6 address.

Step 2 Specify the timeout value for connection attempts to the server.

timeout *seconds*

Specify the timeout interval (1-300 seconds) for the server; the default is 10 seconds. For each AAA transaction the ASA retries connection attempts (based on the interval defined on the **retry-interval** command) until the timeout is reached. If the number of consecutive failed transactions reaches the limit specified on the **max-failed-attempts** command in the AAA server group, the AAA server is deactivated and the ASA starts sending requests to another AAA server if it is configured.

Example:

```
ciscoasa(config-aaa-server-host)# timeout 15
```

Step 3 Specify the retry interval, which is the time the system waits before retrying a connection request.

retry-interval *seconds*

You can specify 1-10 seconds. The default is 10.

Example:

```
ciscoasa(config-aaa-server-host)# retry-interval 6
```

Step 4 Specify the server port if it is different from the default RSA SecurID port, which is TCP/5500. The ASA contacts the RSA SecurID server on this port.

server-port *port_number*

Example:

```
ciscoasa(config-aaa-server-host)# server-port 5555
```

Import the SDI Node Secret File

You can manually import the node-secret file that is generated by the RSA Authentication Manager (SecurID) server.

Procedure

Step 1 Export the node secret file from the RSA Authentication Manager server. For details, see the RSA Authentication Manager documentation.

Step 2 Place an unzipped version of the node secret file on a server you can access from the ASA, or copy it to the ASA itself.

The server must support one of the following transfer protocols: FTP, HTTP, HTTPS, SCP, SMB, TFTP.

Step 3 Import the node secret file.

aaa sdi import-node-secret *filepath rsa_server_address password*

where:

- *filepath* is the complete path to the unzipped node secret file that was exported from the RSA Authentication Manager. Files on the local system can be addressed as disk0:, disk1:, or flash:. For files on a remote server, use standard URL notation, such as ftp://.
- *rsa_server_address* is the IP address or fully-qualified hostname of the RSA Authentication Manager server to which the node secret belongs.
- *password* is the password used to protect the file when you exported it.

Example:

```
ciscoasa# aaa sdi import-node-secret nodesecret.rec rsaam.example.com mysecret
nodesecret.rec imported successfully
ciscoasa#
```

Monitor RSA SecurID Servers for AAA

You can use the following commands to monitor and clear RSA SecurID-related information.

- **show aaa-server**

  Shows the AAA server statistics. Use the **clear aaa-server statistics** command to clear the server statistics.

- **show running-config aaa-server**

  Shows the AAA servers that are configured for the system. Use the **clear configure aaa-server** command to remove the AAA server configuration.

- **show aaa sdi node-secrets**

  Shows which RSA SecurID servers have an imported node secret file. Use the **clear aaa sdi node-secret** command to remove a node secret file.

History for RSA SecurID Servers for AAA

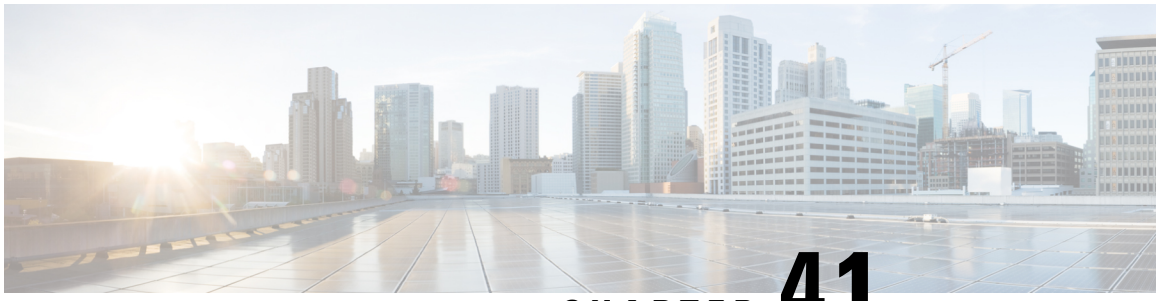
Feature Name	Platform Releases	Description
SecurID Servers	7.2(1)	Support for SecurID servers for AAA for management authentication. SecurID was supported in previous releases for VPN authentication.
IPv6 addresses for AAA	9.7(1)	You can now use either an IPv4 or IPv6 address for the AAA server.
Increased limits for AAA server groups and servers per group.	9.13(1)	You can configure more AAA server groups. In single context mode, you can configure 200 AAA server groups (the former limit was 100). In multiple context mode, you can configure 8 (the former limit was 4). In addition, in multiple context mode, you can configure 8 servers per group (the former limit was 4 servers per group). The single context mode per-group limit of 16 remains unchanged. We modified the following commands to accept these new limits: **aaa-server**, **aaa-server host**.
Manual import of node secret file from the RSA Authentication Manager for SDI AAA server groups.	9.15(1)	You can import the node secret file that you export from the RSA Authentication Manager for use with SDI AAA server groups. We added the following commands: **aaa sdi import-node-secret**, **clear aaa sdi node-secret**, **show aaa sdi node-secrets**.



PART VII

System Administration

- Management Access, on page 1161
- Software and Configurations, on page 1213
- Response Automation for System Events, on page 1255
- Testing and Troubleshooting, on page 1267



CHAPTER 41

Management Access

This chapter describes how to access the ASA for system management through Telnet, SSH, and HTTPS (using ASDM), how to authenticate and authorize users, and how to create login banners.

- Configure Management Remote Access, on page 1161
- Configure AAA for System Administrators, on page 1181
- Monitoring Device Access, on page 1202
- History for Management Access, on page 1205

Configure Management Remote Access

This section describes how to configure ASA access for ASDM, Telnet, or SSH, and other management parameters such as a login banner.

Configure SSH Access

SSH Guidelines

- To access the ASA interface for SSH access, you do not also need an access rule allowing the host IP address. You only need to configure SSH access according to this section.
- SSH access to an interface other than the one from which you entered the ASA is not supported. For example, if your SSH host is located on the outside interface, you can only initiate a management connection directly to the outside interface. The only exception to this rule is through a VPN connection (only supported for the ASA SSH stack). See Configure Management Access Over a VPN Tunnel, on page 1175.
- The ASA allows a maximum of 5 concurrent SSH connections per context/single mode, with a maximum of 100 connections divided among all contexts. However, because configuration commands might obtain locks on resources being changed, you should make changes in one SSH session at a time to ensure all changes are applied correctly.
- By default, the ASA uses a proprietary ASA SSH stack. You can choose to enable the CiscoSSH stack instead, which is based on OpenSSH. CiscoSSH supports:
  - FIPS compliance
  - Regular updates, including updates from Cisco and the open source community

Note that the Cisco SSH stack does not support:

- SSH to a different interface over VPN (management-access)
- EDDSA key pair
- RSA key pair in FIPS mode

If you need these features, you should use the ASA SSH stack.

- To use the ASA **copy** command to copy a file to or from an SCP server, you have to:
  - (CiscoSSH stack only) Enable SSH access on the ASA for the SCP server subnet/host using the **ssh** command.
  - Generate a key pair (for physical ASAs only) using the **crypto key generate** command.
- The SSH default username is no longer supported. You can no longer connect to the ASA using SSH with the **pix** or **asa** username and the login password. To use SSH, you must configure AAA authentication using the **aaa authentication ssh console LOCAL** command; then define a local user by entering the **username** command. If you want to use a AAA server for authentication instead of the local database, we recommend also configuring local authentication as a backup method.
- Only SSH Version 2 is supported.

To configure SSH access to the ASA, enable the SSH server and identify the allowed IP addresses. To authenticate users, you can use the following methods:

- Username and password using either the local database or a AAA server.
- Username and public key using the local database.

Enable the SSH Server

Enable the SSH server and specify which IP addresses are allowed to connect. You can also configure other SSH server settings.

Before you begin

- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter **changeto context** *name*.

Procedure

Step 1 (Optional) Use the CiscoSSH stack instead of the default ASA SSH stack.

ssh stack ciscossh

To return to the ASA SSH stack, use **no ssh stack ciscossh**.

In multiple context mode, use this command in the system execution space. To change from the context to the system configuration, enter **changeto system**.

Step 2 Identify the IP addresses from which the ASA accepts connections for each address or subnet, and the interface on which you can use SSH.

ssh *source_IP_address mask source_interface*

- *source_interface*—Specify any named interface. For bridge groups, specify the bridge group member interface. For VPN management access only (see Configure Management Access Over a VPN Tunnel, on page 1175), specify the named BVI interface.

Unlike Telnet, you can SSH on the lowest security level interface.

Example:

```
ciscoasa(config)# ssh 192.168.3.0 255.255.255.0 inside
```

Step 3 (Optional) Set the duration for how long an SSH session can be idle before the ASA disconnects the session.

ssh timeout *minutes*

Example:

```
ciscoasa(config)# ssh timeout 30
```

Set the timeout from 1 to 60 minutes. The default is 5 minutes. The default duration is too short in most cases, and should be increased until all pre-production testing and troubleshooting have been completed.

Step 4 (Optional) Enable the Secure Copy (SCP) server.

ssh scopy enable

The SCP server does not have directory support. The lack of directory support limits remote client access to the ASA internal files.

The SCP server does not support banners or wildcards.

Step 5 (Optional) Configure SSH cipher encryption algorithms:

ssh cipher encryption {**all** | **fips** | **high** | **low** | **medium** | **custom** *colon-delimited_list_of_encryption_ciphers*}

Example:

```
ciscoasa(config)# ssh cipher encryption custom 3des-cbc:aes128-cbc:aes192-cbc
```

The default is **medium**. Ciphers are used in the order they are listed. For pre-defined lists, they are listed from highest to lowest security.

- The **all** keyword specifies using all ciphers: 3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes128-gcm@openssh.com chacha20-poly1305@openssh.com aes192-ctr aes256-ctr
- The **custom** keyword specifies a custom cipher encryption configuration string, separated by colons.
- The **fips** keyword specifies only FIPS-compliant ciphers: aes128-cbc aes256-cbc
- The **high** keyword specifies only high-strength ciphers: aes256-cbc aes128-gcm@openssh.com chacha20-poly1305@openssh.com aes256-ctr
- The **low** keyword specifies low, medium, and high strength ciphers: 3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr

- The **medium** keyword specifies the medium and high strength ciphers (the default): 3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr

Step 6 (Optional) Configure SSH cipher integrity algorithms:

ssh cipher integrity {**all** | **fips** | **high** | **low** | **medium** | **custom** *colon-delimited_list_of_integrity_ciphers*}

Example:

```
ciscoasa(config)# ssh cipher integrity custom hmac-sha1-96:hmac-md5
```

The default is **high**.

- The **all** keyword specifies using all ciphers: hmac-sha1 hmac-sha1-96 (Deprecated) hmac-md5 (Deprecated) hmac-md5-96 (Deprecated) hmac-sha2-256
- The **custom** keyword specifies a custom cipher encryption configuration string, separated by colons.
- The **fips** keyword specifies only FIPS-compliant ciphers: hmac-sha1 hmac-sha2-256
- The **high** keyword specifies only high-strength ciphers (the default): hmac-sha2-256
- The **low** keyword specifies low, medium, and high strength ciphers: hmac-sha1 hmac-sha1-96 hmac-md5 hmac-md5-96 hmac-sha2-256
- The **medium** keyword specifies the medium and high strength ciphers: hmac-sha1 hmac-sha1-96 (Deprecated) hmac-sha2-256

Step 7 (Optional) (Admin context only) Set the Diffie-Hellman (DH) key exchange mode:

ssh key-exchange group {**curve25519-sha256** | **dh-group14-sha1** | **dh-group14-sha256** | **ecdh-sha2-nistp256**}

Example:

```
ciscoasa(config)# ssh key-exchange group dh-group14-sha1
```

The default is **dh-group14-sha256**

The DH key exchange provides a shared secret that cannot be determined by either party alone. The key exchange is combined with a signature and the host key to provide host authentication. This key-exchange method provides explicit server authentication. For more information about using DH key-exchange methods, see RFC 4253. You can only set the key exchange in the Admin context; this value is used by all contexts.

Examples

The following example shows an SCP session to the ASA. From a client on the external host, perform an SCP file transfer. For example, in Linux enter the following command:

scp -v -pw *password* [*path*/]*source_filename*
*username@asa_address***:**{**disk0**|**disk1**}**:/**[*path*/]*dest_filename*

The **-v** is for verbose, and if **-pw** is not specified, you will be prompted for a password.

Configure SSH for Password Access

Configure SSH authentication using a username and password.

Before you begin

- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter **changeto context** *name*.

Procedure

Step 1 Generate a key pair, which is required for SSH (for physical ASAs only).

For the ASAv, the key pairs are automatically created after deployment.

a) Generate the default key pair.

crypto key generate {**eddsa edwards-curve ed25519** | **ecdsa elliptic-curve** *size* |**rsa modulus** *size*}

Example:

```
ciscoasa(config)# crypto key generate ecdsa elliptic-curve 521
```

- **eddsa edwards-curve ed25519**—The key size is 256 bits. Not supported with the CiscoSSH stack.
- **ecdsa elliptic-curve** *size*—The size in bits is 256, 384, or 521.
- **rsa modulus** *size*—The size in bits is 2048, 3072, or 4096.

The keys used by SSH are called <Default-*type*-Key>. Do not specify the **label** keyword; non-default key pairs are not used. The larger the key size you specify, the longer it takes to generate a key pair. SSH tries keys in the following order: EdDSA, ECDSA, and then RSA. View the keys using the **show crypto key mypubkey** {**eddsa** | **ecdsa** | **rsa**} command.

b) (Optional) If you do not want to use the default key order (EdDSA, ECDSA, and then RSA), identify the key pair you want to use.

ssh key-exchange hostkey {**rsa** | **eddsa** | **ecdsa**}

If you choose RSA, you must use a key size 2048 or higher. For upgrade compatibility, smaller keys are only supported when you use the default key order.

Example:

```
ciscoasa(config)# ssh key-exchange hostkey ecdsa
```

Step 2 Save the keys to persistent flash memory.

write memory

Example:

```
ciscoasa(config)# write memory
```

Step 3 Create a user in the local database or on a AAA server that can be used for SSH access. See the following command to add a local username (recommended).

username *name* **password** *password* **privilege** *level*

You can use public key authentication or password authentication for the same local user. Public key authentication is not supported using a AAA server.

Example:

```
ciscoasa(config)# username admin password Far$cape1999 privilege 15
```

By default, the privilege level is 2; enter a level between 0 and 15, where 15 has all privileges.

Step 4 Enable local or AAA server authentication for SSH access.

aaa authentication ssh console {**LOCAL** | *server_group* [**LOCAL**]}

This command does not affect local public key authentication for usernames that have the **ssh authentication** command set. The ASA implicitly uses the local database for public key authentication. This command only affects local usernames with passwords. If you want to allow either public key authentication or password use by a local user, then you need to explicitly configure local authentication with this command to allow password access.

Example:

```
ciscoasa(config)# aaa authentication ssh console LOCAL
```

Configure SSH for Public Key Access

Configure SSH authentication using a public key.

Before you begin

- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter **changeto context** *name*.

Procedure

Step 1 Generate a key pair, which is required for SSH (for physical ASAs only).

For the ASAv, the key pairs are automatically created after deployment.

a) Generate the default key pair.

crypto key generate {**eddsa edwards-curve ed25519** | **ecdsa elliptic-curve** *size* |**rsa modulus** *size*}

Example:

```
ciscoasa(config)# crypto key generate ecdsa elliptic-curve 521
```

- **eddsa edwards-curve ed25519**—The key size is 256 bits. Not supported with the CiscoSSH stack.

- **ecdsa elliptic-curve** *size*—The size in bits is 256, 384, or 521.
- **rsa modulus** *size*—The size in bits is 2048, 3072, or 4096.

The keys used by SSH are called <Default-*type*-Key>. Do not specify the **label** keyword; non-default key pairs are not used. The larger the key size you specify, the longer it takes to generate a key pair. SSH tries keys in the following order: EdDSA, ECDSA, and then RSA. View the keys using the **show crypto key mypubkey** {**eddsa** | **ecdsa** | **rsa**} command.

b) (Optional) If you do not want to use the default key order (EdDSA, ECDSA, and then RSA), identify the key pair you want to use.

ssh key-exchange hostkey {**rsa** | **eddsa** | **ecdsa**}

If you choose RSA, you must use a key size 2048 or higher. For upgrade compatibility, smaller keys are only supported when you use the default key order.

Example:

```
ciscoasa(config)# ssh key-exchange hostkey ecdsa
```

Step 2 Save the keys to persistent flash memory.

write memory

Example:

```
ciscoasa(config)# write memory
```

Step 3 Create a user in the local database that can be used for SSH access.

username *name* [**password** *password*] **privilege** *level*

Example:

```
ciscoasa(config)# username admin password Far$cape1999 privilege 15
```

By default, the privilege level is 2; enter a level between 0 and 15, where 15 has all privileges. You might want to create a user without a password if you want to force the user to use public key authentication (**ssh authentication**) instead of password authentication. If you configure public key authentication as well as a password in the **username** command, then the user can log in with either method if you explicitly configure AAA authentication in Configure SSH for Password Access, on page 1165. **Note:** Do not use the **username** command **nopassword** option; the **nopassword** option allows *any* password to be entered, not no password.

Step 4 Allow public key authentication for a user instead of/as well as password authentication, and enter the public key on the ASA:

username *name* **attributes**

ssh authentication {**pkf** | **publickey** *key*}

Example:

```
ciscoasa(config)# username admin attributes
ciscoasa(config-username)# ssh authentication pkf
```

```
Enter an SSH public key formatted file.
End with the word "quit" on a line by itself:
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "256-bit ED25519, converted by dean@dwinchester-mac from "
AAAAC3NzaC1lZDI1NTE5AAAAIDmIeTNfEOnuH0094p1MKX80fW2O216g4trnf7gwWe5Q
---- END SSH2 PUBLIC KEY ----
quit
INFO: Import of an SSH public key formatted file SUCCEEDED.
```

For a local **username**, you can enable public key authentication instead of/as well as password authentication. You can generate a public key/private key pair using any SSH key generation software (such as ssh keygen) that can generate ssh-rsa, ssh-ed25519 or ecdsa-sha2-nistp raw keys (with no certificates). Enter the public key on the ASA. The SSH client then uses the private key (and the passphrase you used to create the key pair) to connect to the ASA.

For a **pkf** key, you are prompted to paste in a PKF formatted key, up to 4096 bits. Use this format for keys that are too large to paste inline in Base64 format. For example, you can generate a 4096-bit key using ssh keygen, then convert it to PKF, and use the **pkf** keyword to be prompted for the key. **Note:** You can use the **pkf** option with failover, but the PKF key is not automatically replicated to the standby system. You must enter the **write standby** command to synchronize the PKF key.

For a **publickey** *key*, the key is a Base64-encoded public key. You can generate the key using any SSH key generation software (such as ssh keygen) that can generate ssh-rsa, ssh-ed25519 or ecdsa-sha2-nistp raw keys (with no certificates).

Examples

The following example shows how to authenticate using a PKF formatted key:

```
ciscoasa(config)# crypto key generate rsa modulus 4096
ciscoasa(config)# write memory
ciscoasa(config)# username exampleuser1 password examplepassword1 privilege 15
ciscoasa(config)# username exampleuser1 attributes
ciscoasa(config-username)# ssh authentication pkf
Enter an SSH public key formatted file.
End with the word "quit" on a line by itself:
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "4096-bit RSA, converted by xxx@xxx from OpenSSH"
AAAAB3NzaC1yc2EAAAADAQABAAACAQDNUvkgza37lB/Q/fljpLAv1BbyAd5PJCJXh/U4LO
hleR/qgIROjpnFaS7Az8/+sjHmq0qXC5TXkzWihvRZbhefyPhPHCi0hIt4oUF2ZbXESA/8
jUT4ehXIUE7FrChffBBtbD4d9FkV8A2gwZCDJBxEM26ocbZCSTx9QC//wt6E/zRcdoqiJG
p4ECEdDaM+56l+yf73NUigO7wYkqcrzjmI1rZRDLVcqtj8Q9qD3MqsV+PkJGSGiqZwnyIl
QbfYxXHU9wLdWxhUbA/xOjJuZ15TQMa7KLs2u+RtrpQgeTGTffIh6O+xKh93gwTgzaZTK4
CQ1kuMrRdNRzza0byLeYPtSlv6Lv6F6dGtwlqrX5a+w/tV/aw9WUg/rapekKloz3tsPTDe
p866AFzU+Z7pVR1389iNuNJHQS7IUA2m0cciIuCM2we/tVqMPYJl+xgKAkuHDkBlMS4i8b
Wzyd+4EUMDGGZVeO+corKTLWFO1wIUieRkrUaCzjComGYZdzrQT2mXBcSKQNWlSCBpCHsk
/r5uTGnKpCNWfL7vd/sRCHyHKsxjsXR15C/5zgHmCTAaGOuIq0Rjo34+61+70PCtYXebxM
Wwm19e3eH2PudZd+rj1dedfr2/IrislEBRJWGLoR/N+xsvwVVM1Qqw1uL4r99CbZF9NghY
NRxCQOY/7K77II==
---- END SSH2 PUBLIC KEY ----
quit
INFO: Import of an SSH public key formatted file SUCCEEDED.
ciscoasa(config)#
```

The following example generates a shared key for SSH on a Linux or Macintosh system, and imports it to the ASA:

1. Generate the RSA public and private keys for 4096 bits on your computer:

```
jcrichton-mac:~ john$ ssh-keygen -b 4096
Generating public/private rsa key pair.
Enter file in which to save the key (/Users/john/.ssh/id_rsa):
/Users/john/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase): pa$$phrase
Enter same passphrase again: pa$$phrase
Your identification has been saved in /Users/john/.ssh/id_rsa.
Your public key has been saved in /Users/john/.ssh/id_rsa.pub.
The key fingerprint is:
c0:0a:a2:3c:99:fc:00:62:f1:ee:fa:f8:ef:70:c1:f9 john@jcrichton-mac
The key's randomart image is:
+--[ RSA 4096]----+
| .               |
|  o  .           |
|+...  o          |
|B.+.....         |
|.B ..+  S        |
|  =   o          |
|   + . E         |
|  o o            |
| ooooo           |
+-----------------+
```

2. Convert the key to PKF format:

```
jcrichton-mac:~ john$ cd .ssh
jcrichton-mac:.ssh john$ ssh-keygen -e -f id_rsa.pub
---- BEGIN SSH2 PUBLIC KEY ----
Comment: "4096-bit RSA, converted by john@jcrichton-mac from OpenSSH"
AAAAB3NzaC1yc2EAAAADAQABAAACAQDNUvkgza37lB/Q/fljpLAv1BbyAd5PJCJXh/U4LO
hleR/qgIROjpnDaS7Az8/+sjHmq0qXC5TXkzWihvRZbhefyPhPHCi0hIt4oUF2ZbXESA/8
jUT4ehXIUE7FrChffBBtbD4d9FkV8A2gwZCDJBxEM26ocbZCSTx9QC//wt6E/zRcdoqiJG
p4ECEdDaM+56l+yf73NUigO7wYkqcrzjmI1rZRDLVcqtj8Q9qD3MqsV+PkJGSGiqZwnyIl
QbfYxXHU9wLdWxhUbA/xOjJuZ15TQMa7KLs2u+RtrpQgeTGTffIh6O+xKh93gwTgzaZTK4
CQ1kuMrRdNRzza0byLeYPtSlv6Lv6F6dGtwlqrX5a+w/tV/aw9WUg/rapekKloz3tsPTDe
p866AFzU+Z7pVR1389iNuNJHQS7IUA2m0cciIuCM2we/tVqMPYJl+xgKAkuHDkBlMS4i8b
Wzyd+4EUMDGGZVeO+corKTLWFO1wIUieRkrUaCzjComGYZdzrQT2mXBcSKQNWlSCBpCHsk
/r5uTGnKpCNWfL7vd/sRCHyHKsxjsXR15C/5zgHmCTAaGOuIq0Rjo34+61+70PCtYXebxM
Wwm19e3eH2PudZd+rj1dedfr2/IrislEBRJWGLoR/N+xsvwVVM1Qqw1uL4r99CbZF9NghY
NRxCQOY/7K77IQ==
---- END SSH2 PUBLIC KEY ----
jcrichton-mac:.ssh john$
```

3. Copy the key to your clipboard.
4. In ASDM, choose **Configuration** > **Device Management** > **Users/AAA** > **User Accounts**, select the username and then click **Edit**. Click **Public Key Using PKF** and paste the key into the window:
5. Verify the user can SSH to the ASA. For the password, enter the SSH key password you specified when you created the key pair.

```
jcrichton-mac:.ssh john$ ssh test@10.86.118.5
```

```
The authenticity of host '10.86.118.5 (10.86.118.5)' can't be established.
RSA key fingerprint is 39:ca:ed:a8:75:5b:cc:8e:e2:1d:96:2b:93:b5:69:94.
Are you sure you want to continue connecting (yes/no)? yes
```

The following dialog box appears for you to enter your passphrase:



Meanwhile, in the terminal session:

```
Warning: Permanently added '10.86.118.5' (RSA) to the list of known hosts.
Identity added: /Users/john/.ssh/id_rsa (/Users/john/.ssh/id_rsa)
Type help or '?' for a list of available commands.
asa>
```

Configure Telnet Access

To identify the client IP addresses allowed to connect to the ASA using Telnet, perform the following steps. See the following guidelines:

- To access the ASA interface for Telnet access, you do not also need an access rule allowing the host IP address. You only need to configure Telnet access according to this section.
- Telnet access to an interface other than the one from which you entered the ASA is not supported. For example, if your Telnet host is located on the outside interface, you can only initiate a Telnet connection directly to the outside interface. The only exception to this rule is through a VPN connection. See Configure Management Access Over a VPN Tunnel, on page 1175.
- You cannot use Telnet to the lowest security interface unless you use Telnet inside a VPN tunnel.
- The ASA allows a maximum of 5 concurrent Telnet connections per context/single mode, with a maximum of 100 connections divided among all contexts.

Before you begin

- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter **changeto context** *name*.
- To gain access to the ASA CLI using Telnet, enter the login password set by the **password** command. You must manually set the password before using Telnet.

Procedure

Step 1 Identify the IP addresses from which the ASA accepts connections for each address or subnet on the specified interface.

telnet *source_IP_address mask source_interface*

- *source_interface*—Specify any named interface. For bridge groups, specify the bridge group member interface. For VPN management access only (see Configure Management Access Over a VPN Tunnel, on page 1175), specify the named BVI interface.

If there is only one interface, you can configure Telnet to access that interface as long as the interface has a security level of 100.

Example:

```
ciscoasa(config)# telnet 192.168.1.2 255.255.255.255 inside
```

Step 2 Set the duration for how long a Telnet session can be idle before the ASA disconnects the session.

telnet timeout *minutes*

Example:

```
ciscoasa(config)#  telnet timeout 30
```

Set the timeout from 1 to 1440 minutes. The default is 5 minutes. The default duration is too short in most cases and should be increased until all pre-production testing and troubleshooting have been completed.

Examples

The following example shows how to let a host on the inside interface with an address of 192.168.1.2 access the ASA:

```
ciscoasa(config)# telnet 192.168.1.2 255.255.255.255 inside
```

The following example shows how to allow all users on the 192.168.3.0 network to access the ASA on the inside interface:

```
ciscoasa(config)# telnet 192.168.3.0. 255.255.255.255 inside
```

Configure HTTPS Access for ASDM, Other Clients

To use ASDM or other HTTPS clients such as CSM, you need to enable the HTTPS server, and allow HTTPS connections to the ASA. HTTPS access is enabled as part of the factory default configuration. To configure HTTPS access, perform the following steps. See the following guidelines:

- To access the ASA interface for HTTPS access, you do not also need an access rule allowing the host IP address. You only need to configure HTTPS access according to this section. If, however, you configure HTTP redirect to redirect HTTP connections to HTTPS automatically, you must enable an access rule to allow HTTP; otherwise, the interface cannot listen to the HTTP port.
- Management access to an interface other than the one from which you entered the ASA is not supported. For example, if your management host is located on the outside interface, you can only initiate a management connection directly to the outside interface. The only exception to this rule is through a VPN connection. See Configure Management Access Over a VPN Tunnel, on page 1175.
- In single context mode, you can have a maximum 5 ASDM concurrent sessions. In multiple context mode, you can have a maximum of 5 concurrent ASDM sessions per context, with a maximum of 200 ASDM instances among all contexts.

  ASDM sessions use two HTTPS connections: one for monitoring that is always present, and one for making configuration changes that is present only when you make changes. For example, the multiple-context mode system limit of 200 ASDM sessions represents a limit of 400 HTTPS sessions.
- The ASA allows a maximum of 6 concurrent non-ASDM HTTPS sessions in single context mode or per context, if available, with a maximum or 100 HTTPS sessions among all contexts.
- If you enable both SSL (**webvpn** > **enable** *interface*) and HTTPS access on the same interface, you can access AnyConnect Client from **https://***ip_address* and ASDM from **https://***ip_address*/**admin**, both on port 443. If you also enable **aaa authentication http console**, then you must specify a different port for ASDM access.

Before you begin

- In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter **changeto context** *name*.

Procedure

Step 1 Identify the IP addresses from which the ASA accepts HTTPS connections for each address or subnet on the specified interface.

http *source_IP_address mask source_interface*

- *source_interface*—Specify any named interface. For bridge groups, specify the bridge group member interface. For VPN management access only (see Configure Management Access Over a VPN Tunnel, on page 1175), specify the named BVI interface.

Example:

```
ciscoasa(config)# http 192.168.1.2 255.255.255.255 inside
```

Step 2 Enable the HTTPS server.

http server enable [*port*]

Example:

```
ciscoasa(config)# http server enable 444
```

By default, the port is 443. If you change the port number, be sure to include it in the ASDM access URL. For example, if you change the port number to 444, enter the following URL:

https://10.1.1.1:444

Step 3 Allow non-browser-based HTTPS clients to access HTTPS services on the ASA. By default, ASDM, CSM, and REST API are allowed.

http server basic-auth-client *user_agent*

- *user_agent*—Specifies the client's User-Agent string in the HTTP header of the HTTP request. You can specify the complete string or a partial string; partial strings must match the start of the User-Agent string. We recommend complete strings for better security. Note that the string is case-sensitive.

  For example, **curl** will match the following User-Agent string:

  curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.19.1 Basic ECC zlib/1.2.3 libidn/1.18 libssh2/1.4.2

  curl will *not* match the following User-Agent string:

  abcd curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.19.1 Basic ECC zlib/1.2.3 libidn/1.18 libssh2/1.4.2

  CURL will *not* match the following User-Agent string:

  curl/7.19.7 (x86_64-redhat-linux-gnu) libcurl/7.19.7 NSS/3.19.1 Basic ECC zlib/1.2.3 libidn/1.18 libssh2/1.4.2

Enter each client string using a separate command. Many specialty clients (for example, python libraries, curl, and wget) do not support Cross-site request forgery (CSRF) token-based authentication, so you need to specifically allow these clients to use the ASA basic authentication method. For security purposes, you should only allow required clients.

Example:

```
ciscoasa(config)# http server basic-auth-client curl
```

Step 4 (Optional) Set connection and session timeouts.

http server idle-timeout*minutes*

http server session-timeout*minutes*

http connection idle-timeout*seconds*

- **http server idle-timeout** *minutes*—Set the idle timeout for ASDM connections, from 1-1440 minutes. The default is 20 minutes. The ASA disconnects an ASDM connection that is idle for the set period of time.
- **http server session-timeout** *minutes*—Set the session timeout for ASDM sessions, from 1-1440 minutes. This timeout is disabled by default. The ASA disconnects an ASDM session that exceeds the set period of time.
- **http connection idle-timeout** *seconds*—Set the idle timeout for all HTTPS connections, including ASDM, WebVPN, and other clients, from 10-86400 seconds. This timeout is disabled by default. The

ASA disconnects a connection that is idle for the set period of time. If you set both the **http server idle-timeout** and the **http connection idle-timeout** commands, the **http connection idle-timeout** command takes precendence.

Example:

```
ciscoasa(config)# http server idle-timeout 30
ciscoasa(config)# http server session-timeout 120
```

Examples

The following example shows how to enable the HTTPS server and let a host on the inside interface with an address of 192.168.1.2 access ASDM:

```
ciscoasa(config)# http server enable
ciscoasa(config)# http 192.168.1.2 255.255.255.255 inside
```

The following example shows how to allow all users on the 192.168.3.0/24 network to access ASDM on the inside interface:

```
ciscoasa(config)# http 192.168.3.0 255.255.255.0 inside
```

Configure HTTP Redirect for ASDM Access or Clientless SSL VPN

You must use HTTPS to connect to the ASA using ASDM or clientless SSL VPN. For your convenience, you can redirect HTTP management connections to HTTPS. For example, by redirecting HTTP, you can enter either **http://10.1.8.4/admin/** or **https://10.1.8.4/admin/** and still arrive at the ASDM launch page at the HTTPS address.

You can redirect both IPv4 and IPv6 traffic.

Before you begin

Normally, you do not need an access rule allowing the host IP address. However, for HTTP redirect, you must enable an access rule to allow HTTP; otherwise, the interface cannot listen to the HTTP port.

Procedure

Enable HTTP redirect:

http redirect *interface_name* [*port*]

Example:

```
ciscoasa(config)# http redirect outside 88
```

The *port* identifies the port from which the interface redirects HTTP connections. The default is 80.

Configure Management Access Over a VPN Tunnel

If your VPN tunnel terminates on one interface, but you want to manage the ASA by accessing a different interface, you must identify that interface as a management-access interface. For example, if you enter the ASA from the outside interface, this feature lets you connect to the inside interface using ASDM, SSH, or Telnet; or you can ping the inside interface when entering from the outside interface.

Note This feature is not supported for SSH if you use the CiscoSSH stack.

Note This feature is not supported for SNMP.

VPN access to an interface other than the one from which you entered the ASA is not supported. For example, if your VPN access is located on the outside interface, you can only initiate a connection directly to the outside interface. You should enable VPN on the directly-accessible interface of the ASA and use name resolution so that you don't have to remember multiple addresses.

Management access is available via the following VPN tunnel types: IPsec clients, IPsec Site-to-Site, Easy VPN, and the AnyConnect Client SSL VPN.

Before you begin

- This feature is not supported on management-only interfaces.
- When you use a management-access interface and you configure identity NAT, you must configure NAT with the route lookup option. For more information see the "NAT and VPN Management Access" section in the *NAT Examples and Reference* chapter in the appropriate release of the ASA Firewall CLI Configuration Guide.

Procedure

Specify the name of the management interface that you want to access when entering the ASA from another interface.

management-access *management_interface*

For site-to-site tunnels, you can specify a named BVI (in routed mode).

Example:

```
ciscoasa(config)# management-access inside
```

Configure Management Access for FXOS on Firepower 2100 Platform Mode Data Interfaces

If you want to manage FXOS on the Firepower 2100 in Platform Mode from a data interface, you can configure SSH, HTTPS, and SNMP access. This feature is useful if you want to manage the device remotely, but you want to keep Management 1/1, which is the native way to access FXOS, on an isolated network. If you enable this feature, you can continue to use Management 1/1 for local access only. However, you cannot allow *remote* access to or from Management 1/1 for FXOS at the same time as using this feature. This feature requires forwarding traffic to the ASA data interfaces using an internal path (the default), and you can only specify one FXOS management gateway.

The ASA uses non-standard ports for FXOS access; the standard port is reserved for use by the ASA on the same interface. When the ASA forwards traffic to FXOS, it translates the non-standard destination port to the FXOS port for each protocol (do not change the HTTPS port in FXOS). The packet destination IP address (which is the ASA interface IP address) is also translated to an internal address for use by FXOS. The source address remains unchanged. For returning traffic, the ASA uses its data routing table to determine the correct egress interface. When you access the ASA data IP address for the management application, you must log in using an FXOS username; ASA usernames only apply for ASA management access.

You can also enable FXOS management traffic *initiation* on the ASA data interfaces, which is required for SNMP traps, or NTP and DNS server access, for example. By default, FXOS management traffic initiation is enabled for the ASA outside interface for DNS and NTP server communication (required for Smart Software Licensing communication).

Before you begin

- Single context mode only.
- Excludes ASA management-only interfaces.
- You cannot use a VPN tunnel to the ASA data interface and access FXOS directly. As a workaround for SSH, you can VPN to the ASA, access the *ASA* CLI, and then use the **connect fxos** command to access the FXOS CLI. Note that SSH, HTTPS, and SNMPv3 are/can be encrypted, so direct connection to the data interface is safe.
- Ensure that the FXOS gateway is set to forward traffic to the ASA data interfaces (the default). See the getting started guide for more information about setting the gateway.

Procedure

Step 1 Enable FXOS remote management.

fxos {**https** | **ssh** | **snmp**} **permit** {*ipv4_address netmask* | *ipv6_address/prefix_length*} *interface_name*

Example:

```
ciscoasa(config)# fxos https permit 192.168.1.0 255.255.155.0 inside
ciscoasa(config)# fxos https permit 2001:DB8::34/64 inside
ciscoasa(config)# fxos ssh permit 192.168.1.0 255.255.155.0 inside
ciscoasa(config)# fxos ssh permit 2001:DB8::34/64 inside
```

Step 2 (Optional) Change the default port for the service.

fxos {**https** | **ssh** | **snmp**} **port** *port*

See the following defaults:

- HTTPS default port—3443
- SNMP default port—3061
- SSH default port—3022

Example:

```
ciscoasa(config)# fxos https port 6666
ciscoasa(config)# fxos ssh port 7777
```

Step 3 Allow FXOS to initiate management connections from the ASA interface.

ip-client *interface_name*

By default, the outside interface is enabled.

Example:

```
ciscoasa(config)# ip-client outside
ciscoasa(config)# ip-client services
```

Step 4 Connect to the Firewall Chassis Manager on Management 1/1 (by default https://192.168.45.45, with the username: **admin** and password: **Admin123**).

Step 5 Click the **Platform Settings** tab, and enable **SSH**, **HTTPS**, or **SNMP**.

SSH and HTTPS are enabled by default.

Step 6 Configure an **Access List** on the **Platform Settings** tab to allow your management addresses. SSH and HTTPS only allow the Management 1/1 192.168.45.0 network by default. You need to allow any addresses that you specified in the **FXOS Remote Management** configuration on the ASA.

Change the Console Timeout

The console timeout sets how long a connection can remain in privileged EXEC mode or configuration mode; when the timeout is reached, the session drops into user EXEC mode. By default, the session does not time out. This setting does not affect how long you can remain connected to the console port, which never times out.

Procedure

Specify the idle time in minutes (0 through 60) after which the privileged session ends.

console timeout *number*

Example:

```
ciscoasa(config)# console timeout 0
```

The default timeout is 0, which means the session does not time out.

Customize a CLI Prompt

The ability to add information to a prompt allows you to see at-a-glance which ASA you are logged into when you have multiple modules. During a failover, this feature is useful when both ASAs have the same hostname.

In multiple context mode, you can view the extended prompt when you log in to the system execution space or the admin context. Within a non-admin context, you only see the default prompt, which is the hostname and the context name.

By default, the prompt shows the hostname of the ASA. In multiple context mode, the prompt also displays the context name. You can display the following items in the CLI prompt:

cluster-unit	Displays the cluster unit name. Each unit in a cluster can have a unique name.
context	(Multiple mode only) Displays the name of the current context.
domain	Displays the domain name.
hostname	Displays the hostname.
priority	Displays the failover priority as pri (primary) or sec (secondary).
state	Displays the traffic-passing state or role of the unit. For failover, the following values are displayed for the state keyword: • **act**—Failover is enabled, and the unit is actively passing traffic. • **stby**— Failover is enabled, and the unit is not passing traffic and is in a standby, failed, or other non-active state. • **actNoFailover**—Failover is not enabled, and the unit is actively passing traffic. • **stbyNoFailover**—Failover is not enabled, and the unit is not passing traffic. This might happen when there is an interface failure above the threshold on the standby unit. For clustering, the values for control and data are shown.

Procedure

Customize the CLI prompt by entering the following command:

prompt {[**hostname**] [**context**] [**domain**] [**slot**] [**state**] [**priority**] [**cluster-unit**]}

Example:

```
ciscoasa(config)# prompt hostname context slot state priority
ciscoasa/admin/pri/act(config)#
```

The order in which you enter the keywords determines the order of the elements in the prompt, which are separated by a slash (/).

Configure a Login Banner

You can configure a message to display when a user connects to the ASA, before a user logs in, or before a user enters privileged EXEC mode.

- From a security perspective, it is important that your banner discourage unauthorized access. Do not use the words “welcome” or “please,” as they appear to invite intruders in. The following banner sets the correct tone for unauthorized access:

```
You have logged in to a secure device.
If you are not authorized to access this device,
log out immediately or risk possible criminal consequences.
```

- After a banner has been added, Telnet or SSH sessions to the ASA may close if:
  - There is not enough system memory available to process the banner message(s).
  - A TCP write error occurs when trying to display banner message(s).
- See RFC 2196 for guidelines about banner messages.

Procedure

Add a banner to display at one of three times: when a user first connects (message-of-the-day (motd)), when a user logs in (login), and when a user accesses privileged EXEC mode (exec).

banner {**exec** | **login** | **motd**} *text*

Example:

```
ciscoasa(config)# banner motd Only authorized access is allowed to $(hostname).
```

When a user connects to the ASA, the message-of-the-day banner appears first, followed by the login banner and prompts. After the user successfully logs in to the ASA, the exec banner appears.

To add more than one line, precede each line by the **banner** command.

For the banner text:

- Spaces are allowed, but tabs cannot be entered using the CLI.
- There are no limits for banner length other than those for RAM and flash memory.
- You can dynamically add the hostname or domain name of the ASA by including the strings **$(hostname)** and **$(domain)**.

- If you configure a banner in the system configuration, you can use that banner text within a context by using the **$(system)** string in the context configuration.

Examples

The following examples show how to add a message-of-the-day banner:

```
ciscoasa(config)# banner motd Only authorized access is allowed to $(hostname).

ciscoasa(config)# banner motd Contact me at admin@example.com for any issues.
```

Set a Management Session Quota

You can establish a maximum number of simultaneous ASDM, SSH, and Telnet sessions that are allowed on the ASA. If the maximum is reached, no additional sessions are allowed and a syslog message is generated. To prevent a system lockout, the management session quota mechanism cannot block a console session.

Note In multiple context mode, you cannot configure the number of ASDM sessions, where the maximum is fixed at 5 sessions.

Note If you also set a resource limit per context for the maximum administrative sessions (SSH, etc.), then the lower value will be used.

Before you begin

In multiple context mode, complete this procedure in the context execution space. To change from the system to a context configuration, enter the **changeto context** *name*command.

Procedure

Step 1 Enter the following command:

quota management-session [**ssh** | **telnet** | **http** | **user**] *number*

- **ssh**—Sets the maximum SSH sessions, between 1 and 5. The default is 5.
- **telnet**—Sets the maximum Telnet sessions, between 1 and 5. The default is 5.
- **http**—Sets the maximum HTTPS (ASDM) sessions, between 1 and 5. The default is 5.
- **user**—Sets the maximum sessions per user, between 1 and 5. The default is 5.

- *number*—Sets the number of sessions. When entered without any other keywords, this argument sets the aggregate number of sessions between 1 and 15. The default is 15.

Example:

```
ciscoasa(config)# quota management-session ssh 3
ciscoasa(config)# quota management-session telnet 1
ciscoasa(config)# quota management-session http 4
ciscoasa(config)# quota management-session user 2
```

Step 2 View the current sessions in use.

show quota management-session [**ssh** | **telnet** | **http** | **user**]

Example:

```
ciscoasa(config)#show quota management-session

#Sessions               ConnectionType                  Username
1                              SSH                       cisco
2                              TELNET                    cisco
1                              SSH                       cisco1
```

Configure AAA for System Administrators

This section describes how to configure authentication, management authorization, and command authorization for system administrators.

Configure Management Authentication

Configure authentication for CLI and ASDM access.

About Management Authentication

How you log into the ASA depends on whether or not you enable authentication.

About SSH Authentication

See the following behavior for SSH access with and without authentication:

- No Authentication—SSH is not available without authentication.
- Authentication—When you enable SSH authentication, you enter the username and password as defined on the AAA server or local user database. For public key authentication, the ASA only supports the local database. If you configure SSH public key authentication, then the ASA uses the local database implicitly. You only need to explicitly configure SSH authentication when you use a username and password to log in. You access user EXEC mode.

About Telnet Authentication

See the following behavior for Telnet access with and without authentication:

- No Authentication—If you do not enable any authentication for Telnet, you do not enter a username; you enter the login password (set with the **password** command). There is no default password, so you must set one before you can Telnet to the ASA. You access user EXEC mode.
- Authentication—If you enable Telnet authentication, you enter the username and password as defined on the AAA server or local user database. You access user EXEC mode.

About ASDM Authentication

See the following behavior for ASDM access with and without authentication. You can also configure certificate authentication, with or without AAA authentication.

- No Authentication—By default, you can log into ASDM with a blank username and the enable password set by the **enable password** command, which is blank by default. We suggest that you change the enable password as soon as possible so that it does not remain blank; see Set the Hostname, Domain Name, and the Enable and Telnet Passwords, on page 735. When you enter the **enable** command at the CLI for the first time, you are prompted to change the password; this behavior is not enforced when you log into ASDM. Note that if you enter a username and password at the login screen (instead of leaving the username blank), ASDM checks the local database for a match.
- Certificate Authentication—(Single, routed mode only) You can require that the user have a valid certificate. Enter the certificate username and password, and the ASA validates the certificate against the PKI trustpoint.
- AAA Authentication—When you enable ASDM (HTTPS) authentication, you enter the username and password as defined on the AAA server or local user database. You can no longer use ASDM with a blank username and the enable password.
- AAA Authentication plus Certificate Authentication—(Single, routed mode only) When you enable ASDM (HTTPS) authentication, you enter the username and password as defined on the AAA server or local user database. If the username and password are different for the certificate authentication, you are prompted to enter them as well. You can opt to pre-fill the username derived from your certificate.

About Serial Authentication

See the following behavior for access to the serial console port with and without authentication:

- No Authentication—If you do not enable any authentication for serial access, you do not enter a username or password. You access user EXEC mode.
- Authentication—If you enable authentication for serial access, you enter the username and password as defined on the AAA server or local user database. You access user EXEC mode.

About Enable Authentication

To enter privileged EXEC mode after logging in, enter the **enable** command. How this command works depends on whether or not you enable authentication:

- No Authentication—If you do not configure enable authentication, enter the system enable password when you enter the **enable** command (set by the **enable password** command), which is blank by default. The first time you enter the **enable** command, you are prompted to change it. However, if you do not use enable authentication, after you enter the **enable** command, you are no longer logged in as a particular

user, which can affect user-based features such as command authorization. To maintain your username, use enable authentication.

- Authentication—If you configure enable authentication, the ASA prompts you for your username and password as defined on the AAA server or local user database. This feature is particularly useful when you perform command authorization, in which usernames are important in determining the commands that a user can enter.

For enable authentication using the local database, you can use the **login** command instead of the **enable** command. The **login** command maintains the username, but requires no configuration to turn on authentication.



Caution If you add users to the local database who can gain access to the CLI and whom you do not want to enter privileged EXEC mode, you should configure command authorization. Without command authorization, users can access privileged EXEC mode (and all commands) at the CLI using their own password if their privilege level is 2 or greater (2 is the default). Alternatively, you can discourage the login command by using a AAA server for authentication instead of the local database, or you can set all local users to level 1 so you can control who can use the system enable password to access privileged EXEC mode.

Sessions from the Host Operating System to the ASA

Some platforms support running the ASA as a separate application: for example the ASA on the Firepower 4100/9300. For sessions from the host operating system to the ASA, you can configure serial and Telnet authentication, depending on the type of connection. For example, the **connect asa** command in FXOS on the Firepower 2100 in Platform mode uses a serial connection.

In multiple context mode, you cannot configure any AAA commands in the system configuration. However, if you configure Telnet or serial authentication in the admin context, then authentication also applies to these sessions. The admin context AAA server or local user database is used in this instance.

Configure Authentication for CLI and ASDM Access

Before you begin

- Configure Telnet, SSH, or HTTP access.
- For external authentication, configure a AAA server group. For local authentication, add users to the local database.
- HTTP management authentication does not support the SDI protocol for a AAA server group.
- This feature does not affect SSH public key authentication for local usernames with the **ssh authentication** command. The ASA implicitly uses the local database for public key authentication. This feature only affects usernames with passwords. If you want to allow either public key authentication or password use by a local user, then you need to explicitly configure local authentication with this procedure to allow password access.

Procedure

Authenticate users for management access.

aaa authentication {**telnet** | **ssh** | **http** | **serial**} **console** {**LOCAL** | *server_group* [**LOCAL**]}

Example:

```
ciscoasa(config)# aaa authentication ssh console radius_1 LOCAL
ciscoasa(config)# aaa authentication http console radius_1 LOCAL
ciscoasa(config)# aaa authentication serial console LOCAL
```

The **telnet** keyword controls Telnet access. The **ssh** keyword controls SSH access (password only; public key authentication implicitly uses the local database). The **http** keyword controls ASDM access. The **serial** keyword controls console port access. For the Firepower 2100 in Platform mode, this keyword affects the virtual console accessed from FXOS using the **connect asa** command.

If you use a AAA server group for authentication, you can configure the ASA to use the local database as a fallback method if the AAA server is unavailable. Specify the server group name followed by **LOCAL** (which is case sensitive). We recommend that you use the same username and password in the local database as the AAA server, because the ASA prompt does not give any indication which method is being used. You can alternatively use the local database as your primary method of authentication (with no fallback) by entering **LOCAL** alone.

Configure Enable Authentication (Privileged EXEC Mode)

You can authentication users when they enter the **enable** command.

Before you begin

See About Enable Authentication, on page 1182.

Procedure

Choose one of the following options for authenticating users:

- To authenticate users with a AAA server or the local database, enter the following command:

  aaa authentication enable console {**LOCAL** | *server_group* [**LOCAL**]}

  Example:

  ```
  ciscoasa(config)# aaa authentication enable console LOCAL
  ```

  The user is prompted for the username and password.

  If you use a AAA server group for authentication, you can configure the ASA to use the local database as a fallback method if the AAA server is unavailable. Specify the server group name followed by **LOCAL** (which is case sensitive). We recommend that you use the same username and password in the local database as the AAA server, because the ASA prompt does not give any indication of which method is being used.

  You can alternatively use the local database as your primary method of authentication (with no fallback) by entering **LOCAL** alone.

- To log in as a user from the local database, enter the following command:

login

Example:

```
ciscoasa# login
```

The ASA prompts for your username and password. After you enter your password, the ASA places you in the privilege level that the local database specifies.

Users can log in with their own username and password to access privileged EXEC mode, so you do not have to provide the system enable password to everyone. To allow users to access privileged EXEC mode (and all commands) when they log in, set the user privilege level to 2 (the default) through 15. If you configure local command authorization, then the user can only enter commands assigned to that privilege level or lower.

Configure ASDM Certificate Authentication (Java Web Start)

You can require certificate authentication, with or without AAA authentication. The ASA validates the certificate against the PKI trustpoint.

Before you begin

- Supported with Java Web Start only. The ASDM Launcher is not supported.
- Supported in single, routed mode only.
- You need to have a client (.pfx format) and server certificate (.p12 format). Remember the decryption passphrase for importing the certificate.

Procedure

Step 1 Add an identity certificate to the ASA. See Configure Trustpoints, on page 799

Step 2 Enable certificate authentication:

http authentication-certificate *interface_name*[**match** *certificate_map_name*]

Example:

```
ciscoasa(config)# crypto ca certificate map map1 10
ciscoasa(config-ca-cert-map)# subject-name eq www.example.com
ciscoasa(config)# http authentication-certificate outside match map1
```

You configure certificate authentication for each interface, so that connections on a trusted/inside interface do not have to provide a certificate. You can use the command multiple times to enable certificate authentication on multiple interfaces.

To require the certificate to match a certificate map, specify the **match** keyword and the map name. Configure the map using the **crypto ca certificate map** command.

Step 3 (Optional) Set the attribute used by ASDM to derive the username from the certificate:

http username-from-certificate{*primary-attr* [*secondary-attr*] | **use-entire-name** | **use-script**} [**pre-fill-username**]

Example:

```
ciscoasa(config)# http username-from-certificate CN pre-fill-username
```

By default, ASDM uses CN OU attributes.

- The *primary-attr* argument specifies the attribute to be used to derive the username. The *secondary-attr* argument specifies an additional attribute to use with the primary attribute to derive the username. You can use the following attributes:
  - C—Country
  - CN—Common Name
  - DNQ—DN qualifier
  - EA—Email Address
  - GENQ—Generational qualifier
  - GN—Given Name
  - I—Initials
  - L—Locality
  - N—Name
  - O—Organization
  - OU—Organizational Unit
  - SER—Serial Number
  - SN—Surname
  - SP—State/Province
  - T—Title
  - UID User ID
  - UPN—User Principal Name
- The **use-entire-name** keyword uses the entire DN name.
- The **use-script** keyword uses a Lua script generated by ASDM.
- The **pre-fill-username** keyword pre-fills the username when prompted for authentication. If the username is different from the one you initially typed in, a new dialog box appears with the username pre-filled. You can then enter the password for authentication.

Control CLI and ASDM Access with Management Authorization

The ASA lets you distinguish between administrative and remote-access users when they authenticate. User role differentiation can prevent remote access VPN and network access users from establishing an administrative connection to the ASA.

Before you begin

RADIUS or LDAP (mapped) users

When users are authenticated through LDAP, the native LDAP attributes and their values can be mapped to ASA attributes to provide specific authorization features. Configure Cisco VSA CVPN3000-Privilege-Level with a value between 0 and 15. and then map the LDAP attributes to Cisco VAS CVPN3000-Privilege-Level using the **ldap map-attributes** command.

The RADIUS IETF **service-type** attribute, when sent in an access-accept message as the result of a RADIUS authentication and authorization request, is used to designate which type of service is granted to the authenticated user

The RADIUS Cisco VSA **privilege-level** attribute (Vendor ID 3076, sub-ID 220), when sent in an access-accept message, is used to designate the level of privilege for the user.

TACACS+ users

Authorization is requested with “service=shell,” and the server responds with PASS or FAIL.

Local users

Set the **service-type** command for a given username. By default, the service-type is admin, which allows full access to any services specified by the **aaa authentication console**command.

Management Authorization Attributes

See the following table for AAA server types and valid values for management authorization. The ASA uses these values to determine the level of management access.

Management Level	RADIUS/LDAP (Mapped) Attributes	TACACS+ Attributes	Local Database Attributes
Full Access—Allows full access to any services specified by the **aaa authentication console** commands	Service-Type 6 (Administrative), Privilege-Level 1	PASS, privilege level 1	admin
Partial Access—Allows access to the CLI or ASDM when you configure the **aaa authentication console** commands. However, if you configure **enable** authentication with the **aaa authentication enable console** command, then the CLI user cannot access privileged EXEC mode using the **enable** command.	Service-Type 7 (NAS prompt), Privilege-Level 2 and higher The Framed (2) and Login (1) service types are treated the same way.	PASS, privilege level 2 and higher	nas-prompt

Management Level	RADIUS/LDAP (Mapped) Attributes	TACACS+ Attributes	Local Database Attributes
No Access—Denies management access. The user cannot use any services specified by the **aaa authentication console** commands(excluding the **serial** keyword; serial access is allowed). Remote access (IPsec and SSL) users can still authenticate and terminate their remote access sessions. All other service types (Voice, FAX, and so on) are treated the same way.	Service-Type 5 (Outbound)	FAIL	remote-access

Additional Guidelines

- Serial console access is not included in management authorization.
- You must also configure AAA authentication for management access to use this feature. See Configure Authentication for CLI and ASDM Access, on page 1183.
- If you use external authentication, you must pre-configure a AAA server group before you enable this feature.
- HTTP authorization is supported in single, routed mode only.

Procedure

Step 1 Enable management authorization for Telnet and SSH:

aaa authorization exec {**authentication-server** | **LOCAL**} [**auto-enable**]

The **auto-enable** keyword lets administrators who have sufficient authorization privileges enter privileged EXEC mode automatically when they log in.

Example:

```
ciscoasa(config)# aaa authentication ssh console RADIUS
ciscoasa(config)# aaa authorization exec authentication-server auto-enable
```

Step 2 Enable management authorization for HTTPS (ASDM):

aaa authorization http console {**authentication-server** | **LOCAL**}

Example:

```
ciscoasa(config)# aaa authentication http console RADIUS
ciscoasa(config)# aaa authorization http console authentication-server
```

Step 3

Examples

The following example shows how to define an LDAP attribute map. In this example, the security policy specifies that users being authenticated through LDAP map the user record fields or parameters title and company to the IETF-RADIUS service-type and privilege-level, respectively.

```
ciscoasa(config)# ldap attribute-map admin-control
ciscoasa(config-ldap-attribute-map)# map-name title IETF-RADIUS-Service-Type
ciscoasa(config-ldap-attribute-map)# map-name company
```

The following example applies an LDAP attribute map to an LDAP AAA server:

```
ciscoasa(config)# aaa-server ldap-server (dmz1) host 10.20.30.1
ciscoasa(config-aaa-server-host)# ldap attribute-map admin-control
```

Configure Command Authorization

If you want to control access to commands, the ASA lets you configure command authorization, where you can determine which commands that are available to a user. By default when you log in, you can access user EXEC mode, which offers only minimal commands. When you enter the **enable** command (or the **login** command when you use the local database), you can access privileged EXEC mode and advanced commands, including configuration commands.

You can use one of two command authorization methods:

- Local privilege levels
- TACACS+ server privilege levels

About Command Authorization

You can enable command authorization so only authorized users can enter commands.

Supported Command Authorization Methods

You can use one of two command authorization methods:

- Local privilege levels—Configure the command privilege levels on the ASA. When a local, RADIUS, or LDAP (if you map LDAP attributes to RADIUS attributes) user authenticates for CLI access, the ASA places that user in the privilege level that is defined by the local database, RADIUS, or LDAP server. The user can access commands at the assigned privilege level and below. Note that all users access user EXEC mode when they first log in (commands at level 0 or 1). The user needs to authenticate again with the **enable** command to access privileged EXEC mode (commands at level 2 or higher), or they can log in with the **login** command (local database only).

Note You can use local command authorization without any users in the local database and without CLI or **enable** authentication. Instead, when you enter the **enable** command, you enter the system enable password, and the ASA places you in level 15. You can then create enable passwords for every level, so that when you enter **enable** *n* (2 to 15), the ASA places you in level *n*. These levels are not used unless you enable local command authorization.

- TACACS+ server privilege levels—On the TACACS+ server, configure the commands that a user or group can use after authenticating for CLI access. Every command that a user enters at the CLI is validated with the TACACS+ server.

Security Contexts and Command Authorization

AAA settings are discrete per context, not shared among contexts.

When configuring command authorization, you must configure each security context separately. This configuration provides you the opportunity to enforce different command authorizations for different security contexts.

When switching between security contexts, administrators should be aware that the commands permitted for the username specified when they login may be different in the new context session or that command authorization may not be configured at all in the new context. Failure to understand that command authorizations may differ between security contexts could confuse an administrator.

Note The system execution space does not support AAA commands; therefore, command authorization is not available in the system execution space.

Command Privilege Levels

By default, the following commands are assigned to privilege level 0. All other commands are assigned to privilege level 15.

- **show checksum**
- **show curpriv**
- **enable**
- **help**
- **show history**
- **login**
- **logout**
- **pager**
- **show pager**
- **clear pager**

- **quit**
- **show version**

If you move any configure mode commands to a lower level than 15, be sure to move the **configure** command to that level as well, otherwise, the user cannot enter configuration mode.

Configure Local Command Authorization

Local command authorization lets you assign commands to one of 16 privilege levels (0 to 15). By default, each command is assigned either to privilege level 0 or 15. You can define each user to be at a specific privilege level, and each user can enter any command at the assigned privilege level or below. The ASA supports user privilege levels defined in the local database, a RADIUS server, or an LDAP server (if you map LDAP attributes to RADIUS attributes).

Procedure

Step 1 Assign a command to a privilege level.

privilege [**show** | **clear** | **cmd**] **level** *level* [**mode** {**enable** | **cmd**}] **command** *command*

Example:

```
ciscoasa(config)# privilege show level 5 command filter
```

Repeat this command for each command that you want to reassign.

The options in this command are the following:

- **show** | **clear** | **cmd**—These optional keywords let you set the privilege only for the show, clear, or configure form of the command. The configure form of the command is typically the form that causes a configuration change, either as the unmodified command (without the **show** or **clear** prefix) or as the **no** form. If you do not use one of these keywords, all forms of the command are affected.
- **level** *level*—A level between 0 and 15.
- **mode** {**enable** | **configure**}—If a command can be entered in user EXEC or privileged EXEC mode as well as configuration mode, and the command performs different actions in each mode, you can set the privilege level for these modes separately:
  - **enable**—Specifies both user EXEC mode and privileged EXEC mode.
  - **configure**—Specifies configuration mode, accessed using the **configure terminal** command.
- **command** *command*—The command you are configuring. You can only configure the privilege level of the *main* command. For example, you can configure the level of all **aaa** commands, but not the level of the **aaa authentication** command and the **aaa authorization** command separately.

Step 2 (Optional) Enable AAA users for command authorization. Without this command, the ASA only supports privilege levels for local database users and defaults all other types of users to level 15.

aaa authorization exec authentication-server [auto-enable]

Example:

```
ciscoasa(config)# aaa authorization exec authentication-server
```

This command also enables management authorization. See Control CLI and ASDM Access with Management Authorization, on page 1187.

Step 3 Enable the use of local command privilege levels:

aaa authorization command LOCAL

Example:

```
ciscoasa(config)# aaa authorization command LOCAL
```

When you set command privilege levels, command authorization does not occur unless you configure command authorization with this command.

Examples

The **filter** command has the following forms:

- **filter** (represented by the **configure** option)
- **show running-config filter**
- **clear configure filter**

You can set the privilege level separately for each form, or set the same privilege level for all forms by omitting this option. The following example shows how to set each form separately:

```
ciscoasa(config)# privilege show level 5 command filter
ciscoasa(config)# privilege clear level 10 command filter
ciscoasa(config)# privilege cmd level 10 command filter
```

Alternatively, the following example shows how to set all filter commands to the same level:

```
ciscoasa(config)# privilege level 5 command filter
```

The **show privilege** command separates the forms in the display.

The following example shows the use of the **mode** keyword. The **enable** command must be entered from user EXEC mode, while the **enable password** command, which is accessible in configuration mode, requires the highest privilege level:

```
ciscoasa(config)# privilege cmd level 0 mode enable command enable
ciscoasa(config)# privilege cmd level 15 mode cmd command enable
ciscoasa(config)# privilege show level 15 mode cmd command enable
```

The following example shows an additional command, the **configure** command, which uses the **mode** keyword:

```
ciscoasa(config)# privilege show level 5 mode cmd command configure
ciscoasa(config)# privilege clear level 15 mode cmd command configure
ciscoasa(config)# privilege cmd level 15 mode cmd command configure
ciscoasa(config)# privilege cmd level 15 mode enable command configure
```

Note This last line is for the **configure terminal** command.

Configure Commands on the TACACS+ Server

You can configure commands on a Cisco Secure Access Control Server (ACS) TACACS+ server as a shared profile component, for a group, or for individual users. For third-party TACACS+ servers, see your server documentation for more information about command authorization support.

See the following guidelines for configuring commands in Cisco Secure ACS Version 3.1; many of these guidelines also apply to third-party servers:

- The ASA sends the commands to be authorized as shell commands, so configure the commands on the TACACS+ server as shell commands.

Note Cisco Secure ACS might include a command type called “pix-shell.” Do not use this type for ASA command authorization.

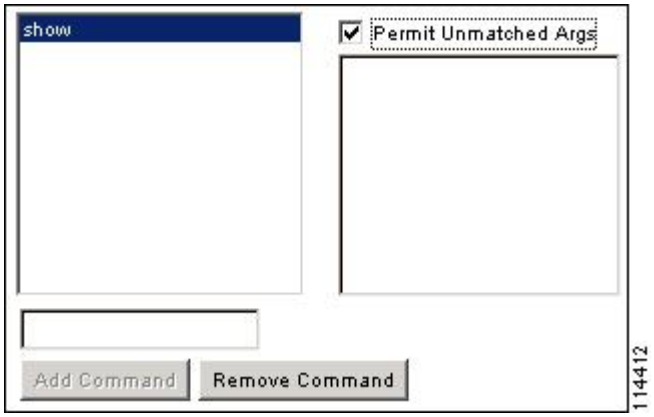
- The first word of the command is considered to be the main command. All additional words are considered to be arguments, which need to be preceded by **permit** or **deny**.

  For example, to allow the **show running-configuration aaa-server** command, add **show running-configuration** to the command field, and type **permit aaa-server** in the arguments field.

- You can permit all arguments of a command that you do not explicitly deny by checking the **Permit Unmatched Args** check box.

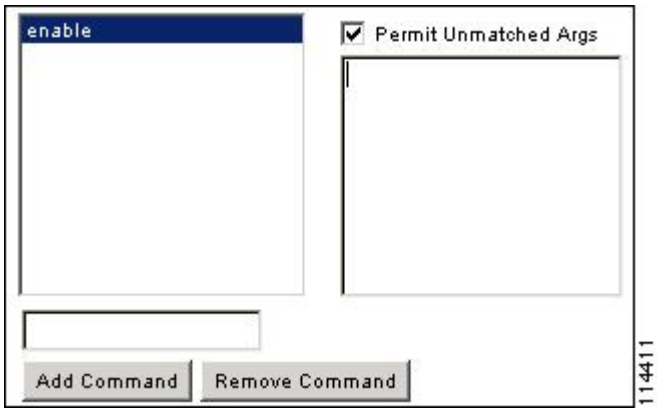
  For example, you can configure just the **show** command, then all the **show** commands are allowed. We recommend using this method so that you do not have to anticipate every variant of a command, including abbreviations and a question mark, which shows CLI usage (see the following figure).

Figure 65: Permitting All Related Commands



- For commands that are a single word, you *must* permit unmatched arguments, even if there are no arguments for the command, for example **enable** or **help** (see the following figure).

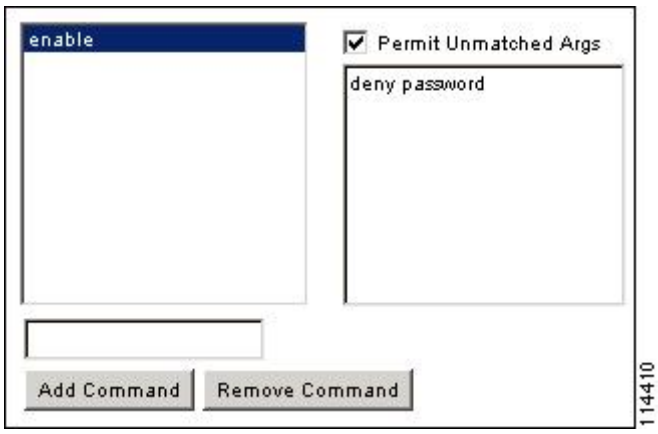
Figure 66: Permitting Single Word Commands



- To disallow some arguments, enter the arguments preceded by **deny**.

  For example, to allow **enable**, but not **enable password**, enter **enable** in the commands field, and **deny password** in the arguments field. Be sure to check the **Permit Unmatched Args** check box so that **enable** alone is still allowed (see the following figure).

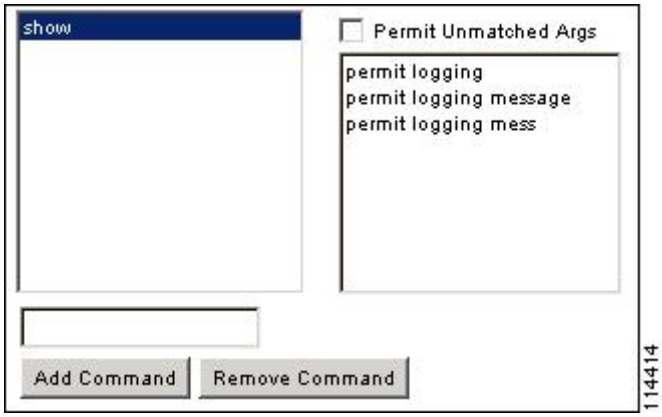
Figure 67: Disallowing Arguments



- When you abbreviate a command at the command line, the ASA expands the prefix and main command to the full text, but it sends additional arguments to the TACACS+ server as you enter them.

  For example, if you enter **sh log**, then the ASA sends the entire command to the TACACS+ server, **show logging**. However, if you enter **sh log mess**, then the ASA sends **show logging mess** to the TACACS+ server, and not the expanded command **show logging message**. You can configure multiple spellings of the same argument to anticipate abbreviations (see the following figure).

Figure 68: Specifying Abbreviations



- We recommend that you allow the following basic commands for all users:
  - **show checksum**
  - **show curpriv**
  - **enable**
  - **help**
  - **show history**
  - **login**
  - **logout**

- **pager**
- **show pager**
- **clear pager**
- **quit**
- **show version**

Configure TACACS+ Command Authorization

If you enable TACACS+ command authorization, and a user enters a command at the CLI, the ASA sends the command and username to the TACACS+ server to determine if the command is authorized.

Before you enable TACACS+ command authorization, be sure that you are logged into the ASA as a user that is defined on the TACACS+ server, and that you have the necessary command authorization to continue configuring the ASA. For example, you should log in as an admin user with all commands authorized. Otherwise, you could become unintentionally locked out.

Do not save your configuration until you are sure that it works the way you want. If you get locked out because of a mistake, you can usually recover access by restarting the ASA.

Be sure that your TACACS+ system is completely stable and reliable. The necessary level of reliability typically requires that you have a fully redundant TACACS+ server system and fully redundant connectivity to the ASA. For example, in your TACACS+ server pool, include one server connected to interface 1, and another to interface 2. You can also configure local command authorization as a fallback method if the TACACS+ server is unavailable.

To configure command authorization using a TACACS+ server, perform the following steps:

Procedure

Enter the following command:

aaa authorization command *tacacs+_server_group* [**LOCAL**]

Example:

```
ciscoasa(config)# aaa authorization command tacacs+_server_group [LOCAL]
```

You can configure the ASA to use the local database as a fallback method if the TACACS+ server is unavailable. To enable fallback, specify the server group name followed by **LOCAL** (**LOCAL** is case sensitive). We recommend that you use the same username and password in the local database as the TACACS+ server because the ASA prompt does not give any indication of which method is being used. Be sure to configure users in the local database and command privilege levels.

Configure a Password Policy for Local Database Users

When you configure authentication for CLI or ASDM access using the local database, you can configure a password policy that requires a user to change their password after a specified amount of time and also requires password standards such as a minimum length and the minimum number of changed characters.

The password policy only applies to administrative users using the local database, and not to other types of traffic that can use the local database, such as VPN or AAA for network access, and not to users authenticated by a AAA server.

After you configure the password policy, when you change a password (either your own or another user's), the password policy applies to the new password. Any existing passwords are grandfathered in. The new policy applies to changing the password with the **username** command as well as the **change-password** command.

Before you begin

- Configure AAA authentication for CLI or ASDM access using the local database.
- Specify usernames in the local database.

Procedure

Step 1 (Optional) Set the interval in days after which passwords expire for remote users.

password-policy lifetime *days*

Example:

```
ciscoasa(config)# password-policy lifetime 180
```

Note
Users at the console port are never locked out because of password expiration.

Valid values are between 0 and 65536 days. The default value is 0 days, a value indicating that passwords will never expire.

Seven days before the password expires, a warning message appears. After the password expires, system access is denied to remote users. To gain access after expiration, do one of the following:

- Have another administrator change your password with the **username** command.
- Log in to the physical console port to change your password.

Step 2 (Optional) Set the minimum number of characters that you must change between new and old passwords.

password-policy minimum-changes *value*

Example:

```
ciscoasa(config)# password-policy minimum-changes 2
```

Valid values are between 0 and 64 characters. The default value is 0.

Character matching is position independent, meaning that new password characters are considered changed only if they do not appear anywhere in the current password.

Step 3 (Optional) Set the minimum length of passwords.

password-policy minimum-length *value*

Example:

```
ciscoasa(config)# password-policy minimum-length 8
```

Valid values are between 3 and 64 characters. We recommend a minimum password length of 8 characters.

Step 4 (Optional) Set the minimum number of upper case characters that passwords must have.

password-policy minimum-uppercase *value*

Example:

```
ciscoasa(config)# password-policy  minimum-uppercase 3
```

Valid values are between 0 and 64 characters. The default value is 0, which means there is no minimum.

Step 5 (Optional) Set the minimum number of lower case characters that passwords must have.

password-policy minimum-lowercase *value*

Example:

```
ciscoasa(config)# password-policy minimum-lowercase 6
```

Valid values are between 0 and 64 characters. The default value is 0, which means there is no minimum.

Step 6 (Optional) Set the minimum number of numeric characters that passwords must have.

password-policy minimum-numeric *value*

Example:

```
ciscoasa(config)# password-policy minimum-numeric 1
```

Valid values are between 0 and 64 characters. The default value is 0, which means there is no minimum.

Step 7 (Optional) Set the minimum number of special characters that passwords must have.

password-policy minimum-special *value*

Example:

```
ciscoasa(config)# password-policy minimum-special 2
```

Valid values are between 0 and 64 characters. Special characters include the following: !, @, #, $, %, ^, &, *, '(' and ')'. The default value is 0, which means there is no minimum.

Step 8 Prohibit the reuse of a password:

password-policy reuse-interval *value*

Example:

```
ciscoasa(config)# password-policy reuse-interval 5
```

You can prohibit the reuse of a password that matches previously used passwords, between 2 and 7 previous passwords. The previous passwords are stored in the configuration under each username in encrypted form using the **password-history** command; this command is not user-configurable.

Step 9 Prohibit a password that matches a username:

password-policy username-check

Step 10 (Optional) Set whether users must change their password using the **change-password** command, instead of letting users change their password with the **username** command.

password-policy authenticate enable

Example:

```
ciscoasa(config)# password-policy authenticate enable
```

The default setting is disabled: a user can use either method to change their password.

If you enable this feature and try to change your password with the **username** command, the following error message appears:

```
ERROR: Changing your own password is prohibited
```

You also cannot delete your own account with the **clear configure username** command. If you try, the following error message appears:

```
ERROR: You cannot delete all usernames because you are not allowed to delete yourself
```

Change Your Password

If you configure a password lifetime in the password policy, you need to change your password to a new one when the old password expires. This password change method is required if you enable password policy authentication. If password policy authentication is not enabled, then you can use this method, or you can change your user account directly.

To change your username password, perform the following steps:

Procedure

Enter the following command:

change-password [**old-password** *old_password* [**new-password** *new_password*]]

Example:

```
ciscoasa# change-password old-password j0hncr1chton new-password a3rynsun
```

If you do not enter the old and new passwords in the command, the ASA prompts you for input.

Enable and View the Login History

By default, the login history is saved for 90 days. You can disable this feature or change the duration, up to 365 days.

Before you begin

- The login history is only saved per unit; in failover and clustering environments, each unit maintains its own login history only.
- Login history data is not maintained over reloads.
- This feature applies to usernames in the local database or from a AAA server when you enable local AAA authentication for one or more of the CLI management methods (SSH, Telnet, serial console). ASDM logins are not saved in the history.

Procedure

Step 1 Set the login history duration:

aaa authentication login-history duration *days*

Example:

```
ciscoasa(config)# aaa authentication login-history duration 365
```

You can set the *days* between 1 and 365. The default is 90. To disable the login history, enter **no aaa authentication login-history**.

When a user logs in, they see their own login history, such as this SSH example:

```
cugel@10.86.194.108's password:
The privilege level for user cugel is 15.  The privilege level at the previous login was
2.
User cugel logged in to ciscoasa at 21:04:10 UTC Dec 14 2016
Last login: 21:01:44 UTC Dec 14 2016 from ciscoasa console
Successful logins over the last 90 days: 6
Authentication failures since the last login: 0
Type help or '?' for a list of available commands.
ciscoasa>
```

Step 2 View the login history:

show aaa login-history [**user** *name*]

Example:

```
ciscoasa(config)# show aaa login-history
Login history for user:    turjan
Logins in last   1 days:   1
Last successful login:     16:44:32 UTC Jul 23 2018 from console
Failures since last login: 0
Last failed login:         None
Privilege level:                                  14
Privilege level changed from 11 to 14 at:         14:07:30 UTC Aug 21 2018
```

Configure Management Access Accounting

You can send accounting messages to the TACACS+ accounting server when you enter any command other than **show** commands at the CLI. You can configure accounting when users log in, when they enter the **enable** command, or when they issue commands.

For command accounting, you can only use TACACS+ servers.

To configure management access and enable command accounting, perform the following steps:

Procedure

Step 1 Enter the following command:

aaa accounting {**serial** | **telnet** | **ssh** | **enable**} **console** *server-tag*

Example:

```
ciscoasa(config)# aaa accounting telnet console group_1
```

Valid server group protocols are RADIUS and TACACS+.

Step 2 Enable command accounting. Only TACACS+ servers support command accounting.

aaa accounting command [**privilege** *level*] *server-tag*

Example:

```
ciscoasa(config)# aaa accounting command privilege 15 group_1
```

The **privilege** *level* keyword-argument pair is the minimum privilege level and the *server-tag* argument is the name of the TACACS+ server group to which the ASA should send command accounting messages.

Recover from a Lockout

In some circumstances, when you turn on command authorization or CLI authentication, you can be locked out of the ASA CLI. You can usually recover access by restarting the ASA. However, if you already saved your configuration, you might be locked out.

The following table lists the common lockout conditions and how you might recover from them.

Table 54: CLI Authentication and Command Authorization Lockout Scenarios

Feature	Lockout Condition	Description	Workaround: Single Mode	Workaround: Multiple Mode
Local CLI authentication	No users have been configured in the local database.	If you have no users in the local database, you cannot log in, and you cannot add any users.	Log in and reset the passwords and **aaa** commands.	Session into the ASA from the switch. From the system execution space, you can change to the context and add a user.
TACACS+ command authorization TACACS+ CLI authentication RADIUS CLI authentication	The server is down or unreachable and you do not have the fallback method configured.	If the server is unreachable, then you cannot log in or enter any commands.	1. Log in and reset the passwords and AAA commands. 2. Configure the local database as a fallback method so you do not get locked out when the server is down.	1. If the server is unreachable because the network configuration is incorrect on the ASA, session into the ASA from the switch. From the system execution space, you can change to the context and reconfigure your network settings. 2. Configure the local database as a fallback method so that you do not get locked out when the server is down.
TACACS+ command authorization	You are logged in as a user without enough privileges or as a user that does not exist.	You enable command authorization, but then find that the user cannot enter any more commands.	Fix the TACACS+ server user account. If you do not have access to the TACACS+ server and you need to configure the ASA immediately, then log into the maintenance partition and reset the passwords and **aaa** commands.	Session into the ASA from the switch. From the system execution space, you can change to the context and complete the configuration changes. You can also disable command authorization until you fix the TACACS+ configuration.
Local command authorization	You are logged in as a user without enough privileges.	You enable command authorization, but then find that the user cannot enter any more commands.	Log in and reset the passwords and **aaa** commands.	Session into the ASA from the switch. From the system execution space, you can change to the context and change the user level.

Monitoring Device Access

See the following commands for monitoring device access:

- **show running-config all privilege all**

This command shows privilege levels for all commands.

For the **show running-config all privilege all** command, the ASA displays the current assignment of each CLI command to a privilege level. The following is sample output from this command:

```
ciscoasa(config)# show running-config all  privilege all
privilege show level 15 command aaa
privilege clear level 15 command aaa
privilege configure level 15 command aaa
privilege show level 15 command aaa-server
privilege clear level 15 command aaa-server
privilege configure level 15 command aaa-server
privilege show level 15 command access-group
privilege clear level 15 command access-group
privilege configure level 15 command access-group
privilege show level 15 command access-list
privilege clear level 15 command access-list
privilege configure level 15 command access-list
privilege show level 15 command activation-key
privilege configure level 15 command activation-key
...
```

- **show running-config privilege level** *level*

  This command shows commands for a specific privilege level. The level argument is an integer between 0 and 15.

  The following example shows the command assignments for privilege level 10:

```
ciscoasa(config)# show running-config all privilege level 10
privilege show level 10 command aaa
```

- **show running-config privilege command** *command*

  This command shows the privilege level of a specific command.

  The following example shows the command assignments for the **access-list** command:

```
ciscoasa(config)# show running-config all privilege command access-list
privilege show level 15 command access-list
privilege clear level 15 command access-list
privilege configure level 15 command access-list
```

- **show curpriv**

  This command shows the currently logged-in user.

  The following is sample output from the **show curpriv** command:

```
ciscoasa# show curpriv
Username: admin
Current privilege level: 15
Current Mode/s: P_PRIV
```

  The following table describes the **show curpriv** command output.

Table 55: show curpriv Command Output Description

Field	Description
Username	Username. If you are logged in as the default user, the name is enable_1 (user EXEC) or enable_15 (privileged EXEC).
Current privilege level	Levels range from 0 to 15. Unless you configure local command authorization and assign commands to intermediate privilege levels, levels 0 and 15 are the only levels that are used.
Current Modes	The available access modes are the following: • P_UNPR—User EXEC mode (levels 0 and 1) • P_PRIV—Privileged EXEC mode (levels 2 to 15) • P_CONF—Configuration mode

- **show quota management-session** [**ssh** | **telnet** | **http** | **username** *user*]

  This command shows the current sessions in use.

  The following is sample output from the **show quota management-session** command:

```
ciscoasa(config)#show quota management-session

#Sessions               ConnectionType                  Username
1                              SSH                      cisco
2                              TELNET                   cisco
1                              SSH                      cisco1
```

- **show aaa login-history** [**user** *name*]

  This command shows the login history per user.

  The following is sample output from the **show aaa login-history** command.

```
ciscoasa(config)# show aaa login-history
Login history for user:    turjan
Logins in last   1 days:   1
Last successful login:     16:44:32 UTC Jul 23 2018 from console
Failures since last login: 0
Last failed login:         None
Privilege level:                                   14
Privilege level changed from 11 to 14 at:          14:07:30 UTC Aug 21 2018
```

History for Management Access

Table 56: History for Management Access

Feature Name	Platform Releases	Description
CiscoSSH stack	9.17(1)	The ASA uses a proprietary SSH stack for SSH connections. You can now choose to use the CiscoSSH stack instead, which is based on OpenSSH. The default stack continues to be the ASA stack. Cisco SSH supports: • FIPS compliance • Regular updates, including updates from Cisco and the open source community Note that the CiscoSSH stack does not support: • SSH to a different interface over VPN (management-access) • EdDSA key pair • RSA key pair in FIPS mode If you need these features, you should continue to use the ASA SSH stack. There is a small change to SCP functionality with the CiscoSSH stack: to use the ASA **copy** command to copy a file to or from an SCP server, you have to enable SSH access on the ASA for the SCP server subnet/host using the **ssh** command. New/Modified commands: **ssh stack ciscossh**
Local user lockout changes	9.17(1)	The ASA can lock out local users after a configurable number of failed login attempts. This feature did not apply to users with privilege level 15. Also, a user would be locked out indefinitely until an admin unlocked their account. Now, users will be unlocked after 10 minutes unless an admin uses the **clear aaa local user lockout** command before then. Privilege level 15 users are also now affected by the lockout setting. New/Modified commands: **aaa local authentication attempts max-fail** , **show aaa local user**
SSH and Telnet password change prompt	9.17(1)	The first time a local user logs into the ASA using SSH or Telnet, they are prompted to change their password. They will also be prompted for the first login after an admin changes their password. If the ASA reloads, however, users will not be prompted even if it is their first login. Note that any service that uses the local user database, such as VPN, will also have to use the new password if it was changed during an SSH or Telnet login. New/Modified commands: **show aaa local user**

Feature Name	Platform Releases	Description
SSH security improvements	9.16(1)	SSH now supports the following security improvements: • Host key format—**crypto key generate** {**eddsa** | **ecdsa**}. In addition to RSA, we added support for the EdDSA and ECDSA host keys. The ASA tries to use keys in the following order if they exist: EdDSA, ECDSA, and then RSA. If you explicitly configure the ASA to use the RSA key with the **ssh key-exchange hostkey rsa** command, you must generate a key that is 2048 bits or higher. For upgrade compatibility, the ASA will use smaller RSA host keys only when the default host key setting is used. RSA support will be removed in a later release. • Key exchange algorithms—**ssh key-exchange group** {**ecdh-sha2-nistp256** | **curve25519-sha256**} • Encryption algorithms—**ssh cipher encryption chacha20-poly1305@openssh.com** • SSH version 1 is no longer supported—The **ssh version** command is removed. New/Modified commands: **crypto key generate eddsa**, **crypto key zeroize eddsa**, **show crypto key mypubkey**, **ssh cipher encryption chacha20-poly1305@openssh.com**, **ssh key-exchange group** {**ecdh-sha2-nistp256** | **curve25519-sha256**}, **ssh key-exchange hostkey**, **ssh version**
Management access for SNMP	9.14(2)	When configuring management access over a VPN tunnel, include the IP address of the outside interface in the crypto map access-list as part of the VPN configuration for secure SNMP polling over a site-to-site VPN.
HTTPS idle timeout setting	9.14(1)	You can now set the idle timeout for all HTTPS connections to the ASA, including ASDM, WebVPN, and other clients. Formerly, using the **http server idle-timeout** command, you could only set the ASDM idle timeout. If you set both timeouts, the new command takes precendence. New/Modified commands: **http connection idle-timeout**
SSH encryption ciphers are now listed in order from highest to lowest security for pre-defined lists	9.13(1)	SSH encryption ciphers are now listed in order from highest security to lowest security for pre-defined lists (such as medium or high). In earlier releases, they were listed from lowest to highest, which meant that a low security cipher would be proposed before a high security cipher. New/Modified commands: **ssh cipher encryption**
Setting the SSH key exchange mode is restricted to the Admin context	9.12(2)	You must set the SSH key exchange in the Admin context; this setting is inherited by all other contexts. New/Modified commands: **ssh key-exchange**

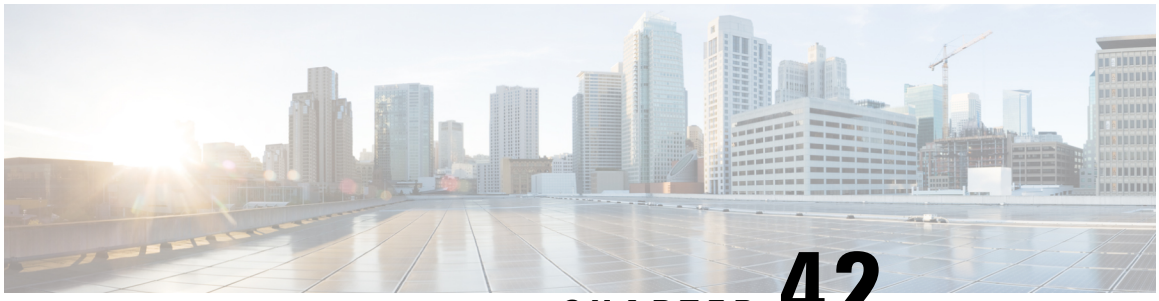
Feature Name	Platform Releases	Description
enable password change now required on login	9.12(1)	The default **enable** password is blank. When you try to access privileged EXEC mode on the ASA, you are now required to change the password to a value of 3 characters or longer. You cannot keep it blank. The **no enable password** command is no longer supported. At the CLI, you can access privileged EXEC mode using the **enable** command, the **login** command (with a user at privilege level 2+), or an SSH or Telnet session when you enable **aaa authorization exec auto-enable**. All of these methods require you to set the enable password. This password change requirement is not enforced for ASDM logins. In ASDM, by default you can log in without a username and with the **enable** password. New/Modified commands: **enable password**
Configurable limitation of admin sessions	9.12(1)	You can configure the maximum number of aggregate, per user, and per-protocol administrative sessions. Formerly, you could configure only the aggregate number of sessions. This feature does not affect console sessions. Note that in multiple context mode, you cannot configure the number of HTTPS sessions, where the maximum is fixed at 5 sessions. The **quota management-session** command is also no longer accepted in the system configuration, and is instead available in the context configuration. The maximum aggregate sessions is now 15; if you configured 0 (unlimited) or 16+, then when you upgrade, the value is changed to 15. New/Modified commands: **quota management-session**, **show quota management-session**
Notifications for administrative privilege level changes	9.12(1)	When you authenticate for enable access (**aaa authentication enable console**) or allow privileged EXEC access directly (**aaa authorization exec auto-enable**), then the ASA now notifies users if their assigned access level has changed since their last login. New/Modified commands: **show aaa login-history**
SSH stronger security	9.12(1)	See the following SSH security improvements: • Diffie-Hellman Group 14 SHA256 key exchange support. This setting is now the default. The former default was Group 1 SHA1. • HMAC-SHA256 integrity cipher support. The default is now the high security set of ciphers (hmac-sha2-256 only). The former default was the medium set. New/Modified commands: **ssh cipher integrity** , **ssh key-exchange group dh-group14-sha256**

Feature Name	Platform Releases	Description
Allow non-browser-based HTTPS clients to access the ASA	9.12(1)	You can allow non-browser-based HTTPS clients to access HTTPS services on the ASA. By default, ASDM, CSM, and REST API are allowed. New/Modified commands: **http server basic-auth-client**
RSA key pair supports 3072-bit keys	9.9(2)	You can now set the modulus size to 3072. New or modified command: **crypto key generate rsa modulus**
VPN management access on Bridged Virtual Interfaces (BVIs)	9.9(2)	You can now enable management services, such as **telnet**, **http**, and **ssh**, on a BVI if VPN **management-access** has been enabled on that BVI. For non-VPN management access, you should continue to configure these services on the bridge group member interfaces. New or Modified commands: **https**, **telnet**, **ssh**, **management-access**
SSH version 1 has been deprecated	9.9(1)	SSH version 1 has been deprecated, and will be removed in a future release. The default setting has changed from both SSH v1 and v2 to just SSH v2. New/Modified commands: **ssh version**
Separate authentication for users with SSH public key authentication and users with passwords	9.6(3)/9.8(1)	In releases prior to 9.6(2), you could enable SSH public key authentication (**ssh authentication**) without also explicitly enabling AAA SSH authentication with the Local user database (**aaa authentication ssh console LOCAL**). In 9.6(2), the ASA required you to explicitly enable AAA SSH authentication. In this release, you no longer have to explicitly enable AAA SSH authentication; when you configure the **ssh authentication** command for a user, local authentication is enabled by default for users with this type of authentication. Moreover, when you explicitly configure AAA SSH authentication, this configuration only applies for for usernames with *passwords*, and you can use any AAA server type (**aaa authentication ssh console radius_1**, for example). For example, some users can use public key authentication using the local database, and other users can use passwords with RADIUS. We did not modify any commands.
Login history	9.8(1)	By default, the login history is saved for 90 days. You can disable this feature or change the duration, up to 365 days. This feature only applies to usernames in the local database when you enable local AAA authentication for one or more of the management methods (SSH, ASDM, Telnet, and so on). We introduced the following commands: **aaa authentication login-history, show aaa login-history**

Feature Name	Platform Releases	Description
Password policy enforcement to prohibit the reuse of passwords, and prohibit use of a password matching a username	9.8(1)	You can now prohibit the reuse of previous passwords for up to 7 generations, and you can also prohibit the use of a password that matches a username. We introduced the following commands: **password-history, password-policy reuse-interval, password-policy username-check**
ASA SSL Server mode matching for ASDM	9.6(2)	For an ASDM user who authenticates with a certificate, you can now require the certificate to match a certificate map. We modified the following command: **http authentication-certificate match**
SSH public key authentication improvements	9.6(2)	In earlier releases, you could enable SSH public key authentication (**ssh authentication**) without also enabling AAA SSH authentication with the Local user database (**aaa authentication ssh console LOCAL**). The configuration is now fixed so that you must explicitly enable AAA SSH authentication. To disallow users from using a password instead of the private key, you can now create a username without any password defined. We modified the following commands: **ssh authentication, username**
ASDM management authorization	9.4(1)	You can now configure management authorization separately for HTTP access vs. Telnet and SSH access. We introduced the following command: **aaa authorization http console**
ASDM username from certificate configuration	9.4(1)	When you enable ASDM certificate authentication (**http authentication-certificate**), you can configure how ASDM extracts the username from the certificate; you can also enable pre-filling the username at the login prompt. We introduced the following command: **http username-from-certificate**
Improved one-time password authentication	9.2(1)	Administrators who have sufficient authorization privileges may enter privileged EXEC mode by entering their authentication credentials once. The **auto-enable** option was added to the **aaa authorization exec** command. We modified the following command: **aaa authorization exec**.
HTTP redirect support for IPV6	9.1(7)/9.6(1)	When you enable HTTP redirect to HTTPS for ASDM access or clientless SSL VPN, you can now redirect traffic sent an to IPv6 address. We added functionality to the following command: **http redirect**

Feature Name	Platform Releases	Description
Configurable SSH encryption and integrity ciphers	9.1(7.4)/9.3(3.6)/9.6(1)	Users can select cipher modes when doing SSH encryption management and can configure HMAC and encryption for varying key exchange algorithms. You might want to change the ciphers to be more or less strict, depending on your application. Note that the performance of secure copy depends partly on the encryption cipher used. By default, the ASA negotiates one of the following algorithms in order: 3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr. If the first algorithm proposed (3des-cbc) is chosen, then the performance is much slower than a more efficient algorithm such as aes128-cbc. To change the proposed ciphers, use **ssh cipher encryption custom aes128-cbc**, for example. We introduced the following commands: **ssh cipher encryption, ssh cipher integrity**.
AES-CTR encryption for SSH	9.1(2)	The SSH server implementation in the ASA now supports AES-CTR mode encryption.
Improved SSH rekey interval	9.1(2)	An SSH connection is rekeyed after 60 minutes of connection time or 1 GB of data traffic. We introduced the following command: **show ssh sessions detail**.
For the ASASM in multiple context mode, support for Telnet and virtual console authentication from the switch.	8.5(1)	Although connecting to the ASASM from the switch in multiple context mode connects to the system execution space, you can configure authentication in the admin context to govern those connections.
Support for administrator password policy when using the local database	8.4(4.1), 9.1(2)	When you configure authentication for CLI or ASDM access using the local database, you can configure a password policy that requires a user to change their password after a specified amount of time and also requires password standards such as a minimum length and the minimum number of changed characters. We introduced the following commands: **change-password**, **password-policy lifetime**, **password-policy minimum changes**, **password-policy minimum-length**, **password-policy minimum-lowercase**, **password-policy minimum-uppercase**, **password-policy minimum-numeric**, **password-policy minimum-special**, **password-policy authenticate enable**, **clear configure password-policy**, **show running-config password-policy**.
Support for SSH public key authentication	8.4(4.1), 9.1(2)	You can enable public key authentication for SSH connections to the ASA on a per-user basis. You can specify a public key file (PKF) formatted key or a Base64 key. The PKF key can be up to 4096 bits. Use PKF format for keys that are too large to for the ASA support of the Base64 format (up to 2048 bits). We introduced the following commands: **ssh authentication**. *PKF key format support is only in 9.1(2) and later.*

Feature Name	Platform Releases	Description
Support for Diffie-Hellman Group 14 for the SSH Key Exchange	8.4(4.1), 9.1(2)	Support for Diffie-Hellman Group 14 for SSH Key Exchange was added. Formerly, only Group 1 was supported. We introduced the following command: **ssh key-exchange**.
Support for a maximum number of management sessions	8.4(4.1), 9.1(2)	You can set the maximum number of simultaneous ASDM, SSH, and Telnet sessions. We introduced the following commands: **quota management-session**, **show running-config quota management-session**, **show quota management-session**.
Increased SSH security; the SSH default username is no longer supported.	8.4(2)	Starting in 8.4(2), you can no longer connect to the ASA using SSH with the pix or asa username and the login password. To use SSH, you must configure AAA authentication using the **aaa authentication ssh console LOCAL** command (CLI) or Configuration > Device Management > Users/AAA > AAA Access > Authentication (ASDM); then define a local user by entering the **username** command (CLI) or choosing Configuration > Device Management > Users/AAA > User Accounts (ASDM). If you want to use a AAA server for authentication instead of the local database, we recommend also configuring local authentication as a backup method.
Management Access	7.0(1)	We introduced this feature. We introduced the following commands: **show running-config all privilege all**, **show running-config privilege level**, **show running-config privilege command**, **telnet**, **telnet timeout**, **ssh**, **ssh timeout**, **http**, **http server enable**, **asdm image disk, banner**, **console timeout**, **icmp**, **ipv6 icmp**, **management access**, **aaa authentication console**, **aaa authentication enable console**, **aaa authentication telnet | ssh console**, **service-type**, **login**, **privilege**, **aaa authentication exec authentication-server**, **aaa authentication command LOCAL**, **aaa accounting serial | telnet | ssh | enable console**, **show curpriv**, **aaa accounting command privilege**.



CHAPTER 42

Software and Configurations

This chapter describes how to manage the ASA software and configurations.

- Upgrade the Software, on page 1213
- Load an Image Using ROMMON (ISA 3000), on page 1213
- Upgrade the ROMMON Image: ISA 3000, on page 1215
- Downgrade Your Software, on page 1216
- Manage Files, on page 1221
- Set the ASA Image, ASDM, and Startup Configuration, on page 1230
- Back Up and Restore Configurations or Other Files, on page 1233
- Hot Swap an SSD on the Secure Firewall 3100, on page 1250
- History for Software and Configurations, on page 1253

Upgrade the Software

See the Cisco ASA Upgrade Guide for full upgrade procedures.

Load an Image Using ROMMON (ISA 3000)

To load a software image onto an ASA from the ROMMON mode using TFTP, perform the following steps.

Procedure

Step 1 Connect to the ASA console port according to the instructions in Access the ISA 3000 Console, on page 13.

Step 2 Power off the ASA, then power it on.

Step 3 During startup, press the **Escape** key when you are prompted to enter ROMMON mode.

Step 4 In ROMMOM mode, define the interface settings to the ASA, including the IP address, TFTP server address, gateway address, software image file, and port, as follows:

```
rommon #1> interface gigabitethernet0/0
rommon #2> address 10.86.118.4
rommon #3> server 10.86.118.21
```

```
rommon #4> gateway 10.86.118.21
rommon #5> file asa961-smp-k8.bin
```

Note
Be sure that the connection to the network already exists.

The **interface** command is ignored on the ASA 5506-X, ASA 5508-X, and ASA 5516-X, and ISA 3000 platforms, and you must perform TFTP recovery on these platforms from the Management 1/1 interface.

Step 5 Validate your settings:

```
rommon #6> set
ROMMON Variable Settings:
  ADDRESS=10.86.118.3
  SERVER=10.86.118.21
  GATEWAY=10.86.118.21
  PORT=GigabitEthernet0/0
  VLAN=untagged
  IMAGE=asa961-smp-k8.bin
  CONFIG=
  LINKTIMEOUT=20
  PKTTIMEOUT=4
  RETRY=20
```

Step 6 Ping the TFTP server:

```
rommon #7> ping server
Sending 20, 100-byte ICMP Echoes to server 10.86.118.21, timeout is 4 seconds:

Success rate is 100 percent (20/20)
```

Step 7 Save the network settings for future use:

```
rommon #8> sync
Updating NVRAM Parameters...
```

Step 8 Load the software image:

```
rommon #9> tftpdnld
ROMMON Variable Settings:
  ADDRESS=10.86.118.3
  SERVER=10.86.118.21
  GATEWAY=10.86.118.21
  PORT=GigabitEthernet0/0
  VLAN=untagged
  IMAGE=asa961-smp-k8.bin
  CONFIG=
  LINKTIMEOUT=20
  PKTTIMEOUT=4
  RETRY=20

tftp asa961-smp-k8.bin@10.86.118.21 via 10.86.118.21

Received 14450688 bytes

Launching TFTP Image...
Cisco ASA Security Appliance admin loader (3.0) #0: Mon Mar 5 16:00:07 MST 2016
```

```
Loading...
```

After the software image is successfully loaded, the ASA automatically exits ROMMON mode.

Step 9 Booting the ASA from ROMMON mode does not preserve the system image across reloads; you must still download the image to flash memory. See the Cisco ASA Upgrade Guide for full upgrade procedures.

Upgrade the ROMMON Image: ISA 3000

Follow these steps to upgrade the ROMMON image for the ISA 3000. For the ASA models, the ROMMON version on your system must be 1.1.8 or greater. We recommend that you upgrade to the latest version.

You can only upgrade to a new version; you cannot downgrade.

Caution The ISA 3000 ROMMON upgrade for 1.0.5 takes twice as long as previous ROMMON versions, approximately 15 minutes. **Do not** power cycle the device during the upgrade. If the upgrade is not complete within 30 minutes or it fails, contact Cisco technical support; **do not** power cycle or reset the device.

Before you begin

Obtain the new ROMMON image from Cisco.com, and put it on a server to copy to the ASA. The ASA supports FTP, TFTP, SCP, HTTP(S), and SMB servers. Download the image from:

- ISA 3000: https://software.cisco.com/download/home/286288493/type

Procedure

Step 1 Copy the ROMMON image to the ASA flash memory. This procedure shows an FTP copy; enter **copy ?** for the syntax for other server types.

copy ftp://[*username***:***password***@**]*server_ip***/asa5500-firmware-***xxxx***.SPA disk0:asa5500-firmware-***xxxx***.SPA**

Step 2 To see your current version, enter the **show module** command and look at the Fw Version in the output for Mod 1 in the MAC Address Range table:

```
ciscoasa# show module
[...]
Mod  MAC Address Range                 Hw Version   Fw Version   Sw Version
---- --------------------------------- ------------ ------------ ---------------
   1 7426.aceb.ccea to 7426.aceb.ccf2  0.3          1.1.5        9.4(1)
 sfr 7426.aceb.cce9 to 7426.aceb.cce9  N/A          N/A
```

Step 3 Upgrade the ROMMON image:

upgrade rommon disk0:asa5500-firmware-*xxxx***.SPA**

Example:

```
ciscoasa# upgrade rommon disk0:asa5500-firmware-1108.SPA
Verifying file integrity of disk0:/asa5500-firmware-1108.SPA

Computed Hash   SHA2: d824bdeecee1308fc64427367fa559e9
                      eefe8f182491652ee4c05e6e751f7a4f
                      5cdea28540cf60acde3ab9b65ff55a9f
                      4e0cfb84b9e2317a856580576612f4af

Embedded Hash   SHA2: d824bdeecee1308fc64427367fa559e9
                      eefe8f182491652ee4c05e6e751f7a4f
                      5cdea28540cf60acde3ab9b65ff55a9f
                      4e0cfb84b9e2317a856580576612f4af


Digital signature successfully validated
File Name                     : disk0:/asa5500-firmware-1108.SPA
Image type                    : Release
    Signer Information
        Common Name           : abraxas
        Organization Unit     : NCS_Kenton_ASA
        Organization Name     : CiscoSystems
    Certificate Serial Number : 553156F4
    Hash Algorithm            : SHA2 512
    Signature Algorithm       : 2048-bit RSA
    Key Version               : A
Verification successful.
Proceed with reload? [confirm]
```

Step 4 Confirm to reload the ASA when you are prompted.

The ASA upgrades the ROMMON image, and then reloads the operating system.

Downgrade Your Software

In many cases, you can downgrade your ASA software and restore a backup configuration from the previous software version. The method of downgrading depends on your ASA platform.

Guidelines and Limitations for Downgrading

See the following guidelines before downgrading:

- **There is no official Zero Downtime Downgrade support for clustering**—However, in some cases, Zero Downtime Downgrading will work. If you encounter issues, you need to reload your cluster units, which results in downtime.
- **VPN tunnels are replicated to the standby unit even if the standby unit is running a version of software that does not support the Ciphersuite that the original tunnel negotiated**—This scenario occurs when downgrading. In this case, disconnect your VPN connection and reconnect.

Incompatible Configuration Removed After Downgrading

When you downgrade to an old version, commands that were introduced in later versions will be removed from the configuration. There is no automated way to check the configuration against the target version before you downgrade. You can view when new commands were added in ASA new features by release.

You can view rejected commands *after* you downgrade using the **show startup-config errors** command. If you can perform a downgrade on a lab device, you can preview the effects using this command before you perform the downgrade on a production device.

In some cases, the ASA migrates commands to new forms automatically when you upgrade, so depending on your version, even if you did not manually configure new commands, the downgrade could be affected by configuration migrations. We recommend that you have a backup of your old configuration that you can use when you downgrade. In the case of upgrading to 8.3, a backup is automatically created (<old_version>_startup_cfg.sav). Other migrations do not create back-ups. See the "Version-Specific Guidelines and Migrations" in the ASA Upgrade guide for more information about automatic command migrations that could affect downgrading.

See also known downgrade issues in Guidelines and Limitations for Downgrading, on page 1216.

For example, an ASA running version 9.8(2) includes the following commands:

```
access-list acl1 extended permit sctp 192.0.2.0 255.255.255.0 198.51.100.0 255.255.255.0
username test1 password $sha512$1234$abcdefghijklmnopqrstuvwxyz privilege 15
snmp-server user snmpuser1 snmpgroup1 v3 engineID abcdefghijklmnopqrstuvwxyz encrypted auth
 md5 12:ab:34 priv aes 128 12:ab:34
```

When you downgrade to 9.0(4), you will see the following errors on startup:

```
access-list acl1 extended permit sctp 192.0.2.0 255.255.255.0 198.51.100.0 255.255.255.0
                                       ^
ERROR: % Invalid input detected at '^' marker.

username test1 password $sha512$1234$abcdefghijklmnopqrstuvwxyz pbkdf2 privilege 15
                                                                ^
ERROR: % Invalid input detected at '^' marker.

snmp-server user snmpuser1 snmpgroup1 v3 engineID abcdefghijklmnopqrstuvwxyz encrypted auth
 md5 12:ab:34 priv aes 128 12:ab:34
                                          ^
ERROR: % Invalid input detected at '^' marker.
```

In this example, support for **sctp** in the **access-list extended** command was added in version 9.5(2), support for **pbkdf2** in the **username** command was added in version 9.6(1), and support for **engineID** in the **snmp-server user** command was added in version 9.5(3).

Downgrade the ASA Appliance

You can downgrade the ASA software version by setting the ASA version to the old version, restoring the backup configuration to the startup configuration, and then reloading. This procedure applies to the following models:

- Firepower 1000
- Secure Firewall 1200
- Firepower 2100

- Secure Firewall 3100
- Secure Firewall 4200

Before you begin

This procedure requires a backup configuration of the ASA before you upgraded, so you can restore the old configuration. If you do not restore the old configuration, you may have incompatible commands representing new or changed features. Any new commands will be rejected when you load the old software version.

Procedure

Step 1 Load the old ASA software version using the upgrade procedure in the ASA upgrade guide for standalone, failover, or clustering deployments. In this case, specify the old ASA version instead of a new version. **Important:** Do *not* reload the ASA yet.

Step 2 At the ASA CLI, copy the backup ASA configuration to the startup configuration. For failover, perform this step on the active unit. This step replicates the command to the standby unit.

copy *old_config_url* **startup-config**

It's important that you do not save the running configuration to the startup configuration using **write memory**; this command will overwrite your backup configuration.

Example:

```
ciscoasa# copy disk0:/9.13.1_cfg.sav startup-config
```

Step 3 Reload the ASA.

ASA CLI

reload

ASDM

Choose **Tools** > **System Reload**.

Downgrade the Firepower 2100 in Platform Mode

You can downgrade the ASA software version by restoring the backup configuration to the startup configuration, setting the ASA version to the old version, and then reloading.

Before you begin

This procedure requires a backup configuration of the ASA before you upgraded, so you can restore the old configuration. If you do not restore the old configuration, you may have incompatible commands representing new or changed features. Any new commands will be rejected when you load the old software version.

Procedure

Step 1 At the ASA CLI, copy the backup ASA configuration to the startup configuration. For failover, perform this step on the active unit. This step replicates the command to the standby unit.

copy *old_config_url* **startup-config**

It's important that you do not save the running configuration to the startup configuration using **write memory**; this command will overwrite your backup configuration.

Example:

```
ciscoasa# copy disk0:/9.12.4_cfg.sav startup-config
```

Step 2 In FXOS, use the Firewall Chassis Manager or FXOS CLI to use the old ASA software version using the upgrade procedure in the ASA upgrade guide for standalone, failover, or clustering deployments. In this case, specify the old ASA version instead of a new version.

Downgrade the Firepower 4100/9300

You can downgrade the ASA software version by restoring the backup configuration to the startup configuration, setting the ASA version to the old version, and then reloading.

Before you begin

- This procedure requires a backup configuration of the ASA before you upgraded, so you can restore the old configuration. If you do not restore the old configuration, you may have incompatible commands representing new or changed features. Any new commands will be rejected when you load the old software version.
- Make sure the old ASA version is compatibile with the current FXOS version. If not, downgrade FXOS as the first step before you restore the old ASA configuration. Just make sure the downgraded FXOS is also compatible with the current ASA version (before you downgrade it). If you cannot achieve compatibility, we suggest you do not perform a downgrade.

Procedure

Step 1 At the ASA CLI, copy the backup ASA configuration to the startup configuration. For failover or clustering, perform this step on the active/control unit. This step replicates the command to the standby/data units.

copy *old_config_url* **startup-config**

It's important that you do not save the running configuration to the startup configuration using **write memory**; this command will overwrite your backup configuration.

Example:

```
ciscoasa# copy disk0:/9.8.4_cfg.sav startup-config
```

Step 2 In FXOS, use the Firewall Chassis Manager or FXOS CLI to use the old ASA software version using the upgrade procedure in the ASA upgrade guide for standalone, failover, or clustering deployments. In this case, specify the old ASA version instead of a new version.

Step 3 If you are also downgrading FXOS, use the Firewall Chassis Manager or FXOS CLI to set the old FXOS software version to be the current version using the upgrade procedure in the ASA upgrade guide for standalone, failover, or clustering deployments.

Downgrade the ISA 3000

The downgrade feature provides a shortcut for completing the following functions on ISA 3000 models:

- Clearing the boot image configuration (**clear configure boot**).
- Setting the boot image to be the old image (**boot system**).
- (Optional) Entering a new activation key (**activation-key**).
- Saving the running configuration to startup (**write memory**). This sets the BOOT environment variable to the old image, so when you reload, the old image is loaded.
- Copying the old configuration backup to the startup configuration (**copy** *old_config_url* **startup-config**).
- Reloading (**reload**).

Before you begin

- This procedure requires a backup configuration of the ASA before you upgraded, so you can restore the old configuration.

Procedure

Downgrade the software and restore the old configuration.

downgrade [**/noconfirm**] *old_image_url old_config_url* [**activation-key** *old_key*]

Example:

```
ciscoasa(config)# downgrade /noconfirm disk0:/asa821-k8.bin disk0:/8_2_1_0_startup_cfg.sav
```

The **/noconfirm** option downgrades without prompting. The *image_url* is the path to the old image on disk0, disk1, tftp, ftp, or smb. The *old_config_url* is the path to the saved, pre-migration configuration. If you need to revert to a pre-8.3 activation key, then you can enter the old activation key.

Manage Files

View Files in Flash Memory

You can view files in flash memory and see information about files.

Procedure

Step 1 View files in flash memory:

dir [**disk***n***:**]

disk0: is the internal memory. Other drive numbers represent external storage such as a USB drive, SSD, or SD card.

Example:

```
hostname# dir

Directory of disk0:/
500    -rw-  4958208     22:56:20 Nov 29 2004  cdisk.bin
2513   -rw-  4634        19:32:48 Sep 17 2004  first-backup
2788   -rw-  21601       20:51:46 Nov 23 2004  backup.cfg
2927   -rw-  8670632     20:42:48 Dec 08 2004  asdmfile.bin
```

Step 2 View extended information about a specific file:

show file information [path:/]*filename*

Example:

```
hostname# show file information cdisk.bin

disk0:/cdisk.bin:
  type is image (XXX) []
  file size is 4976640 bytes version 7.0(1)
```

The file size listed is for example only.

The default path is the root directory of the internal flash memory (disk0:/).

Delete Files from Flash Memory

You can remove files from flash memory that you no longer need.

Procedure

Delete a file from flash memory:

delete disk*n***:** *filename*

disk0: is the internal memory. Other drive numbers represent external storage such as a USB drive, SSD, or SD card.

By default, the file is deleted from the current working directory if you do not specify a path. You may use wildcards when deleting files. You are prompted with the filename to delete, and then you must confirm the deletion.

Erase the Flash File System

To erase the flash file system, perform the following steps.

Procedure

Step 1 Connect to the ASA console port according to the instructions inAccess the ISA 3000 Console, on page 13.

Step 2 Power off the ASA, then power it on.

Step 3 During startup, press the **Escape** key when you are prompted to enter ROMMON mode.

Step 4 Enter the **erase** command, which overwrites all files and erases the file system, including hidden system files:

rommon #1> **erase** [**disk0:** | **disk***n***:** | **usb:**]

disk0: is the internal memory. Other drive numbers represent external storage. Newer models use **usb:** for the external USB drive.

Configure File Access

The ASA can use an FTP client, secure copy client, or TFTP client. You can also configure the ASA as a secure copy server so you can use a secure copy client on your computer.

Configure the FTP Client Mode

The ASA can use FTP to upload or download image files or configuration files to or from an FTP server. In passive FTP, the client initiates both the control connection and the data connection. The server, which is the recipient of the data connection in passive mode, responds with the port number to which it is listening for the specific connection.

Procedure

Set the FTP mode to passive:

ftp mode passive

Example:

```
ciscoasa(config)# ftp mode passive
```

Configure the ASA Secure Copy Client

You can configure SCP settings when the ASA acts as an SCP client using the **copy** command.

The performance of SCP depends partly on the encryption cipher used. By default, the ASA negotiates one of the following algorithms in order: 3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr. If the first algorithm proposed (3des-cbc) is chosen, then the performance is much slower than a more efficient algorithm such as aes128-cbc. To change the proposed ciphers, use the **ssh cipher encryption** command; for example, **ssh cipher encryption custom aes128-cbc**

Before you begin

- The ASA license must have the strong encryption (3DES/AES) license to support SSH Version 2 connections.
- Unless otherwise specified, for multiple context mode, complete this procedure in the system execution space. To change from the context to the system execution space, enter the **changeto system** command.
- For the SCP server, enable SSH on the ASA according to Configure SSH Access, on page 1161.

Procedure

Step 1 (Optional) The ASA stores the SSH host key for each SCP server to which it connects. You can manually manage keys if desired.

ssh pubkey-chain [**no**] **server** *ip_address* {**key-string** *key_string* **exit**| **key-hash** {**md5** | **sha256**} *fingerprint*}

Example:

```
ciscoasa(config)# ssh pubkey-chain
ciscoasa(config-ssh-pubkey-chain)# server 10.7.8.9
ciscoasa(config-ssh-pubkey-server)# key-string
Enter the base 64 encoded RSA public key.
End with the word "exit" on a line by itself
ciscoasa(config-ssh-pubkey-server-string)# c1:b1:30:29:d7:b8:de:6c:97:77:10:d7:46:41:63:87
ciscoasa(config-ssh-pubkey-server-string)# exit
ciscoasa(config-ssh-pubkey-server)# show running-config ssh pubkey-chain
ssh pubkey-chain
  server 10.7.8.9
```

```
    key-hash sha256 f1:22:49:47:b6:76:74:b2:db:26:fb:13:65:d8:99:19:
e7:9e:24:46:59:be:13:7f:25:27:70:9b:0e:d2:86:12
```

For each server, you can specify the **key-string** (public key) or **key-hash** (hashed value) of the SSH host.

The *key_string* is the Base64 encoded RSA public key of the remote peer. You can obtain the public key value from an open SSH client; that is, from the .ssh/id_rsa.pub file. After you submit the Base64 encoded public key, that key is then hashed via SHA-256.

The **key-hash** {**md5** | **sha256**} *fingerprint* enters the already hashed key (using an MD5 or SHA-256 key); for example, a key that you copied from **show** command output.

Step 2 (Optional) Enable or disable SSH host key checking. For multiple context mode, enter this command in the admin context.

[**no**] **ssh stricthostkeycheck**

Example:

```
ciscoasa# ssh stricthostkeycheck
ciscoasa# copy x scp://cisco@10.86.95.9/x
The authenticity of host '10.86.95.9 (10.86.95.9)' can't be established.
RSA key fingerprint is dc:2e:b3:e4:e1:b7:21:eb:24:e9:37:81:cf:bb:c3:2a.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.86.95.9' (RSA) to the list of known hosts.
Source filename [x]?

Address or name of remote host [10.86.95.9]?

Destination username [cisco]?

Destination password []? cisco123

Destination filename [x]?
```

By default, this option is enabled. When this option is enabled, you are prompted to accept or reject the host key if it is not already stored on the ASA. When this option is disabled, the ASA accepts the host key automatically if it was not stored before.

Examples

The following example adds an already hashed host key for the server at 10.86.94.170:

```
ciscoasa(config)# ssh pubkey-chain
ciscoasa(config-ssh-pubkey-chain)# server 10.86.94.170
ciscoasa(config-ssh-pubkey-server)# key-hash sha256 65:d9:9d:fe:1a:bc:61:aa:
64:9d:fc:ee:99:87:38:df:a8:8e:d9:e9:ff:42:de:e8:8d:2d:bf:a9:2b:85:2e:19
```

The following example adds a host string key for the server at 10.7.8.9:

```
ciscoasa(config)# ssh pubkey-chain
ciscoasa(config-ssh-pubkey-chain)# server 10.7.8.9
ciscoasa(config-ssh-pubkey-server)# key-string
Enter the base 64 encoded RSA public key.
End with the word "exit" on a line by itself
ciscoasa(config-ssh-pubkey-server-string)# c1:b1:30:29:d7:b8:de:6c:97:77:10:d7:
```

```
46:41:63:87
ciscoasa(config-ssh-pubkey-server-string)# exit
```

Configure the ASA TFTP Client Path

TFTP is a simple client/server file transfer protocol, which is described in RFC 783 and RFC 1350 Rev. 2. You can configure the ASA as a TFTP client so that it can copy files to or from a TFTP server. In this way, you can back up and propagate configuration files to multiple ASAs.

This section lets you predefine the path to a TFTP server so you do not need to enter it in commands such as **copy** and **configure net**.

Procedure

Predefine the TFTP server address and filename for use with **configure net** and **copy** commands:

tftp-server *interface_name server_ip filename*

Example:

```
ciscoasa(config)# tftp-server inside 10.1.4.7 files/config1.cfg
ciscoasa(config)# copy tftp: test.cfg

Address or name of remote host [10.1.4.7]?

Source filename [files/config1.cfg]?config2.cfg

Destination filename [test.cfg]?

Accessing tftp://10.1.4.7/files/config2.cfg;int=outside...
```

You can override the filename when you enter the command; for example, when you use the **copy** command, you can take advantage of the predefined TFTP server address but still enter any filename at the interactive prompts.

For the **copy** command, enter **tftp:** to use the tftp-server value instead of **tftp://***url*.

Copy a File to the ASA

This section describes how to copy the application image, ASDM software, a configuration file, or any other file that needs to be downloaded to internal or external flash memory from a TFTP, FTP, SMB, HTTP, HTTPS, SCP server, or from a drive such as a USB drive.

Guidelines

- **disk0:** is the internal memory. Other drive numbers represent external storage such as a USB drive, SSD, or SD card.
- Make sure the USB drive is formatted as EXT2/3/4 or VFAT/FAT32.

- You cannot have two files with the same name but with different letter case in the same directory in flash memory. For example, if you attempt to download the file, Config.cfg, to a location that contains the file, config.cfg, you receive the following error message:

```
%Error opening disk0:/Config.cfg (File exists)
```

- For multiple context mode, you must be in the system execution space unless you configure private storage (**storage-url private**). Within a context, you can also copy the running or startup configuration (see Copy a File to the Startup or Running Configuration, on page 1228 if you have private storage and Back Up a Context Configuration within a Context, on page 1243), or a capture. SCP is not supported within a context using the CiscoSSH stack.

Before you begin

- To use the ASA **copy** command to copy a file to or from an SCP server, you have to:
  - (CiscoSSH stack only) Enable SSH access on the ASA for the SCP server subnet/host using the **ssh** command.
  - Generate a key pair (for physical ASAs only) using the **crypto key generate** command.

Procedure

Copy a file using one of the following server types.

- Copy from a TFTP server:

  copy [**/noconfirm**] [*interface_name*] **tftp:**//*server*[/*path*]/*src_filename* **disk***n***:/**[*path*/]*dest_filename*

  Example:

```
ciscoasa# copy tftp://10.1.1.67/files/context1.cfg disk0:/context1.cfg

Address or name of remote host [10.1.1.67]?

Source filename [files/context1.cfg]?

Destination filename [context1.cfg]?
Cryptochecksum: db8ba196 9ad189a8 7f5f501f 1bec469b
!!!!!!!!!!!
11143 bytes copied in 5.710 secs (2228 bytes/sec)
```

- Copy from an FTP server:

  copy [**/noconfirm**] [*interface_name*] **ftp:**//[*user*[**:***password*]**@**]*server*[/*path*]/*src_filename* **disk***n***:/**[*path*/]*dest_filename*

  Example:

```
ciscoasa# copy ftp://jcrichton:aeryn@10.1.1.67/files/context1.cfg
disk0:/contexts/context1.cfg

Address or name of remote host [10.1.1.67]?
```

```
Source username [jcrichton]?

Source password [aeryn]?

Source filename [files/context1.cfg]?

Destination filename [contexts/context1.cfg]?
Cryptochecksum: db8ba196 9ad189a8 7f5f501f 1bec469b
!!!!!!!!!!!
11143 bytes copied in 5.710 secs (2228 bytes/sec)
```

Note
The password should not contain the '@' special character. The copy operation fails if the password contains this character.

- Copy from an HTTP(S) server:

  copy [**/noconfirm**] [*interface_name*] **http**[**s**]**://**[*user*[**:***password*]**@**]*server*[**:***port*][*/path*]*/src_filename* **disk***n***:/**[*path/*]*dest_filename*

  Example:

```
ciscoasa# copy https://asun:john@10.1.1.67/files/moya.cfg disk0:/contexts/moya.cfg

Address or name of remote host [10.1.1.67]?

Source username [asun]?

Source password [john]?

Source filename [files/moya.cfg]?

Destination filename [contexts/moya.cfg]?
Cryptochecksum: db8ba196 9ad189a8 7f5f501f 1bec469b
!!!!!!!!!!!
11143 bytes copied in 5.710 secs (2228 bytes/sec)
```

- Copy from an SMB server:

  copy [**/noconfirm**] [*interface_name*] **smb://**[*user*[**:***password*]**@**]*server*[*/path*]*/src_filename* **disk***n***:/**[*path/*]*dest_filename*

  Example:

```
ciscoasa# copy /noconfirm smb://chiana:dargo@10.1.1.67/test.xml disk0:/test.xml

Cryptochecksum: db8ba196 9ad189a8 7f5f501f 1bec469b
!!!!!!!!!!!
11143 bytes copied in 5.710 secs (2228 bytes/sec)
```

- Copy from a SCP server:

  The **;int=***interface* option bypasses the route lookup and always uses the specified interface to reach the SCP server.

  copy [**/noconfirm**] [*interface_name*] **scp://**[*user*[**:***password*]**@**]*server*[*/path*]*/src_filename*[**;int=***interface_name*] **disk***n***:/**[*path/*]*dest_filename*

Example:

```
ciscoasa# copy scp://pilot@10.86.94.170/test.cfg disk0:/test.cfg

Address or name of remote host [10.86.94.170]?

Source username [pilot]?

Destination filename [test.cfg]?

The authenticity of host '10.86.94.170 (10.86.94.170)' can't be established.
RSA key fingerprint is
<65:d9:9d:fe:1a:bc:61:aa:64:9d:fc:ee:99:87:38:df:a8:8e:d9:e9:ff:42:de:e8:8d:2d:bf:a9:2b:85:2e:19>(SHA256).
Are you sure you want to continue connecting (yes/no)? yes

Please use the following commands to add the hash key to the configuration:
  ssh pubkey-chain
    server 10.86.94.170
      key-hash sha256
65:d9:9d:fe:1a:bc:61:aa:64:9d:fc:ee:99:87:38:df:a8:8e:d9:e9:ff:42:de:e8:8d:2d:bf:a9:2b:85:2e:19


Password: <type in password>
!!!!!!
6006 bytes copied in 8.160 secs (750 bytes/sec)
```

- Copy from USB or an internal or external drive:

  copy [**/noconfirm**] **disk***n***:/**[*path*]*/src_filename* **disk***n***:/**[*path/*]*dest_filename*

  Example:

```
ciscoasa# copy /noconfirm disk1:/test.xml disk0:/test.xml

Cryptochecksum: db8ba196 9ad189a8 7f5f501f 1bec469b
!!!!!!!!!!!
11143 bytes copied in 5.710 secs (2228 bytes/sec)
```

Copy a File to the Startup or Running Configuration

You can download a text file to the running or startup configuration from a TFTP, FTP, SMB, HTTP(S), or SCP server, or from the flash memory.

When you copy a configuration to the running configuration, you merge the two configurations. A merge adds any new commands from the new configuration to the running configuration. If the configurations are the same, no changes occur. If commands conflict or if commands affect the running of the ASA, then the effect of the merge depends on the command. You might get errors, or you might have unexpected results.

(Optional) Specify the interface through which the ASA communicates with the server. If you do not specify the interface, the ASA checks the management-only routing table; if there are no matches, it then checks the data routing table.

Within a context in multiple context mode, you can use this procedure if you enable private storage (**storage-url private**). Without private storage, to download from a TFTP server to the running context configuration, use the **configure net** command. SCP is not supported within a context using the CiscoSSH stack

Procedure

To copy a file to the startup configuration or running configuration, enter one of the following commands for the appropriate download server:

- Copy from a TFTP server:

  copy [**/noconfirm**] [*interface_name*] **tftp://***server*[*/path*]*/src_filename* {**startup-config** | **running-config**}

  Example:

  ```
  ciscoasa# copy tftp://10.1.1.67/files/old-running.cfg running-config
  ```

- Copy from an FTP server:

  copy [**/noconfirm**] [*interface_name*] **ftp://**[*user*[**:***password*]**@**]*server*[*/path*]*/src_filename* {**startup-config** | **running-config**}

  Example:

  ```
  ciscoasa# copy ftp://jcrichton:aeryn@10.1.1.67/files/old-startup.cfg startup-config
  ```

- Copy from an HTTP(S) server:

  copy [**/noconfirm**] [*interface_name*] **http**[**s**]**://**[*user*[**:***password*]**@**]*server*[**:***port*][*/path*]*/src_filename* {**startup-config** | **running-config**}

  Example:

  ```
  ciscoasa# copy https://asun:john@10.1.1.67/files/new-running.cfg running-config
  ```

- Copy from an SMB server:

  copy [**/noconfirm**] [*interface_name*] **smb://**[*user*[**:***password*]**@**]*server*[*/path*]*/src_filename* {**startup-config** | **running-config**}

  Example:

  ```
  ciscoasa# copy /noconfirm smb://chiana:dargo@10.1.1.67/new-running.cfg running-config
  ```

- Copy from a SCP server:

  copy [**/noconfirm**] [*interface_name*] **scp://**[*user*[**:***password*]**@**]*server*[*/path*]*/src_filename*[**;int=***interface_name*] {**startup-config** | **running-config**}

  Example:

```
ciscoasa# copy scp://pilot:moya@10.86.94.170/new-startup.cfg startup-config
```

The **;int=***interface* option bypasses the route lookup and always uses the specified interface to reach the SCP server.

Examples

For example, to copy the configuration from a TFTP server, enter the following command:

```
ciscoasa# copy tftp://209.165.200.226/configs/startup.cfg startup-config
```

To copy the configuration from an FTP server, enter the following command:

```
ciscoasa# copy ftp://admin:letmein@209.165.200.227/configs/startup.cfg startup-config
```

To copy the configuration from an HTTP server, enter the following command:

```
ciscoasa# copy http://209.165.200.228/configs/startup.cfg startup-config
```

Set the ASA Image, ASDM, and Startup Configuration

If you have more than one ASA or ASDM image, you should specify the image that you want to boot. If you do not set the image, the default boot image is used, and that image may not be the one intended. For the startup configuration, you can optionally specify a file in the visible file system instead of a hidden directory.

See the following model guidelines:

- Firepower 4100/9300 chassis—ASA upgrades are managed by FXOS; you cannot upgrade the ASA within the ASA operating system, so do not use this procedure for the ASA image. You can upgrade the ASA and FXOS separately from each other, and they are listed separately in the FXOS directory listing. The ASA package always includes ASDM.
- Firepower 2100 in Platform mode—The ASA, ASDM, and FXOS images are bundled together into a single package. Package updates are managed by FXOS; you cannot upgrade the ASA within the ASA operating system, so do not use this procedure for the ASA image. You *cannot* upgrade the ASA and FXOS separately from each other; they are always bundled together.
- Firepower 1000, 2100 in Appliance mode, Secure Firewall 3100—The ASA, ASDM, and FXOS images are bundled together into a single package. Package updates are managed by the ASA using this procedure. Although these platforms use the ASA to identify the image to boot, the underlying mechanism is different from legacy ASAs. See the command description below for more information.
- ASDM for the models—ASDM can be upgraded from within the ASA operating system, so you do not need to only use the bundled ASDM image. For the Firepower 2100 in Platform mode and Firepower 4100/9300, ASDM images that you upload manually do not appear in the FXOS image list; you must manage ASDM images from the ASA.

Note When you upgrade the ASA bundle, the ASDM image in the bundle replaces the previous ASDM bundle image on the ASA because they have the same name (**asdm.bin**). But if you manually chose a different ASDM image that you uploaded (for example, **asdm-782.bin**), then you continue to use that image even after a bundle upgrade. To make sure that you are running a compatible version of ASDM, you should either upgrade ASDM before you upgrade the bundle, or you should reconfigure the ASA to use the bundled ASDM image (**asdm.bin**) just before upgrading the ASA bundle.

- ASAv—The initial deployment ASAv package puts the ASA image in the read-only boot:/ partition. When you upgrade the ASAv, you specify a different image in flash memory. Note that if you later clear your configuration (**clear configure all**), then the ASAv will revert to loading the original deployment image. The initial deployment ASAv package also includes an ASDM image that it places in flash memory. You can upgrade the ASDM image separately.
- **disk0:** is the internal memory. Other drive numbers represent external storage such as a USB drive, SSD, or SD card.

See the following default settings:

- ASA image:
  - Firepower 1000, 2100 in Appliance mode, Secure Firewall 3100—Boots the previously-running boot image.
  - ISA 3000—Boots the first application image that it finds in internal flash memory.
  - ASAv—Boots the image in the read-only boot:/ partition that was created when you first deployed.
  - Firepower 4100/9300 chassis—The FXOS system determines which ASA image to boot. You cannot use this procedure to set the ASA image.
  - Firepower 2100 in Platform mode—The FXOS system determines which ASA/FXOS package to boot. You cannot use this procedure to set the ASA image.
- ASDM image on all ASAs—Boots the first ASDM image that it finds in internal flash memory, or if one does not exist in this location, then in external flash memory.
- Startup configuration—By default, the ASA boots from a startup configuration that is a hidden file.

Procedure

Step 1 Set the ASA boot image location:

boot system *url*

Example:

```
ciscoasa(config)# boot system disk0:/images/asa921.bin
```

The URL can be:

- **disk** *n***:/**[*path*/]*filename*
- **tftp:**//[*user*[**:***password*]**@**]*server*[**:***port*]/[*path*/]*filename*

  The TFTP option is not supported on all models.

Firepower 1000, 2100 in Appliance mode, Secure Firewall 3100: You can only enter a single **boot system** command. If you upgrade to a new image, then you must enter **no boot system** to remove the previous image you set. Note that you may not have a **boot system** command present in your configuration; for example, if you installed the image from ROMMON, have a new device, or you removed the command manually. The **boot system** command performs an action when you enter it: the system validates and unpacks the image and copies it to the boot location (an internal location on disk0 managed by FXOS). The new image will load when you reload the ASA. If you change your mind prior to reloading, you can enter the **no boot system** command to delete the new image from the boot location, so the current image continues to run. You can even delete the original image file from the ASA flash memory after you enter this command, and the ASA will boot correctly from the boot location; however, we recommend keeping any images you want to use in flash memory because the **boot system** command only works with images in flash memory. Unlike other models, this command in the startup configuration does not affect the booting image, and is essentially cosmetic. The last-loaded boot image will always run upon reload. If you do not save the configuration after you enter this command, then when you reload, the old command will be present in your configuration, even though the new image was booted. Be sure to save the configuration so that the configuration remains in sync. You can only load images with the original filename from the Cisco download site. If you change the filename, it will not load. You can also reimage to the Firewall Threat Defense by loading the Firewall Threat Defense image. In this case, you are prompted to reload immediately.

Other models: You can enter up to four **boot system** command entries to specify different images to boot from in order; the ASA boots the first image it finds successfully. When you enter the **boot system** command, it adds an entry at the bottom of the list. To reorder the boot entries, you must remove all entries using the the **clear configure boot system** command, and re-enter them in the order you desire. Only one **boot system tftp** command can be configured, and it must be the first one configured.

Note
If the ASA is stuck in a cycle of constant booting, you can reboot the ASA into ROMMON mode. For more information about the ROMMON mode, see View Debugging Messages, on page 1271.

Example:

```
asa(config)# boot system disk0:/cisco-asa-fp2k.9.13.2.SPA
The system is currently installed with security software package 9.13.1, which has:
   - The platform version:  2.7.1
   - The CSP (asa) version: 9.13.1
Preparing new image for install...
!!!!!!!!!!!!!
Image download complete (Successful unpack the image).
Installation of version 9.13.2 will do the following:
   - upgrade to the new platform version 2.7.2
   - upgrade to the CSP ASA version 9.13.2
After the installation is complete, reload to apply the new image.
Finalizing image install process...

Install_status: ready...........
Install_status: validating-images.....
Install_status: update-software-pack-completed
asa(config)#
```

Step 2 Set the ASDM image to boot:

asdm image disk*n***:/**[*path*/]*filename*

Example:

```
ciscoasa(config)# asdm image disk0:/images/asdm721.bin
```

If you do not specify the image to boot, even if you have only one image installed, then the ASA inserts the **asdm image** command into the running configuration. To avoid problems with Auto Update (if configured), and to avoid the image search at each startup, you should specify the ASDM image that you want to boot in the startup configuration.

Step 3 (Optional) Set the startup configuration to be a known file instead of the default hidden file:

boot config disk*n***:/**[*path*/]*filename*

This feature is important for when you work with large configurations that do not fit in the hidden directory. If you save a large configuration and see the following error message, be sure to instead save the configuration to a new file using this command:

```
%Error writing. nvram:/startup-config (No space left on device:)
```

Example:

```
ciscoasa(config)# boot config disk0:/configs/startup1.cfg
```

Back Up and Restore Configurations or Other Files

We recommend that you make regular backups of your configuration and other system files to guard against system failure.

Perform a Complete System Backup or Restoration

These procedures describe how to back up and restore configurations and images to a tar.gz file and transfer it to your local computer.

Before You Begin Backup or Restore

- You should have at least 300 MB of disk space available at the backup or restore location before you start a backup or restore.
- You can start only one backup or restore at a time.
- If you make any configuration changes during or after a backup, those changes will not be included in the backup. If you change a configuration after making the backup, then perform a restore, this configuration change will be overwritten. As a result, the ASA might behave differently.
- Do not overwrite or restore the system configuration (*cspCfg.xml*) file from one ASA to another, especially between different ASA platform models. The *cspCfg.xml* file may contain device-specific hardware information. Copying this file from a different ASA can create a hardware mismatch, such as differences

in CPU core count. This mismatch may cause the target ASA to fail to boot properly and eventually enter into Failsafe mode.

- You can only restore a configuration to the same ASA version as when you performed the original backup. You cannot use the restore tool to migrate a configuration from one ASA version to another. If a configuration migration is required, the ASA automatically upgrades the resident startup configuration when it loads the new ASA OS.
- If you use clustering, you can only back up or restore the startup-configuration, running-configuration, and identity certificates. You must create and restore a backup separately for each unit.
- If you use failover, you must create and restore a backup separately for the active and standby units.
- If you set a master passphrase for the ASA, then you need that master passphrase to restore the backup configuration that you create with this procedure. If you do not know the master passphrase for the ASA, see Configure the Master Passphrase, on page 744 to learn how to reset it before continuing with the backup.
- If you import PKCS12 data (with the **crypto ca trustpoint** command) and the trustpoint uses RSA keys, the imported key pair is assigned the same name as the trustpoint. Because of this limitation, if you specify a different name for the trustpoint and its key pair after you have restored an ASDM configuration, the startup configuration will be the same as the original configuration, but the running configuration will include a different key pair name. This means that if you use different names for the key pair and trustpoint, you cannot restore the original configuration. To work around this issue, make sure that you use the same name for the trustpoint and its key pair.
- You cannot back up using the CLI and restore using ASDM, or vice versa.
- Each backup file includes the following content:
  - Running-configuration
  - Startup-configuration
  - All security images

    Cisco Secure Desktop and Host Scan images

    Cisco Secure Desktop and Host Scan settings

    AnyConnect Client (SVC) images and profiles

    AnyConnect Client (SVC) customizations and transforms
  - Identity certificates (includes RSA key pairs tied to identity certificates; excludes standalone keys)
  - VPN pre-shared keys
  - SSL VPN configurations
  - Application Profile Custom Framework (APCF)
  - Bookmarks
  - Customizations
  - Dynamic Access Policy (DAP)
  - Plug-ins
  - Pre-fill scripts for connection profiles

- Proxy Auto-config
- Translation table
- Web content
- Version information

Back Up the System

This procedure describes how to perform a complete system backup.

Procedure

Step 1 Back up the system:

backup [**/noconfirm**] [**context** *ctx-name*] [**interface** *name*] [**passphrase** *value*] [**location** *path*]

Example:

```
ciscoasa# backup location disk0:/sample-backup]
Backup location [disk0:/sample-backup]?
```

If you do not specify the **interface** *name*, the ASA checks the management-only routing table; if there are no matches, it then checks the data routing table.

In multiple context mode from the system execution space, enter the **context** keyword to backup the specified context. Each context must be backed up individually; that is, re-enter the **backup** command for each file.

During the backup of VPN certificates and preshared keys, a secret key identified by the **passphrase** keyword is required to encode the certificates. You must provide a passphrase to be used for encoding and decoding the certificates in PKCS12 format. The backup only includes RSA key pairs tied to the certificates and excludes any standalone certificates.

The backup **location** can be a local disk or a remote URL. If you do not provide a location, the following default names are used:

- Single mode—disk0:*hostname*.backup.*timestamp*.tar.gz
- Multiple mode—disk0:*hostname*.context-*ctx-name*.backup.*timestamp*.tar.gz

Step 2 Follow the prompts:

Example:

```
ciscoasa# backup location disk0:/sample-backup
Backup location [disk0:/sample-backup]?

Begin backup…
Backing up [ASA version] … Done!
Backing up [Running Config] … Done!
Backing up [Startup Config]  … Done!

Enter a passphrase to encrypt identity certificates. The default is cisco.
You will be required to enter the same passphrase while doing a restore: cisco
```

```
Backing up [Identity Certificates] … Done!

IMPORTANT: This device uses master passphrase encryption. If this backup file
is used to restore to a device with a different master passphrase,
you will need to provide the current master passphrase during restore.
Backing up [VPN Pre-shared keys] … Done!
Backing up [SSL VPN Configurations: Application Profile Custom Framework] … Done!
Backing up [SSL VPN Configurations: Bookmarks]… Done!
Backing up [SSL VPN Configurations: Customization] … Done!
Backing up [SSL VPN Configurations: Dynamic Access Policy] … Done!
Backing up [SSL VPN Configurations: Plug-in] … Done!
Backing up [SSL VPN Configurations: Pre-fill scripts for Connection Profile] … Done!
Backing up [SSL VPN Configurations: Proxy auto-config] … Done!
Backing up [SSL VPN Configurations: Translation table] … Done!
Backing up [SSL VPN Configurations: Web Content] … Done!
Backing up [Anyconnect(SVC) client images and profiles] … Done!
Backing up [Anyconnect(SVC) customizations and transforms] … Done!
Backing up [Cisco Secure Desktop and Host Scan images] … Done!
Backing up [UC-IME tickets] … Done!
Compressing the backup directory ... Done!
Copying Backup ... Done!
Cleaning up ... Done!
Backup finished!
```

Restore the Backup

You can specify configurations and images to restore from a zip tar.gz file on your local computer.

Procedure

Step 1 Restore the system from the backup file.

restore [**/noconfirm**] [**context** *ctx-name*] [**passphrase** *value*] [**location** *path*]

Example:

```
ciscoasa# restore location disk0:/5525-2051.backup.2014-07-09-223$
restore location [disk0:/5525-2051.backup.2014-07-09-223251.tar.gz]?
```

When using the **context** keyword to restore multiple contexts, each backed up context file must be restored individually; that is, re-enter the **restore** command for each file.

Step 2 Follow the prompts:

Example:

```
ciscoasa# restore location disk0:/5525-2051.backup.2014-07-09-223$
restore location [disk0:/5525-2051.backup.2014-07-09-223251.tar.gz]?

Copying Backup file to local disk... Done!
Extracting the backup file ... Done!
Warning: The ASA version of the device is not the same as the backup version,
some configurations might not work after restore!
 Do you want to continue? [confirm] y
Begin restore ...
IMPORTANT: This backup configuration uses master passphrase encryption.
```

```
Master passphrase is required to restore running configuration,
startup configuration and VPN pre-shared keys.
Backing up [VPN Pre-shared keys] … Done!
Backing up [SSL VPN Configurations: Application Profile Custom Framework] … Done!
Backing up [SSL VPN Configurations: Bookmarks]… Done!
Backing up [SSL VPN Configurations: Customization] … Done!
Backing up [SSL VPN Configurations: Dynamic Access Policy] … Done!
Backing up [SSL VPN Configurations: Plug-in] … Done!
Backing up [SSL VPN Configurations: Pre-fill scripts for Connection Profile] … Done!
Backing up [SSL VPN Configurations: Proxy auto-config] … Done!
Backing up [SSL VPN Configurations: Translation table] … Done!
Backing up [SSL VPN Configurations: Web Content] … Done!
Backing up [Anyconnect(SVC) client images and profiles] … Done!
Backing up [Anyconnect(SVC) customizations and transforms] … Done!
Backing up [Cisco Secure Desktop and Host Scan images] … Done!
Backing up [UC-IME tickets] … Done!
Restoring [Running Configuration]
Following messages are as a result of applying the backup running-configuration to
this device, please note them for future reference.

ERROR: Interface description was set by failover and cannot be changed
ERROR: Unable to set this url, it has already been set
Remove the first instance before adding this one
INFO: No change to the stateful interface
Failed to update LU link information
.Range already exists.
WARNING: Advanced settings and commands should only be altered or used
under Cisco supervision.
ERROR: Failed to apply media termination address 198.0.1.228 to interface outside,
the IP is already used as media-termination address on interface outside.
ERROR: Failed to apply media termination address 198.0.0.223 to interface inside,
the IP is already used as media-termination address on interface inside.
WARNING: PAC settings will override http- and https-proxy configurations.
Do not overwrite configuration file if you want to preserve the old http-
and https-proxy configurations.

Cryptochecksum (changed): 98d23c2c ccb31dc3 e51acf88 19f04e28
 Done!
Restoring UC-IME ticket ... Done!
Enter the passphrase used while backup to encrypt identity certificates.
The default is cisco. If the passphrase is not correct, certificates will not be restored.

No passphrase was provided for identity certificates.
Using the default value: cisco. If the passphrase is not correct,
certificates will not be restored.
Restoring Certificates ...
Enter the PKCS12 data in base64 representation....
ERROR: A keypair named Main already exists.
INFO: Import PKCS12 operation completed successfully
. Done!
Cleaning up ... Done!
Restore finished!
```

Configure Automatic Backup and Restore (ISA 3000)

On the ISA 3000, you can configure automatic backups to a particular location every time you save your configuration using **write memory**.

Automatic restore lets you easily configure new devices with a complete configuration loaded on an SD flash memory card. Automatic restore is enabled in the default factory configuration.

Configure Automatic Backup (ISA 3000)

On the ISA 3000, you can configure automatic backups to a particular location every time you save your configuration using **write memory**.

Before you begin

This feature is only available on the ISA 3000.

Procedure

Step 1 Set the back-up package parameters:

backup-package backup [**interface** *name*] **location** {**disk***n***:** | *url*} [**passphrase** *string*]

- **interface** *name*—Specifies the interface to reach the backup URL, if you specify off-device storage. If you do not specify the interface name, the ASA checks the management-only routing table; if there are no matches, it then checks the data routing table.
- **location** {**disk***n***:** | *url*}—Specifies the storage medium to be used for backing up data. You can specify a URL or local storage. disk0 is the internal flash drive. disk1 is an optional USB memory stick on USB 1. disk2 is an optional USB memory stick on USB 2. And disk3 is the SD memory card. Note that the default settings for automatic restore use disk3.
- **passphrase** *string*—Sets the passphrase to secure the backed-up data. Note that the default settings for automatic restore use "cisco" as the passphrase.

These settings are also used by default with the manual **backup** command. See Back Up the System, on page 1235. Note that if you use the manual **backup** command when you have automatic backup or restore enabled, then the system saves a backup file with the specified name, as well as the "auto-backup-asa.tgz" name used by automatic backup and restore.

Example:

```
ciscoasa(config)# backup-package backup location disk3: passphrase cisco
```

Step 2 Enable automatic mode for back-up and restore:

backup-package backup auto

When you save the configuration using **write memory**, the configuration is automatically saved to the backup location as well as to the startup configuration. The backup file has the name "auto-backup-asa.tgz". To disable automatic backups, use the **no** form of the command.

Example:

```
ciscoasa(config)# backup-package backup auto
```

Configure Automatic Restore (ISA 3000)

Automatic restore mode restores the system configuration on a device without any user intervention. For example, you insert an SD memory card containing a saved backup configuration into a new device and then power the device on. When the device comes up, it checks the SD card to decide if the system configuration needs to be restored. (The restoration is only initiated if the backup file has the "fingerprint" of a different device. The fingerprint of the backup file is updated to match the current device during a backup or restore operation. So if the device has already completed a restore, or if it has created its own backup, then the automatic restore is skipped.) If the fingerprint shows a restoration is required, the device replaces the system configuration (startup-config, running-config, SSL VPN configuration, and so on; see Back Up the System, on page 1235 for details about the contents of the backup). When the device finishes booting, it is running the saved configuration.

Automatic restore is enabled in the default factory configuration, so you can easily configure new devices with a complete configuration loaded on an SD memory card without having to perform any pre-configuration of the device.

Because the device needs to decide early in the boot process if the system configuration needs to be restored, it checks ROMMON variables to determine if the device is in automatic restore mode and to obtain the location of the backup configuration. The following ROMMON variables are used:

- **RESTORE_MODE =** {**auto** | **manual**}

  The default is **auto**.

- **RESTORE_LOCATION =** {**disk0:** | **disk1:** | **disk2:** | **disk3:**}

  The default is **disk3:**.

- **RESTORE_PASSPHRASE =** *key*

  The default is **cisco**.

To change the automatic restore settings, complete the following procedure.

Before you begin

- This feature is only available on the ISA 3000.
- If you use the default restore settings, you need an SD memory card installed (part number SD-IE-1GB=).
- If you need to restore the default configuration to ensure that automatic restore is enabled, use the **configure factory default** command. This command is only available in transparent firewall mode, so if you are in routed firewall mode, use the **firewall transparent** command first.

Procedure

Step 1 Set the restore package parameters.

backup-package restore location {**disk***n***:** | *url*} [**passphrase** *string*]

- **location disk***n***:**—Specifies the storage medium to be used for restoring data. disk0 is the internal flash drive. disk1 is an optional USB memory stick on USB 1. disk2 is an optional USB memory stick on USB 2. And disk3 is the SD memory card. The default is disk3.

- **passphrase** *string*—Sets the passphrase to read the backed-up data. The default is "cisco".

These settings are also used by default with the manual **restore** command. See Back Up the System, on page 1235.

Example:

```
ciscoasa(config)# backup-package restore location disk1: passphrase $upe3rnatural
```

Step 2 Enable or disable automatic mode for restore.

[**no**] **backup-package restore auto**

The name of the file that is restored is "auto-backup-asa.tgz".

Example:

```
ciscoasa(config)# no backup-package restore auto
```

Back up the Single Mode Configuration or Multiple Mode System Configuration

In single context mode or from the system configuration in multiple mode, you can copy the startup configuration or running configuration to an external server or to the local flash memory.

Before you begin

(Optional) Specify the interface through which the ASA communicates with the server. If you do not specify the interface, the ASA checks the management-only routing table; if there are no matches, it then checks the data routing table.

Procedure

Back up the configuring using one of the following server types:

- Copy to a TFTP server:

  copy [**/noconfirm**] [*interface_name*] {**startup-config** | **running-config**} **tftp:**//*server*[/*path*]/*dst_filename*

  Example:

  ```
  ciscoasa# copy running-config tftp://10.1.1.67/files/new-running.cfg
  ```

- Copy to an FTP server:

  copy [**/noconfirm**] [*interface_name*] {**startup-config** | **running-config**} **ftp:**//[*user*[**:***password*]**@**]*server*[/*path*]/*dst_filename*

  Example:

```
ciscoasa# copy startup-config ftp://jcrichton:aeryn@10.1.1.67/files/new-startup.cfg
```

- Copy to an SMB server:

  copy [**/noconfirm**] [*interface_name*] {**startup-config** | **running-config**} **smb://**[*user*[**:***password*]**@**]*server*[*/path*]*/dst_filename*

  Example:

```
ciscoasa# copy /noconfirm running-config smb://chiana:dargo@10.1.1.67/new-running.cfg
```

- Copy to a SCP server:

  copy [**/noconfirm**] [*interface_name*] {**startup-config** | **running-config**} **scp://**[*user*[**:***password*]**@**]*server*[*/path*]*/dst_filename*[**;int=***interface_name*]

  Example:

```
ciscoasa# copy startup-config
scp://pilot:moya@10.86.94.170/new-startup.cfg
```

  The **;int=***interface* option bypasses the route lookup and always uses the specified interface to reach the SCP server.

- Copy to the local flash memory:

  copy [**/noconfirm**] {**startup-config** | **running-config**} {**disk0**|**disk1**}**:/**[*path*/]*dst_filename*

  Example:

```
ciscoasa# copy /noconfirm running-config disk0:/new-running.cfg
```

  Be sure that the destination directory exists. If it does not exist, first create the directory using the **mkdir** command.

Back Up a Context Configuration or Other File in Flash Memory

Copy context configurations or other files that are on the local flash memory by entering one of the following commands in the system execution space.

Before you begin

(Optional) Specify the interface through which the ASA communicates with the server. If you do not specify the interface, the ASA checks the management-only routing table; if there are no matches, it then checks the data routing table.

Procedure

Back up a context configuration using one of the following server types:

- Copy from flash to a TFTP server:

  copy [**/noconfirm**] [*interface_name*] {**disk0**|**disk1**}**:/**[*path*/]*src_filename* **tftp://***server*[*/path*]*/dst_filename*

  Example:

  ```
  ciscoasa# copy disk0:/asa-os.bin tftp://10.1.1.67/files/asa-os.bin
  ```

- Copy from flash to an FTP server:

  copy [**/noconfirm**] [*interface_name*] {**disk0**|**disk1**}**:/**[*path*/]*src_filename*
  ftp://[*user*[**:***password*]**@**]*server*[*/path*]*/dst_filename*

  Example:

  ```
  ciscoasa# copy disk0:/asa-os.bin ftp://jcrichton:aeryn@10.1.1.67/files/asa-os.bin
  ```

- Copy from flash to an SMB server:

  copy [**/noconfirm**] [*interface_name*] {**disk0**|**disk1**}**:/**[*path*/]*src_filename*
  smb://[*user*[**:***password*]**@**]*server*[*/path*]*/dst_filename*

  Example:

  ```
  ciscoasa# copy /noconfirm copy disk0:/asdm.bin
  smb://chiana:dargo@10.1.1.67/asdm.bin
  ```

- Copy from flash to SCP server:

  copy [**/noconfirm**] [*interface_name*] {**disk0**|**disk1**}**:/**[*path*/]*src_filename*
  scp://[*user*[**:***password*]**@**]*server*[*/path*]*/dst_filename*[**;int=***interface_name*]

  Example:

  ```
  ciscoasa# copy disk0:/context1.cfg
  scp://pilot:moya@10.86.94.170/context1.cfg
  ```

  The **;int=***interface* option bypasses the route lookup and always uses the specified interface to reach the SCP server.

- Copy from flash to the local flash memory:

  copy [**/noconfirm**] {**disk0**|**disk1**}**:/**[*path*/]*src_filename* {**disk0**|**disk1**}**:/**[*path*/]*dst_filename*

  Example:

  ```
  ciscoasa# copy /noconfirm disk1:/file1.cfg disk0:/file1.cfgnew-running.cfg
  ```

Be sure that the destination directory exists. If it does not exist, first create the directory using the **mkdir** command.

Back Up a Context Configuration within a Context

In multiple context mode, from within a context, you can perform the following backups.

Procedure

Step 1 Copy the running configuration to the startup configuration location (if on a remote server, it is connected to the admin context). The startip configuration location is specified by the **config-url** command.

copy running-config startup-config

Example:

```
ciscoasa/contexta# copy running-config startup-config
```

Step 2 Copy the running configuration to an FTP or TFTP server connected to the *context* network.

copy running-config {**ftp** | **tftp**}**:**//*url*/*dest_filename*

See the **copy** command help for exact URL options.

If you enable private storage (**storage-url private**), other server options are available. However, SCP is not supported within a context using the CiscoSSH stack.

Example:

```
ciscoasa/contexta# copy running-config tftp:/10.89.6.8/configs/1010-1.cfg
```

Copy the Configuration from the Terminal Display

Procedure

Step 1 Print the configuration to the terminal:

more system:running-config

Step 2 Copy the output from this command, and then paste the configuration into a text file.

Back Up Additional Files Using the Export and Import Commands

Additional files essential to your configuration might include the following:

- Files that you import using the **import webvpn** command. Currently, these files include customizations, URL lists, web content, plug-ins, and language translations.
- DAP policies (dap.xml).
- CSD configurations (data.xml).
- Digital keys and certificates.
- Local CA user database and certificate status files.

The CLI lets you back up and restore individual elements of your configuration using the **export** and **import** commands.

To back up these files, for example, those files that you imported with the **import webvpn** command or certificates, perform the following steps.

Procedure

Step 1 Run the applicable **show** command(s) as follows:

```
ciscoasa # show import webvpn plug-in
ica
rdp
ssh, telnet
vnc
```

Step 2 Run the **export** command for the file that you want to back up (in this example, the rdp file):

```
ciscoasa # export webvpn plug-in protocol rdp tftp://tftpserver/backupfilename
```

Use a Script to Back Up and Restore Files

You can use a script to back up and restore the configuration files on your ASA, including all extensions that you import via the **import webvpn** CLI, the CSD configuration XML files, and the DAP configuration XML file. For security reasons, we do not recommend that you perform automated backups of digital keys and certificates or the local CA key.

This section provides instructions for doing so and includes a sample script that you can use as is or modify as your environment requires. The sample script is specific to a Linux system. To use it for a Microsoft Windows system, you need to modify it using the logic of the sample.

Note You can alternatively use the **backup** and **restore** commands. See Perform a Complete System Backup or Restoration, on page 1233 for more information.

Before You Begin Using Backup and Restore Scripts

To use a script to back up and restore an ASA configuration, first perform the following tasks:

- Install Perl with an Expect module.
- Install an SSH client that can reach the ASA.
- Install a TFTP server to send files from the ASA to the backup site.

Another option is to use a commercially available tool. You can put the logic of this script into such a tool.

Run the Script

To run a backup-and-restore script, perform the following steps.

Procedure

Step 1 Download or cut-and-paste the script file to any location on your system.

Step 2 At the command line, enter **Perl***scriptname*, where *scriptname* is the name of the script file.

Step 3 Press **Enter**.

Step 4 The system prompts you for values for each option. Alternatively, you can enter values for the options when you enter the **Perl***scriptname* command before you press **Enter**. Either way, the script requires that you enter a value for each option.

Step 5 The script starts running, printing out the commands that it issues, which provides you with a record of the CLIs. You can use these CLIs for a later restore, which is particularly useful if you want to restore only one or two files.

Sample Script

```
#!/usr/bin/perl
#Description: The objective of this script is to show how to back up
configurations/extensions.
# It currently backs up the running configuration, all extensions imported via "import
webvpn" command, the CSD configuration XML file, and the DAP configuration XML file.
#Requirements: Perl with Expect, SSH to the ASA, and a TFTP server.
#Usage: backupasa -option option_value
#       -h: ASA hostname or IP address
#       -u: User name to log in via SSH
#       -w: Password to log in via SSH
#       -e: The Enable password on the security appliance
#       -p: Global configuration mode prompt
#       -s: Host name or IP address of the TFTP server to store the configurations
#       -r: Restore with an argument that specifies the file name. This file is produced
during backup.
```

```
#If you don't enter an option, the script will prompt for it prior to backup.
#
#Make sure that you can SSH to the ASA.

use Expect;
use Getopt::Std;

#global variables
%options=();
$restore = 0; #does backup by default
$restore_file = '';
$asa = '';
$storage = '';
$user = '';
$password = '';
$enable = '';
$prompt = '';
$date = `date +%F`;
chop($date);
my $exp = new Expect();

getopts("h:u:p:w:e:s:r:",\%options);
do process_options();

do login($exp);
do enable($exp);
if ($restore) {
   do restore($exp,$restore_file);
}
else {
   $restore_file = "$prompt-restore-$date.cli";
   open(OUT,">$restore_file") or die "Can't open $restore_file\n";
   do running_config($exp);
   do lang_trans($exp);
   do customization($exp);
   do plugin($exp);
   do url_list($exp);
   do webcontent($exp);
   do dap($exp);
   do csd($exp);
   close(OUT);
}
do finish($exp);

sub enable {
   $obj = shift;
   $obj->send("enable\n");
   unless ($obj->expect(15, 'Password:')) {
      print "timed out waiting for Password:\n";
   }
   $obj->send("$enable\n");
   unless ($obj->expect(15, "$prompt#")) {
      print "timed out waiting for $prompt#\n";
   }
}

sub lang_trans {
   $obj = shift;
   $obj->clear_accum();
   $obj->send("show import webvpn translation-table\n");
   $obj->expect(15, "$prompt#" );
   $output = $obj->before();
   @items = split(/\n+/, $output);
```

```
   for (@items) {
     s/^\s+//;
     s/\s+$//;
     next if /show import/ or /Translation Tables/;
     next unless (/^.+\s+.+$/);
     ($lang, $transtable) = split(/\s+/,$_);
     $cli = "export webvpn translation-table $transtable language $lang
$storage/$prompt-$date-$transtable-$lang.po";
     $ocli = $cli;
     $ocli =~ s/^export/import/;
     print "$cli\n";
     print OUT "$ocli\n";
     $obj->send("$cli\n");
     $obj->expect(15, "$prompt#" );
   }
}

sub running_config {
  $obj = shift;
  $obj->clear_accum();
  $cli ="copy /noconfirm running-config $storage/$prompt-$date.cfg";
  print "$cli\n";
  $obj->send("$cli\n");
  $obj->expect(15, "$prompt#" );
}


sub customization {
  $obj = shift;
  $obj->clear_accum();
  $obj->send("show import webvpn customization\n");
  $obj->expect(15, "$prompt#" );
  $output = $obj->before();
  @items = split(/\n+/, $output);

  for (@items) {
    chop;
    next if /^Template/ or /show import/ or /^\s*$/;
    $cli = "export webvpn customization $_ $storage/$prompt-$date-cust-$_.xml";
    $ocli = $cli;
    $ocli =~ s/^export/import/;
    print "$cli\n";
    print OUT "$ocli\n";
    $obj->send("$cli\n");
    $obj->expect(15, "$prompt#" );
  }
}

sub plugin {
   $obj = shift;
   $obj->clear_accum();
   $obj->send("show import webvpn plug-in\n");
   $obj->expect(15, "$prompt#" );
   $output = $obj->before();
   @items = split(/\n+/, $output);

   for (@items) {
     chop;
     next if /^Template/ or /show import/ or /^\s*$/;
     $cli = "export webvpn plug-in protocol $_ $storage/$prompt-$date-plugin-$_.jar";
     $ocli = $cli;
     $ocli =~ s/^export/import/;
     print "$cli\n";
     print OUT "$ocli\n";
```

```
      $obj->send("$cli\n");
      $obj->expect(15, "$prompt#" );
    }
}

sub url_list {
    $obj = shift;
    $obj->clear_accum();
    $obj->send("show import webvpn url-list\n");
    $obj->expect(15, "$prompt#" );
    $output = $obj->before();
    @items = split(/\n+/, $output);

    for (@items) {
      chop;
      next if /^Template/ or /show import/ or /^\s*$/ or /No bookmarks/;
      $cli="export webvpn url-list $_ $storage/$prompt-$date-urllist-$_.xml";
      $ocli = $cli;
      $ocli =~ s/^export/import/;
      print "$cli\n";
      print OUT "$ocli\n";
      $obj->send("$cli\n");
      $obj->expect(15, "$prompt#" );
    }
}

sub dap {
    $obj = shift;
    $obj->clear_accum();
    $obj->send("dir dap.xml\n");
    $obj->expect(15, "$prompt#" );

    $output = $obj->before();
    return 0 if($output =~ /Error/);

    $cli="copy /noconfirm dap.xml $storage/$prompt-$date-dap.xml";
    $ocli="copy /noconfirm $storage/$prompt-$date-dap.xml disk0:/dap.xml";
    print "$cli\n";
    print OUT "$ocli\n";
    $obj->send("$cli\n");
    $obj->expect(15, "$prompt#" );
}

sub csd {
    $obj = shift;
    $obj->clear_accum();
    $obj->send("dir sdesktop\n");
    $obj->expect(15, "$prompt#" );

    $output = $obj->before();
    return 0 if($output =~ /Error/);

    $cli="copy /noconfirm sdesktop/data.xml $storage/$prompt-$date-data.xml";
    $ocli="copy /noconfirm $storage/$prompt-$date-data.xml disk0:/sdesktop/data.xml";
    print "$cli\n";
    print OUT "$ocli\n";
    $obj->send("$cli\n");
    $obj->expect(15, "$prompt#" );
}

sub webcontent {
    $obj = shift;
    $obj->clear_accum();
    $obj->send("show import webvpn webcontent\n");
```

```
   $obj->expect(15, "$prompt#" );
   $output = $obj->before();
   @items = split(/\n+/, $output);

   for (@items) {
     s/^\s+//;
     s/\s+$//;
     next if /show import/ or /No custom/;
     next unless (/^.+\s+.+$/);
     ($url, $type) = split(/\s+/,$_);
     $turl = $url;
     $turl =~ s/\/\+//;
     $turl =~ s/\+\//-/;
     $cli = "export webvpn webcontent $url $storage/$prompt-$date-$turl";
     $ocli = $cli;
     $ocli =~ s/^export/import/;
     print "$cli\n";
     print OUT "$ocli\n";
     $obj->send("$cli\n");
     $obj->expect(15, "$prompt#" );
   }
}

sub login {
    $obj = shift;
    $obj->raw_pty(1);
    $obj->log_stdout(0); #turn off console logging.
    $obj->spawn("/usr/bin/ssh $user\@$asa") or die "can't spawn ssh\n";
    unless ($obj->expect(15, "password:" )) {
        die "timeout waiting for password:\n";
    }

    $obj->send("$password\n");

    unless ($obj->expect(15, "$prompt>" )) {
        die "timeout waiting for $prompt>\n";
    }
}

sub finish {
    $obj = shift;
    $obj->hard_close();
    print "\n\n";

}

sub restore {
   $obj = shift;
   my $file = shift;
   my $output;
   open(IN,"$file") or die "can't open $file\n";
   while (<IN>) {
      $obj->send("$_");
      $obj->expect(15, "$prompt#" );
      $output = $obj->before();
      print "$output\n";
   }
   close(IN);
}

sub process_options {
  if (defined($options{s})) {
      $tstr= $options{s};
      $storage = "tftp://$tstr";
```

```
    }
    else {
        print "Enter TFTP host name or IP address:";
        chop($tstr=<>);
        $storage = "tftp://$tstr";
    }
    if (defined($options{h})) {
        $asa = $options{h};
    }
    else {
        print "Enter ASA host name or IP address:";
        chop($asa=<>);
    }

    if (defined ($options{u})) {
        $user= $options{u};
    }
    else {
        print "Enter user name:";
        chop($user=<>);
    }

    if (defined ($options{w})) {
        $password= $options{w};
    }
    else {
        print "Enter password:";
        chop($password=<>);
    }
    if (defined ($options{p})) {
        $prompt= $options{p};
    }
    else {
        print "Enter ASA prompt:";
        chop($prompt=<>);
    }
    if (defined ($options{e})) {
        $enable = $options{e};
    }
    else {
        print "Enter enable password:";
        chop($enable=<>);
    }

    if (defined ($options{r})) {
       $restore = 1;
       $restore_file = $options{r};
    }
}
```

Hot Swap an SSD on the Secure Firewall 3100

If you have two SSDs, they form a RAID when you boot up. You can perform the following tasks at the CLI while the firewall is powered up:

- Hot swap one of the SSDs—If an SSD is faulty, you can replace it. Note that if you only have one SSD, you cannot remove it while the firewall is powered on.
- Remove one of the SSDs—If you have two SSDs, you can remove one.

- Add a second SSD—If you have one SSD, you can add a second SSD and form a RAID.

  When you use only one SSD in Secure Firewall devices, the **Drive State** will show as **degraded**.



Caution Do not remove an SSD without first removing it from the RAID using this procedure. You can cause data loss.

Procedure

Step 1 Remove one of the SSDs.

a) Remove the SSD from the RAID.

raid remove-secure local-disk {**1** | **2**}

The **remove-secure** keyword removes the SSD from the RAID, disables the self-encrypting disk feature, and performs a secure erase of the SSD. If you only want to remove the SSD from the RAID and want to keep the data intact, you can use the **remove** keyword.

Example:

```
ciscoasa(config)# raid remove-secure local-disk 2
```

b) Monitor the RAID status until the SSD no longer shows in the inventory.

show raid

After the SSD is removed from the RAID, the **Operability** and **Drive State** will show as **degraded**. The second drive will no longer be listed as a member disk.

Example:

```
ciscoasa# show raid
Virtual Drive
ID:                         1
Size (MB):                  858306
Operability:                operable
Presence:                   equipped
Lifecycle:                  available
Drive State:                optimal
Type:                       raid
Level:                      raid1
Max Disks:                  2
Meta Version:               1.0
Array State:                active
Sync Action:                idle
Sync Completed:             unknown
Degraded:                   0
Sync Speed:                 none

RAID member Disk:
Device Name:                nvme0n1
Disk State:                 in-sync
Disk Slot:                  1
Read Errors:                0
```

```
Recovery Start:             none
Bad Blocks:
Unacknowledged Bad Blocks:

Device Name:                nvme1n1
Disk State:                 in-sync
Disk Slot:                  2
Read Errors:                0
Recovery Start:             none
Bad Blocks:
Unacknowledged Bad Blocks:

ciscoasa# show raid
Virtual Drive
ID:                         1
Size (MB):                  858306
Operability:                degraded
Presence:                   equipped
Lifecycle:                  available
Drive State:                degraded
Type:                       raid
Level:                      raid1
Max Disks:                  2
Meta Version:               1.0
Array State:                active
Sync Action:                idle
Sync Completed:             unknown
Degraded:                   1
Sync Speed:                 none

RAID member Disk:
Device Name:                nvme0n1
Disk State:                 in-sync
Disk Slot:                  1
Read Errors:                0
Recovery Start:             none
Bad Blocks:
Unacknowledged Bad Blocks:
```

c) Physically remove the SSD from the chassis.

Step 2 Add an SSD.

a) Physically add the SSD to the empty slot.

b) Add the SSD to the RAID.

raid add local-disk {**1** | **2**}

It can take several hours to complete syncing the new SSD to the RAID, during which the firewall is completely operational. You can even reboot, and the sync will continue after it powers up. Use the **show raid** command to show the status.

If you install an SSD that was previously used on another system, and is still locked, enter the following command:

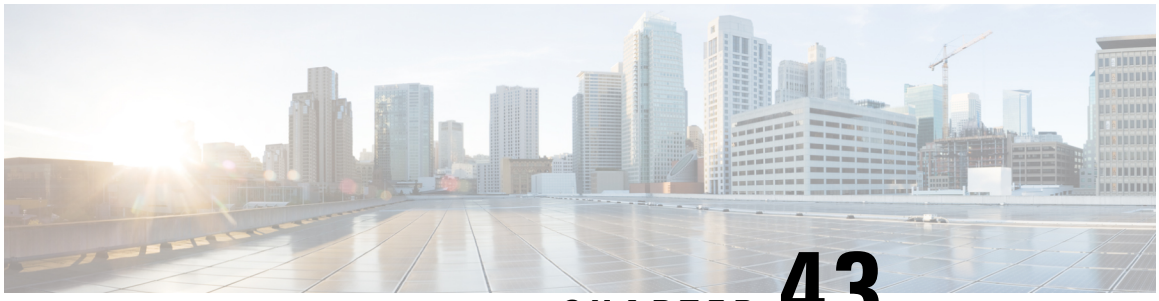
raid add local-disk {**1** | **2**} *psid*

The *psid* is printed on the label attached to the back of the SSD. Alternatively, you can reboot the system, and the SSD will be reformatted and added to the RAID.

History for Software and Configurations

Feature Name	Platform Releases	Feature Information
Secure Copy client and server	9.1(5)/9.2(1)	The ASA now supports the Secure Copy (SCP) client and server to transfer files to and from a SCP server. We introduced the following commands: **ssh pubkey-chain**, **server (ssh pubkey-chain)**, **key-string**, **key-hash**, **ssh stricthostkeycheck**. We modified the following command: **copy scp**.
Configurable SSH encryption and integrity ciphers	9.1(7)/9.4(3)/9.5(3)/9.6(1)	Users can select cipher modes when doing SSH encryption management and can configure HMAC and encryption for varying key exchange algorithms. You might want to change the ciphers to be more or less strict, depending on your application. Note that the performance of secure copy depends partly on the encryption cipher used. By default, the ASA negotiates one of the following algorithms in order: 3des-cbc aes128-cbc aes192-cbc aes256-cbc aes128-ctr aes192-ctr aes256-ctr. If the first algorithm proposed (3des-cbc) is chosen, then the performance is much slower than a more efficient algorithm such as aes128-cbc. To change the proposed ciphers, use **ssh cipher encryption custom aes128-cbc**, for example. We introduced the following commands: **ssh cipher encryption**, **ssh cipher integrity**
Auto Update server certificate verification enabled by default	9.2(1)	The Auto Update server certificate verification is now enabled by default; for new configurations, you must explicitly disable certificate verification. If you are upgrading from an earlier release, and you did not enable certificate verification, then certificate verification is not enabled, and you see the following warning: `WARNING: The certificate provided by the auto-update servers will not be verified. In order to verify this certificate please use the verify-certificate option.` The configuration will be migrated to explicitly configure no verification. **auto-update server no-verification** We modified the following command: **auto-update server** {**verify-certificate** | **no-verification**}.

Feature Name	Platform Releases	Feature Information
System backup and restore using the CLI	9.3(2)	You can now back up and restore complete system configurations, including images and certificates, using the CLI. We introduced the following commands: **backup** and **restore**.
Recovering and loading a new ASA 5506W-X image	9.4(1)	We now support the recovery and loading of a new ASA 5506W-X image. We introduced the following command: **hw-module module wlan recover image**.
Automatic Backup and Restore for the ISA 3000	9.7(1)	You can enable auto-backup and/or auto-restore functionality using pre-set parameters in the backup and restore commands. The use cases for these features include initial configuration from external media; device replacement; roll back to an operable state. We introduced the following commands: **backup-package location, backup-package auto, show backup-package status, show backup-package summary**
CiscoSSH stack requires SSH access when using the SCP client	9.17(1)	If you use the CiscoSSH stack, to use the ASA **copy** command to copy a file to or from an SCP server, you have to enable SSH access on the ASA for the SCP server subnet/host using the **ssh** command.
RAID support for SSDs on the Secure Firewall 3100	9.17(1)	The SSDs are self-encrypting drives (SEDs), and if you have 2 SSDs, they form a software RAID. New/Modified commands: **raid, show raid, show ssd**



CHAPTER 43

Response Automation for System Events

This chapter describes how to configure the Embedded Event Manager (EEM).

- About the EEM, on page 1255
- Guidelines for the EEM, on page 1256
- Configure the EEM, on page 1257
- Examples for the EEM, on page 1264
- Monitoring the EEM, on page 1265
- History for the EEM, on page 1266

About the EEM

The EEM service enables you to debug problems and provides general purpose logging for troubleshooting. There are two components: events to which the EEM responds or listens, and event manager applets that define actions as well as the events to which the EEM responds. You may configure multiple event manager applets to respond to different events and perform different actions.

Supported Events

The EEM supports the following events:

- Syslog—The ASA uses syslog message IDs to identify syslog messages that trigger an event manager applet. You may configure multiple syslog events, but the syslog message IDs may not overlap within a single event manager applet.
- Timers—You may use timers to trigger events. You may configure each timer only once for each event manager applet. Each event manager applet may have up to three timers. The three types of timers are the following:
  - Watchdog (periodic) timers trigger an event manager applet after the specified time period following the completion of the applet actions and restart automatically.
  - Countdown (one-shot) timers trigger an event manager applet once after the specified time period and do not restart unless they are removed, then re-added.
  - Absolute (once-a-day) timers cause an event to occur once a day at a specified time, and restart automatically. The time-of-day format is in hh:mm:ss.

    You may configure only one timer event of each type for each event manager applet.

- None—The none event is triggered when you run an event manager applet manually using the CLI or ASDM.
- Crash—The crash event is triggered when the ASA crashes. In some scenarios, a force crash is triggered:

  If ASA is configured to reload on block depletion, and when ASA remains out of memory for a configured duration, it sends out syslog and collects troubleshoot data. ASA force crashes and triggers the reload process to release the memory block. In a HA setup, under such cases, failover is triggered. On a cluster setup, the node leaves the cluster.

  Regardless of the value of the **output** command, the **action** commands are directed to the crashinfo file. The output is generated before the **show tech** command.

Actions on Event Manager Applets

When an event manager applet is triggered, the actions on the event manager applet are performed. Each action has a number that is used to specify the sequence of the actions. The sequence number must be unique within an event manager applet. You may configure multiple actions for an event manager applet. The commands are typical CLI commands, such as **show blocks**.

Output Destinations

You may send the output from the actions to a specified location using the **output** command. Only one output value may be enabled at any one time. The default value is **output none**. This value discards any output from the **action** commands. The command runs in global configuration mode as a user with privilege level 15 (the highest). The command may not accept any input, because it is disabled. You may send the output of the **action** CLI commands to one of three locations:

- **None**, which is the default and discards the output
- **Console**, which sends the output to the ASA console
- **File**, which sends the output to a file. The following four file options are available:
  - **Create a unique file**, which creates a new, uniquely named file each time that an event manager applet is invoked
  - **Create/overwrite a file**, which overwrites a specified file each time that an event manager applet is invoked.
  - **Create/append to a file**, which appends to a specified file each time that an event manager applet is invoked. If the file does not yet exist, it is created.
  - **Create a set of files**, which creates a set of uniquely named files that are rotated each time that an event manager applet is invoked.

Guidelines for the EEM

This section describes guidelines and limitations that you should check before configuring the EEM.

Context Mode Guidelines

Not supported in multiple context mode.

Additional Guidelines

- During a crash, the state of the ASA is generally unknown. Some commands may not be safe to run during this condition.
- The name of an event manager applet may not contain spaces.
- You cannot modify the None event and Crashinfo event parameters.
- Performance may be affected because syslog messages are sent to the EEM for processing.
- The default output is **output none** for each event manager applet. To change this setting, you must enter a different output value.
- You may have only one output option defined for each event manager applet.

Configure the EEM

Configuring the EEM consists of the following tasks:

Procedure

Step 1 Create an Event Manager Applet and Configure Events, on page 1257.
Step 2 Configure an Action and Destinations for Output from an Action, on page 1259.
Step 3 Run an Event Manager Applet, on page 1261.
Step 4 Track Memory Allocation and Memory Usage, on page 1261.

Create an Event Manager Applet and Configure Events

To create an event manager applet and configure events, perform the following steps:

Procedure

Step 1 Create an event manager applet and enter event manager applet configuration mode.

event manager applet *name*

Example:

```
ciscoasa(config)# event manager applet exampleapplet1
```

The *name* argument may be up to 32 alphanumeric characters long. Spaces are not allowed.

To remove an event manager applet, enter the **no** form of this command.

Step 2 Describe an event manager applet.

description *text*

Example:

```
ciscoasa(config-applet)# description applet1example
```

The *text* argument may be up to 256 characters long. You may include spaces in description text if it is placed within quotes.

Step 3 To configure a specified event, enter one of the following commands. To remove the configured event, enter the **no** form of each of the commands.

- To configure a syslog event, identify a single syslog message or a range of syslog messages that trigger an event manager applet.

  event syslog id *nnnnnn* [*-nnnnnn*] [**occurs** *n*] [**period** *seconds*]

  Example:

  ```
  ciscoasa(config-applet)# event syslog id 106201
  ```

  The *nnnnnn* argument identifies the syslog message ID. The **occurs** *n* keyword-argument pair indicates the number of times that the syslog message must occur for an event manager applet to be invoked. The default is 1 occurrence every 0 seconds. Valid values are from 1 - 4294967295. The **period** *seconds* keyword-argument pair indicates the number of seconds in which the event must occur, and limits how frequently an event manager applet is invoked to at most once in the configured period. Valid values are from 0 - 604800. A value of 0 means that no period is defined.

- To configure an event to occur once per configured period and restart automatically.

  event timer watchdog time *seconds*

  Example:

  ```
  ciscoasa(config-applet)# event timer watchdog time 30
  ```

  The number of seconds may range from 1 - 604800.

- To configure an event to occur once and not restart unless it is removed, then re-added.

  event timer countdown time *seconds*

  Example:

  ```
  ciscoasa(config-applet)# event timer countdown time 60
  ```

  The number of seconds may range from 1 - 604800. Use the **no** form of this command remove a countdown timer event.

  Note
  This timer reruns when you reboot if it is the startup configuration.

- To configure an event to occur once a day at a specified time and restart automatically.

  event timer absolute time *hh:mm:ss*

  Example:

  ```
  ciscoasa(config-applet)# event timer absolute time 10:30:20
  ```

  The time-of-day format is in hh:mm:ss. The time range is from 00:00:00 (midnight) to 23:59:59.

- Trigger a crash event when the ASA crashes.

  event crashinfo

  Example:

  ```
  ciscoasa(config-applet)# event crashinfo
  ```

  Regardless of the value of the **output** command, the **action** commands are directed to the crashinfo file. The output is generated before the **show tech** command.

Configure an Action and Destinations for Output from an Action

To configure an action and specific destinations for sending output from an action, perform the following steps:

Procedure

Step 1 Configure an action on an event manager applet.

action *n* **cli command** *"command"*

Example:

```
ciscoasa(config-applet)# action 1 cli command "show version"
```

The *n* option is an action ID. Valid IDs range from 0 - 4294967295. The value of the *command* option must be in quotes; otherwise, an error occurs if the command consists of more than one word. The command runs in global configuration mode as a user with privilege level 15 (the highest). The command may not accept any input, because it is disabled. Use the **noconfirm** option if the command has it available.

Step 2 Choose one of the available output destination options. Use the **no** form of each command to remove an output destination,

- The **None** option discards any output from the **action** commands, which is the default setting:

  output none

  Example:

```
ciscoasa(config-applet)# output none
```

- The **Console** option sends the output of the **action** commands to the console.

  output console

  Example:

  ```
  ciscoasa(config-applet)# output console
  ```

  Note
  Running this command affects performance.

- The **New File** option sends the output of the **action** commands to a new file for each event manager applet that is invoked.

  output file new

  Example:

  ```
  ciscoasa(config-applet)# output file new
  ```

  The filename has the format of eem-*applet-timestamp*.log, in which *applet* is the name of the event manager applet and *timestamp* is a dated time stamp in the format of YYYYMMDD-hhmmss.

- The **New Set of Rotated Files** option creates a set of files that are rotated. When a new file is to be written, the oldest file is deleted, and all subsequent files are renumbered before the first file is written.

  output file rotate *n*

  Example:

  ```
  ciscoasa(config-applet)# output file rotate 50
  ```

  The newest file is indicated by 0, and the oldest file is indicated by the highest number (*n*-1). The *n* option is the rotate value. Valid values range from 2 - 100. The filename format is eem-*applet-x*.log, in which *applet* is the name of the applet, and *x* is the file number.

- The **Single Overwritten File** option writes the **action** command output to a single file, which is overwritten every time.

  output file overwrite *filename*

  Example:

  ```
  ciscoasa(config-applet)# output file overwrite examplefile1
  ```

  The *filename* argument is a local (to the ASA) filename. This command may also use FTP, TFTP, and SMB targeted files.

- The **Single Appended File** option writes the **action** command output to a single file, but that file is appended to every time.

  output file append *filename*

Example:

```
ciscoasa(config-applet)# output file append examplefile1
```

The *filename* argument is a local (to the ASA) filename.

Run an Event Manager Applet

To run an event manager applet, perform the following steps:

Procedure

Run an event manager applet.

event manager run *applet*

Example:

```
ciscoasa# event manager run exampleapplet1
```

If you run an event manager applet that has not been configured with the **event none** command, an error occurs. The *applet* argument is the name of the event manager applet.

Track Memory Allocation and Memory Usage

To log memory allocation and memory usage, perform the following steps:

Procedure

Step 1 Enable memory logging.

memory logging [1024-4194304] [**wrap**] [**size** [1-2147483647]] [**process** *process-name*] [**context** *context-name*]

Example:

```
ciscoasa(config)# memory logging 202980
```

The only required argument is the number of entries in the memory logging buffer. The **wrap** option tells the memory logging utility to save the buffer when it wraps. It can only be saved once.

If the memory logging buffer wraps multiple times, it can be overwritten. When the buffer wraps, a trigger is sent to the event manager to enable saving of the data. The **size** option monitors a particular size. The **process** option monitors a particular process.

Note
The Checkheaps process is completely ignored as a process because it uses the memory allocator in a non-standard way.

The **context** option records memory logging for a given virtual context by the given name.

To change memory logging parameters, you must disable it, then reenable it.

Step 2 Display the memory logging results.

```
show memory logging [brief | wrap]
show memory logging include [address] [caller] [operator] [size] [process] [time] [context]
```

Example:

```
ciscoasa# show memory logging
Number of free                          6
Number of calloc                        0
Number of malloc                        8
Number of realloc-new                   0
Number of realloc-free                  0
Number of realloc-null                  0
Number of realloc-same                  0
Number of calloc-fail                   0
Number of malloc-fail                   0
Number of realloc-fail                  0
Total operations 14
Buffer size: 50 (3688 x2 bytes)
process=[ci/console] time=[13:26:33.407] oper=[malloc]
addr=0x00007fff2cd0a6c0 size=72 @ 0x00000000016466ea 0x0000000002124542
0x000000000131911a 0x0000000000442bfd process=[ci/console] time=[13:26:33.407] oper=[free]
addr=0x00007fff2cd0a6c0 size=72 @ 0x00000000021246ef 0x00000000013193e8
0x0000000000443455 0x0000000001318f5b
process=[CMGR Server Process] time=[13:26:35.964] oper=[malloc]
addr=0x00007fff2cd0aa00 size=16 @ 0x00000000016466ea 0x0000000002124542
0x000000000182774d 0x000000000182cc8a process=[CMGR Server Process]
time=[13:26:35.964] oper=[malloc]
addr=0x00007fff224bb9f0 size=512 @ 0x00000000016466ea 0x0000000002124542
0x0000000000bfef9a 0x0000000000bff606 process=[CMGR Server Process]
time=[13:26:35.964] oper=[free]
addr=0x00007fff224bb9f0 size=512 @ 0x00000000021246ef 0x0000000000bff3d8
0x0000000000bff606 0x000000000182ccb0
process=[CMGR Server Process] time=[13:26:35.964] oper=[malloc]
addr=0x00007fff224b9460 size=40 @ 0x00000000016466ea 0x0000000002124542
0x0000000001834188 0x000000000182ce83
process=[CMGR Server Process] time=[13:26:37.964] oper=[free]
addr=0x00007fff2cd0aa00 size=16 @ 0x00000000021246ef 0x0000000001827098
0x000000000182c08d 0x000000000182c262 process=[CMGR Server Process]
time=[13:26:37.964] oper=[free]
addr=0x00007fff224b9460 size=40 @ 0x00000000021246ef 0x000000000182711b
0x000000000182c08d 0x000000000182c262 process=[CMGR Server Process]
time=[13:26:38.464] oper=[malloc]
addr=0x00007fff2cd0aa00 size=16 @ 0x00000000016466ea 0x0000000002124542
0x000000000182774d 0x000000000182cc8a process=[CMGR Server Process]
time=[13:26:38.464] oper=[malloc]
addr=0x00007fff224bb9f0 size=512 @ 0x00000000016466ea 0x0000000002124542
0x0000000000bfef9a 0x0000000000bff606 process=[CMGR Server Process]
time=[13:26:38.464] oper=[free]
addr=0x00007fff224bb9f0 size=512 @ 0x00000000021246ef 0x0000000000bff3d8
0x0000000000bff606 0x000000000182ccb0
```

```
process=[CMGR Server Process] time=[13:26:38.464] oper=[malloc]
addr=0x00007fff224b9460 size=40 @ 0x00000000016466ea 0x0000000002124542
0x0000000001834188 0x000000000182ce83
process=[ci/console] time=[13:26:38.557] oper=[malloc]
addr=0x00007fff2cd0a6c0 size=72 @ 0x00000000016466ea 0x0000000002124542
0x000000000131911a 0x0000000000442bfd process=[ci/console] time=[13:26:38.557] oper=[free]
addr=0x00007fff2cd0a6c0 size=72 @ 0x00000000021246ef 0x00000000013193e8
0x0000000000443455 0x0000000001318f5b

ciscoasa# show memory logging include process operation size
Number of free                          6
Number of calloc                        0
Number of malloc                        8
Number of realloc-new                   0
Number of realloc-free                  0
Number of realloc-null                  0
Number of realloc-same                  0
Number of calloc-fail                   0
Number of malloc-fail                   0
Number of realloc-fail                  0
Total operations 14
Buffer size: 50 (3688 x2 bytes)
process=[ci/console] oper=[malloc] size=72 process=[ci/console] oper=[free]
size=72 process=[CMGR Server Process] oper=[malloc] size=16
process=[CMGR Server Process] oper=[malloc] size=512 process=[CMGR Server Process]
oper=[free] size=512 process=[CMGR Server Process] oper=[malloc] size=40
process=[CMGR Server Process] oper=[free] size=16 process=[CMGR Server Process]
oper=[free] size=40 process=[CMGR Server Process] oper=[malloc] size=16
process=[CMGR Server Process] oper=[malloc] size=512 process=[CMGR Server Process]
oper=[free] size=512 process=[CMGR Server Process] oper=[malloc] size=40
process=[ci/console] oper=[malloc] size=72 process=[ci/console]
oper=[free] size=72 ciscoasa# show memory logging brief
Number of free                          6
Number of calloc                        0
Number of malloc                        8
Number of realloc-new                   0
Number of realloc-free                  0
Number of realloc-null                  0
Number of realloc-same                  0
Number of calloc-fail                   0
Number of malloc-fail                   0
Number of realloc-fail                  0
Total operations 14
Buffer size: 50 (3688 x2 bytes)
```

Without any options, **show memory logging** displays statistics and then the recorded operations. The **brief** option shows only statistics. The **wrap** option shows the buffer upon wrap, then purges the data so that duplicate data does not appear or get saved. The **include** option includes only the specified fields in the output. You can specify the fields in any order, but they always appear in the following order:

a. Process

b. Time

c. Context (unless in single mode)

d. Operation (free/malloc/etc.)

e. Address

f. Size

g. Callers

The output format is:

process=[XXX] time=[XXX] context=[XXX] oper=[XXX] address=0xXXXXXXXX size=XX @ XXXXXXXX XXXXXXXX XXXXXXXX XXXXXXXX

Up to four caller addresses appear. The types of operations are listed in the output (Number of...) shown in the example.

Step 3 Respond to memory logging wrap events.

event memory-logging-wrap

Example:

```
ciscoasa(config)# event manager applet memlog
ciscoasa(config)# event memory-logging-wrap
ciscoasa(config)# action 0 cli command "show memory logging wrap"
ciscoasa(config)# output file append disk0:/memlog.log
```

The example shows an applet that records all memory allocations. When wrap is enabled for memory logging, the memory logger sends an event to the event manager to trigger configured applets.

Examples for the EEM

The following example shows an event manager applet that records block leak information every hour and writes the output to a rotating set of log files, keeping a day's worth of logs:

```
ciscoasa(config)# event manager applet blockcheck
ciscoasa(config-applet)# description "Log block usage"
ciscoasa(config-applet)# event timer watchdog time 3600
ciscoasa(config-applet)# output rotate 24
ciscoasa(config-applet)# action 1 cli command "show blocks old"
```

The following example shows an event manager applet that reboots the ASA every day at 1 am, saving the configuration as needed:

```
ciscoasa(config)# event manager applet dailyreboot
ciscoasa(config-applet)# description "Reboot every night"
ciscoasa(config-applet)# event timer absolute time 1:00:00
ciscoasa(config-applet)# output none
ciscoasa(config-applet)# action 1 cli command "reload save-config noconfirm"
```

The following example shows event manager applets that disable the given interface between midnight and 3 am.

```
ciscoasa(config)# event manager applet disableintf
ciscoasa(config-applet)# description "Disable the interface at midnight"
ciscoasa(config-applet)# event timer absolute time 0:00:00
ciscoasa(config-applet)# output none
ciscoasa(config-applet)# action 1 cli command "interface GigabitEthernet 0/0"
```

```
ciscoasa(config-applet)# action 2 cli command "shutdown"
ciscoasa(config-applet)# action 3 cli command "write memory"

ciscoasa(config)# event manager applet enableintf
ciscoasa(config-applet)# description "Enable the interface at 3am"
ciscoasa(config-applet)# event timer absolute time 3:00:00
ciscoasa(config-applet)# output none
ciscoasa(config-applet)# action 1 cli command "interface GigabitEthernet 0/0"
ciscoasa(config-applet)# action 2 cli command "no shutdown"
ciscoasa(config-applet)# action 3 cli command "write memory"
```

Monitoring the EEM

See the following commands to monitor the EEM.

- **clear configure event manager**

  This command removes the event manager running configuration.

- **clear configure event manager applet** *appletname*

  This command removes the named event manager applet from the configuration.

- **show counters protocol eem**

  This command shows the counters for the event manager.

- **show event manager**

  This command shows information about the configured event manager applets, including hit counts and when the event manager applets were last invoked.

- **show memory logging, show memory logging include**

  These commands show statistics about the memory allocations and memory usage.

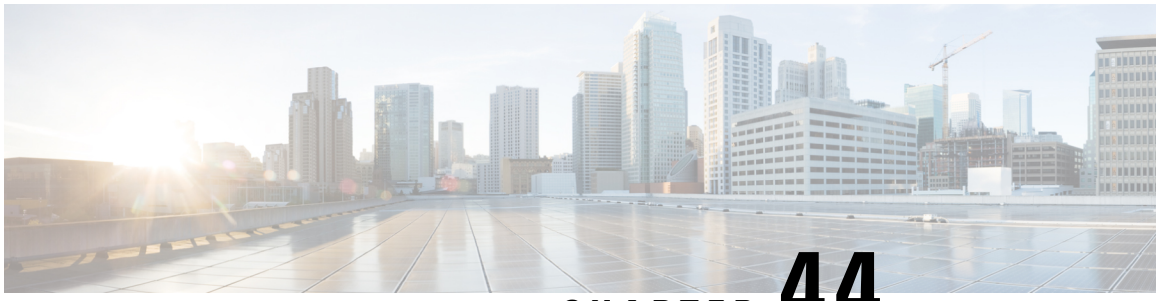
- **show running-config event manager**

  This command shows the running configuration of the event manager.

History for the EEM

Table 57: History for the EEM

Feature Name	Platform Releases	Description
Embedded Event Manager (EEM)	9.2(1)	The EEM service enables you to debug problems and provides general purpose logging for troubleshooting. There are two components: events to which the EEM responds or listens, and event manager applets that define actions as well as the events to which the EEM responds. You may configure multiple event manager applets to respond to different events and perform different actions. We introduced or modified the following commands: **event manager applet**, **description**, **event syslog id**, **event none**, **event timer** {**watchdog time** *seconds* | **countdown time** *seconds* | **absolute time** *hh:mm:ss*}, **event crashinfo**, **action cli command**, **output** {**none** | **console** | **file** {**append** *filename* | **new** | **overwrite** *filename* | **rotate** *n*}}, **show running-config event manager**, **event manager run**, **show event manager**, **show counters protocol eem**, **clear configure event manager**, **debug event manager**, **debug menu eem**.
Memory tracking for the EEM	9.4(1)	We have added a new debugging feature to log memory allocations and memory usage, and to respond to memory logging wrap events. We introduced or modified the following commands: **memory logging**, **show memory logging**, **show memory logging include**, **event memory-logging-wrap**.



CHAPTER 44

Testing and Troubleshooting

This chapter describes how to troubleshoot the ASA and test basic connectivity.

- Recover Enable and Telnet Passwords, on page 1267
- View Debugging Messages, on page 1271
- Packet Capture, on page 1271
- View the Crash Dump, on page 1277
- View the Coredump, on page 1277
- CPU Usage and Reporting, on page 1277
- Test Your Configuration, on page 1282
- Monitoring Connections, on page 1295
- History for Testing and Troubleshooting , on page 1295

Recover Enable and Telnet Passwords

If you forget the enable or Telnet passwords, you can recover them for ASAv and ISA 3000 models. You must perform the task using the CLI.

Note For other platforms, you cannot recover lost passwords. You can only restore the factory default configuration, and reset the passwords to the default. For Firepower 4100/9300, see the FXOS configuration guide. For other models, see the FXOS troubleshooting guide.

Recover Passwords on the ISA 3000

To recover passwords for the ISA 3000 perform the following steps:

Procedure

Step 1 Connect to the ASA console port.

Step 2 Power off the ASA, then power it on.

Step 3 After startup, press the **Escape** key when you are prompted to enter ROMMON mode.

Step 4 To update the configuration register value, enter the following command:

```
rommon #1> confreg 0x41

You must reset or power cycle for new config to take effect
```

The ASA displays the current configuration register value and a list of configuration options. Record the current configuration register value, so you can restore it later.

```
Configuration Register: 0x00000041

Configuration Summary
 [ 0 ] password recovery
 [ 1 ] display break prompt
 [ 2 ] ignore system configuration
 [ 3 ] auto-boot image in disks
 [ 4 ] console baud: 9600
 boot: ...... auto-boot index 1 image in disks
```

Step 5 Reload the ASA by entering the following command:

```
rommon #2> boot
Launching BootLoader...
Boot configuration file contains 1 entry.

Loading disk0:/asa932-226-k8.bin... Booting...Loading...
```

The ASA loads the default configuration instead of the startup configuration.

Step 6 Access the privileged EXEC mode by entering the following command:

```
ciscoasa# enable
```

Step 7 When prompted for the password, press **Enter**.

The password is blank.

Step 8 Load the startup configuration by entering the following command:

```
ciscoasa# copy startup-config running-config
```

Step 9 Access the global configuration mode by entering the following command:

```
ciscoasa# configure terminal
```

Step 10 Change the passwords, as required, in the default configuration by entering the following commands:

```
ciscoasa(config)# password password
ciscoasa(config)# enable password password
ciscoasa(config)# username name password password
```

Step 11 Load the default configuration by entering the following command:

```
ciscoasa(config)# no config-register
```

The default configuration register value is 0x1. See the command reference for more information about the configuration register.

Step 12 Save the new passwords to the startup configuration by entering the following command:

```
ciscoasa(config)# copy running-config startup-config
```

Recover Passwords or Images on the ASAv

To recover passwords or images on the ASAv, perform the following steps:

Procedure

Step 1 Copy the running configuration to a backup file on the ASAv:

copy running-config *filename*

Example:

```
ciscoasa# copy running-config backup.cfg
```

Step 2 Restart the ASAv:

reload

Step 3 From the GNU GRUB menu, press the down arrow, choose the <**filename**> **with no configuration load** option, then press **Enter**. The filename is the default boot image filename on the ASAv. The default boot image is never automatically booted through the **fallback** command. Then load the selected boot image.

```
GNU GRUB version 2.0(12)4
bootflash:/asa100123-20-smp-k8.bin
bootflash: /asa100123-20-smp-k8.bin with no configuration load
```

Example:

```
GNU GRUB version 2.0(12)4
bootflash: /asa100123-20-smp-k8.bin with no configuration load
```

Step 4 Copy the backup configuration file to the running configuration.

copy *filename* **running-config**

Example:

```
ciscoasa (config)# copy backup.cfg running-config
```

Step 5 Reset the password.

enable password *password*

Example:

```
ciscoasa(config)# enable password cisco123
```

Step 6 Save the new configuration.

write memory

Example:

```
ciscoasa(config)# write memory
```

Disable Password Recovery for ISA 3000 Hardware

Note You cannot disable password recovery on the ASAv, FirepowerSecure Firewall models.

To disable password recovery to ensure that unauthorized users cannot use the password recovery mechanism to compromise the ASA, perform the following steps.

Before you begin

On the ASA, the **no service password-recovery** command prevents you from entering ROMMON mode with the configuration intact. When you enter ROMMON mode, the ASA prompts you to erase all Flash file systems. You cannot enter ROMMON mode without first performing this erasure. If you choose not to erase the Flash file system, the ASA reloads. Because password recovery depends on using ROMMON mode and maintaining the existing configuration, this erasure prevents you from recovering a password. However, disabling password recovery prevents unauthorized users from viewing the configuration or inserting different passwords. In this case, to restore the system to an operating state, load a new image and a backup configuration file, if available.

The **service password-recovery** command appears in the configuration file for information only. When you enter the command at the CLI prompt, the setting is saved in NVRAM. The only way to change the setting is to enter the command at the CLI prompt. Loading a new configuration with a different version of the command does not change the setting. If you disable password recovery when the ASA is configured to ignore the startup configuration at startup (in preparation for password recovery), then the ASA changes the setting to load the startup configuration as usual. If you use failover, and the standby unit is configured to ignore the startup configuration, then the same change is made to the configuration register when the **no service password-recovery** command replicates to the standby unit.

Procedure

Disable password recovery.

no service password-recovery

Example:

```
ciscoasa (config)# no service password-recovery
```

View Debugging Messages

Because debugging output is assigned high priority in the CPU process, it can render the system unusable. For this reason, use **debug** commands only to troubleshoot specific problems or during troubleshooting sessions with Cisco TAC. Moreover, it is best to use **debug** commands during periods of less network traffic and fewer users. Debugging during these periods decreases the likelihood that increased **debug** command processing overhead will affect system use. To enable debugging messages, see the **debug** commands in the command reference.

Packet Capture

Capturing packets may be useful when troubleshooting connectivity problems or monitoring suspicious activity. We recommend that you contact Cisco TAC if you want to use the packet capture service.

Guidelines for Packet Capture

Context Mode

- You can configure captures on the cluster control link within a context; only the packet that is associated with the context sent in the cluster control link is captured.
- You can only configure one capture for a shared VLAN; if you configure a capture in multiple contexts on the shared VLAN, then only the last capture that was configured is used.
- If you remove the last-configured (active) capture, no captures become active, even if you have previously configured a capture in another context; you must remove the capture and add it again to make it active.
- All traffic that enters the interface to which the capture is attached is captured, including traffic to other contexts on the shared VLAN. Therefore, if you enable a capture in Context A for a VLAN that is also used by Context B, both Context A and Context B ingress traffic are captured.
- For egress traffic, only the traffic of the context with the active capture is captured. The only exception is when you do not enable the ICMP inspection (therefore the ICMP traffic does not have a session in the accelerated path). In this case, both ingress and egress ICMP traffic for all contexts on the shared VLAN is captured.

Additional Guidelines

- If the ASA receives packets with an incorrectly formatted TCP header and drops them because of the *invalid-tcp-hdr-length* ASP drop reason, the **show capture** command output on the interface where those packets are received does not show those packets.
- You can only capture IP traffic; you cannot capture non-IP packets such as ARPs.
- For inline SGT tagged packets, captured packets contain an additional CMD header that your PCAP viewer might not understand.
- Packet captures include packets that the system modifies or injects into the connection due to inspection, NAT, TCP normalization, or other features that adjust the content of a packet.
- The trace of the lifespan of an injected virtual packet in a datapath does not exactly reflect how the datapath handles the physical packets. This difference depends on the software version, configuration, and type of the injected virtual packets. Following are configuration settings that might lead to the disparity:
  - at least 2 NAT statements for the same host exist.
  - forward and reverse flows of a connection having different protocols. For example, forward flow is UDP or TCP, reverse flow is ICMP.
  - ICMP error inspection being enabled.

Capture Packets

To capture packets, perform the following steps.

Procedure

Step 1 Enable packet capture capabilities for packet sniffing and network fault isolation:

capture *capture_name* [**type** {**asp-drop** [**all** | *drop-code*] | **tls-proxy** | **raw-data** | **isakmp** [**ikev1** | **ikev2**] | **inline-tag** [*tag*] | **webvpn user** *webvpn-user*}] [**access-list** *access_list_name*] {**interface** {*interface_name* | **asa_dataplane** | **asa_mgmt_plane** | **cplane**} } [**buffer** *buf_size*] [**ethernet-type** *type*] [**reeinject-hide**] [**packet-length** *bytes*] [**circular-buffer**] [**trace** [**trace-count** *number*]] [**real-time** [**dump**] [**detail**]] [**file-size**] [**headers-only**] [**match** *protocol* {**host** *source-ip* | *source-ip mask* | **any** | **any4**|**any6**} [*operator src_port*] {**host** *dest_ip* |*dest_ip mask* | **any** | **any4**|**any6**} [*operator dest_port*]]

Example:

```
ciscoasa# capture captest interface inside
```

You must configure an interface for any packets to be captured. Use the same *capture_name* on multiple **capture** statements to capture multiple types of traffic.

The **type asp-drop** keyword captures packets dropped by the accelerated security path. In a cluster, dropped forwarded data packets from one unit to another are also captured. In multiple context mode, when this option is issued in the system execution space, all dropped data packets are captured; when this option is issued in a context, only dropped data packets that enter from interfaces belonging to the context are captured.

The **type raw-data** keywords capture inbound and outbound packets. This setting is the default.

The **inline-tag** *tag* keyword-argument pair specifies a tag for a particular SGT value or leaves it unspecified to capture a tagged packet with any SGT value.

The **buffer** keyword defines the buffer size used to store the packet. When the byte buffer is full, packet capture stops. When used in a cluster, this is the per-unit size, not the sum of all units. The **circular-buffer** keyword overwrites the buffer, starting from the beginning, when the buffer is full.

The **ethernet-type** keyword sets an ethernet type to capture. Supported Ethernet types include 8021Q, ARP, IP, IP6, LACP, PPPOED, PPPOES, RARP, and VLAN. An exception occurs with the 802.1Q or VLAN type. The 802.1Q tag is automatically skipped and the inner Ethernet type is used for matching. IP is the default ethernet type.

The **interface** keyword sets the name of the interface on which to use packet capture.

To capture packets on the dataplane, use the **asa_dataplane** keyword.

To configure the size of capture file, use the **file-size**keyword. The file size can be between 32 and 10000 MB.

If you want to capture only L2, L3 and L4 headers of packet without data in them, use the **headers-only** command.

The **match** keyword captures matching the protocol and source and destination IP addresses and optional ports. You can use this keyword up to three times in one command. The **any** keyword captures IPv4 traffic only. You can use the **any4** and **any6** keywords to capture the matching IPv4 and IPv6 network traffic respectively. The operator can be one of the following:

- lt—less than
- gt—greater than
- eq—equal to

The **real-time** keyword displays the captured packets continuously in real-time.

The **reinject-hide** keyword specifies that no reinjected packets will be captured and applies only in a clustering environment.

Note
If ACL optimization is configured, you cannot use the **access-list** command in capture. You can only use the **access-group** command. An error appears if you try to use the **access-list** command in this case.

Step 2 Capture cluster control-link traffic:

capture *capture_name* {**type lacp interface** *interface_id* [**buffer** *buf_size*] [**packet-length** *bytes*] [**circular-buffer**] [**real-time** [**dump**] [**detail**]]

capture *capture_name* **interface cluster** [**buffer** *buf_size*] [**cp-cluster**] [**ethernet-type** *type*] [**packet-length** *bytes*] [**circular-buffer**] [**trace** [**trace-count** *number*]] [**real-time** [**dump**] [**detail**]] [**trace**] [**match** *protocol* {**host** *source-ip* | *source-ip mask* | **any** | **any4**|**any6**} [*operator src_port*] {**host** *dest_ip* |*dest_ip mask* | **any** | **any4**|**any6**} [*operator dest_port*]]

Example:

```
ciscoasa# capture ccl type lacp interface GigabitEthernet0/0
ciscoasa# capture ccl interface cluster match udp any eq 49495 any
```

```
ciscoasa# capture ccl interface cluster match udp any any eq 49495
```

You can capture cluster control link traffic in two ways: to capture all the traffic on the cluster control link, use the **cluster** keyword for the interface name. To capture only cLACP packets, specify **type lacp**, and specify the physical interface ID instead of the interface name. There are two types of packets on the cluster control link: control plane packets and data plane packets, which both include forwarded data traffic and cluster LU messages. The TTL field in the IP address header is encoded to differentiate between these two types of packets. When forwarded data packets are captured, their clustering trailers are included in the capture file for debugging purposes.

The **cp-cluster** keyword only captures control plane packets on the cluster control link (and no data plane packets). This option is useful in the system in multiple context mode where you cannot match traffic using an ACL.

Step 3 Capture packets cluster-wide:

cluster exec capture *capture_name arguments*

Step 4 Stop the packet capture:

no capture *capture_name*

To terminate a real-time packet capture, enter **Ctrl + c.** To permanently remove the capture, use the **no** form of this command. The real-time option applies only to **raw-data** and **asp-drop** captures.

Step 5 To manually stop the packet capture without removing packets from the buffer:

capture *name* **stop**

Step 6 To start the capture again:

no capture *name* **stop**

Step 7 Capture persistent packet traces on cluster units:

cluster exec *capture_test* **persist**

Step 8 Clear persistent packet traces:

cluster exec *clear* **packet-trace**

Step 9 Capture decrypted IPsec packets:

cluster exec *capture_test* **include-decrypted**

Step 10 Clear the capture:

clear capture *capture_name*

Examples

Control Plane Packets

All packets to and from the control plane have a TTL of 255, and port number 49495 is used for the clustering control-plane listen port. The following example shows how to create a LACP capture for the clustering environment:

```
ciscoasa# capture lacp type lacp interface GigabitEthernet0/0
```

The following example shows how to create a capture for control path packets in the clustering link:

```
ciscoasa# capture cp interface cluster match udp any eq 49495 any
ciscoasa# capture cp interface cluster match udp any any eq 49495
```

Data Plane Packets

Data packets include those forwarded from one unit to another unit (its connection owner) and cluster LU messages. Regular cluster LU update messages have a TTL of 254, and there is a special LU packet that has a TTL of 253. This special LU packet is only for TCP, and it only happens when the director elects a new flow owner; the director sends back the requesting packet along with the CLU_FULL update packet. The LU packet is filled with the original packet's L3/L4 header to avoid a potential race condition at the receiver side. Forwarded data packets have a TTL of less than 4. The following example shows how to create a capture for data path packets in the cluster control link. To capture all inter-cluster dataplane "flow logical update" messages, use port 4193.

```
ciscoasa# access-list cc1 extended permit udp any any eq 4193
ciscoasa# access-list cc1 extended permit udp any eq 4193 any
ciscoasa# capture dp interface cluster access-list ccl
```

View a Packet Capture

You can view a packet capture at the CLI, in a browser, or download a capture to a server of your choice.

Procedure

Step 1 View the capture at the CLI:

[**cluster exec**] **show capture** [*capture_name*] [**access-list** *access_list_name*] [**count** *number*] [**decode**] [**detail**] [**dump**] [**packet-number** *number*]

Example:

```
ciscoasa# show capture capin

 8 packets captured

 1: 03:24:35.526812       192.168.10.10 > 203.0.113.3: icmp: echo request
 2: 03:24:35.527224       203.0.113.3 > 192.168.10.10: icmp: echo reply
 3: 03:24:35.528247       192.168.10.10 > 203.0.113.3: icmp: echo request
 4: 03:24:35.528582       203.0.113.3 > 192.168.10.10: icmp: echo reply
 5: 03:24:35.529345       192.168.10.10 > 203.0.113.3: icmp: echo request
 6: 03:24:35.529681       203.0.113.3 > 192.168.10.10: icmp: echo reply
 7: 03:24:57.440162       192.168.10.10 > 203.0.113.3: icmp: echo request
 8: 03:24:57.440757       203.0.113.3 > 192.168.10.10: icmp: echo reply
```

The **access-list** keyword displays information for packets that are based on IP or higher fields for the specific access list identification.

The **cluster exec** keyword lets you issue the **show capture** command in one unit and run the command in all the other units at the same time.

The **count** keyword displays the number of packets specified data.

The **decode** keyword is useful when a capture of type **isakmp** is applied to an interface. All ISAKMP data flowing through that interface will be captured after decryption and shown with more information after decoding the fields. The decoded output of the packets depend on the protocol of the packet. Typically, this command supports IP decode for the ICMP, UDP, and TCP protocols. From version 9.10(1), this command also supports IP decode for GRE and IPinIP.

The **detail** keyword displays additional protocol information for each packet.

The **dump** keyword displays a hexadecimal dump of the packets that are transported over the data link.

The **packet-number** keyword starts the display at the specified packet number.

Step 2 View the packet capture with your browser:

https://*ip_of_asa***/admin/capture/***capture_name***/pcap**

If you leave out the **pcap** keyword, then only the equivalent of the **show capture** *capture_name* command output is provided.

In multiple context mode, the **copy capture** command is available only in the system execution space.

Step 3 Copy the packet capture to a server. This example shows FTP.

[**cluster exec**] **copy /pcap capture:**[*context-name/*]*capture_name* **ftp://***username***:***password***@***server_ip***/***path*

If you leave out the **pcap** keyword, then only the equivalent of the **show capture** *capture_name* command output is provided.

Note
When you copy a packet capture to a disk, ensure that the capture filename is less than or equal to 63 characters. When the filename is more than 63 characters, though the packet capture is successful, copying the capture to a disk fails.

Examples

The following example shows a type asp-drop capture:

```
ciscoasa# capture asp-drop type asp-drop acl-drop
ciscoasa# show capture asp-drop

 2 packets captured

 1: 04:12:10.428093       192.168.10.10.34327 > 10.94.0.51.15868: S
    2669456341:2669456341(0) win 4128 <mss 536> Drop-reason: (acl-drop)
    Flow is denied by configured rule
 2: 04:12:12.427330       192.168.10.10.34327 > 10.94.0.51.15868: S
    2669456341:2669456341(0) win 4128 <mss 536> Drop-reason: (acl-drop)
    Flow is denied by configured rule
 2 packets shown

ciscoasa# show capture asp-drop
```

```
2 packets captured

1: 04:12:10.428093       192.168.10.10.34327 > 10.94.0.51.15868: S
   2669456341:2669456341(0) win 4128 <mss 536> Drop-reason: (acl-drop)
   Flow is denied by configured rule
2: 04:12:12.427330       192.168.10.10.34327 > 10.94.0.51.15868: S
   2669456341:2669456341(0) win 4128 <mss 536> Drop-reason: (acl-drop)
   Flow is denied by configured rule
2 packets shown
```

The following example shows an ethernet-type capture:

```
ciscoasa# capture arp ethernet-type arp interface inside
ciscoasa# show cap arp

22 packets captured

   1: 05:32:52.119485       arp who-has 10.10.3.13 tell 10.10.3.12
   2: 05:32:52.481862       arp who-has 192.168.10.123 tell 192.168.100.100
   3: 05:32:52.481878       arp who-has 192.168.10.50 tell 192.168.100.10
   4: 05:32:53.409723       arp who-has 10.106.44.135 tell 10.106.44.244
   5: 05:32:53.772085       arp who-has 10.106.44.108 tell 10.106.44.248
   6: 05:32:54.782429       arp who-has 10.106.44.135 tell 10.106.44.244
   7: 05:32:54.784695       arp who-has 10.106.44.1 tell 11.11.11.112:
```

View the Crash Dump

If the ASA or ASAv crashes, you can view the crash dump information. We recommend that you contact Cisco TAC if you want to interpret the crash dump. See the **show crashdump** command in the command reference.

View the Coredump

A coredump is a snapshot of the running program when the program has terminated abnormally or crashed. Coredumps are used to diagnose or debug errors and save a crash for future off-site analysis. Cisco TAC may request that you enable the coredump feature to troubleshoot application or system crashes on the ASA or ASAv. See the **coredump** command in the command reference.

CPU Usage and Reporting

The CPU Utilization report summarizes the percentage of the CPU used within the time specified. Typically, the core operates at approximately 30 to 40 percent of total CPU capacity during non-peak hours and approximately 60 to 70 percent capacity during peak hours.

vCPU Usage in the ASAv

Use the **show cpu usage** command on the ASAv to display CPU utilization statistics. The ASAv vCPU usage shows the amount of vCPUs used for the data path, control point, and external processes.

The vCPU usage reported by the cloud service provider (such as VMware, Azure, OCI, and so on) includes the ASAv usage as described plus:

- ASAv idle time
- %SYS overhead used for the ASA virtual VM
- Overhead of moving packets between vSwitches, vNICs, and pNICs. This overhead can be quite significant.

CPU Usage Example

The following is an example in which the reported vCPU usage is substantially different:

- ASAv reports: 40%
- DP: 35%
- External Processes: 5%
- vSphere reports: 95%
- ASA (as ASAv reports): 40%
- ASA idle polling: 10%
- Overhead: 45%

The overhead is used to perform hypervisor functions and to move packets between NICs and vNICs using the vSwitch.

Usage can exceed 100% because the ESXi server can use additional compute resources for overhead on behalf of the ASAv.

VMware CPU Usage Reporting

In vSphere, click the **VM Performance** tab, then click **Advanced** to display the **Chart Options** drop-down list, which shows vCPU usage for each state (%USER, %IDLE, %SYS, and so on) of the VM. This information is useful for understanding VMware's perspective on where CPU resources are being used.

On the ESXi server shell (you access the shell by using SSH to connect to the host), esxtop is available. Esxtop has a similar look and feel to the Linux **top** command and provides VM state information for vSphere performance, including the following:

- Details on vCPU, memory, and network usage
- vCPU usage for each state of each VM.
- Memory (type M while running) and network (type N while running), as well as statistics and the number of RX drops

ASAv and vCenter Graphs

There are differences in the CPU % numbers between the ASAv and vCenter:

- The vCenter graph numbers are always higher than the ASAv numbers.
- vCenter calls it %CPU usage; the ASAv calls it %CPU utilization.

The terms "%CPU utilization" and "%CPU usage" mean different things:

- CPU utilization provides statistics for physical CPUs.
- CPU usage provides statistics for logical CPUs, which is based on CPU hyperthreading. But because only one vCPU is used, hyperthreading is not turned on.

vCenter calculates the CPU % usage as follows:

Amount of actively used virtual CPUs, specified as a percentage of the total available CPUs

This calculation is the host view of the CPU usage, not the guest operating system view, and is the average CPU utilization over all available virtual CPUs in the virtual machine.

For example, if a virtual machine with one virtual CPU is running on a host that has four physical CPUs and the CPU usage is 100%, the virtual machine is using one physical CPU completely. The virtual CPU usage calculation is Usage in MHz / number of virtual CPUs x core frequency

When you compare the usage in MHz, both the vCenter and ASAv numbers match. According to the vCenter graph, MHz % CPU usage is calculated as 60/(2499 x 1 vCPU) = 2.4

Amazon CloudWatch CPU Usage Reporting

You can view the metrics explorer to monitor resources by their tags and properties. Perform the following steps to view CPU utilization statistics for a specific instance:

Procedure

Step 1 Open the **CloudWatch** console and choose **Metrics** in the navigation pane.

Step 2 Select the **EC2** metric namespace and select **Per-instance Metrics** dimension.

Step 3 Enter **CPUUtilization** in the search field and press Enter. Select the row for the required instance to display a graph for the **CPUUtilization** metric for that instance.

See Amazon CloudWatch documentation for more information.

ASAv and Amazon CloudWatch Graphs

The Amazon CloudWatch graph numbers are higher than the numbers because of the different ways in which CPU usage is calculated on the ASAv and the CloudWatch.

When ASAv is running in poll mode, each CPU runs a loop of lightweight commands instead of entering power saving mode or any other idle state. This improves performance by keeping each core active at all times instead of having to be turned on/off or have their clocks adjusted by the Intel power states.

Inside ASAv, this activity is understood to be idling behaviour, and the CPU usage is correctly calculated. However, on the Amazon CloudWatch, the idle behaviour appears like normal CPU activity because all CPU

cycles have instructions to run, and this causes the CloudWatch to display a high CPU usage percentage (85-90%).

Azure CPU Usage Reporting

Perform the following steps to view CPU utilization % across all monitored VMs by using VM Insights from the Azure Monitor:

Procedure

Step 1 Go to the Azure portal, select **Monitor** and choose **Virtual Machines** in the **Solutions** section.

Step 2 Select the **Performance** tab to display the **CPU Utilization %** chart. This chart displays the top five machines with the highest average processor utilization.

Perform the following steps to view the CPU utilization % chart directly from a specific Azure VM:

Procedure

Step 1 Go to the Azure portal and select **Virtual Machines**.

Step 2 From the list of VMs, choose a VM.

Step 3 In the **Monitoring** section, select **Insights**.

Step 4 Select the **Performance** tab.

See How to chart performance with VM Insights for more information.

ASAv and Azure Graphs

There are differences in the CPU % numbers between the ASAv and Azure. The Azure graph numbers are always higher than the ASAv numbers because Azure calculates the CPU % usage as the amount of actively used virtual CPUs, specified as a percentage of the total available CPUs.

This calculation is the host view of the CPU usage, not the guest operating system view, and is the average CPU utilization over all available virtual CPUs in the virtual machine.

For example, if a virtual machine with one virtual CPU is running on a host that has four physical CPUs and the CPU usage is 100%, the virtual machine is using one physical CPU completely. The virtual CPU usage calculation is Usage in MHz / number of virtual CPUs x core frequency.

Azure also rate limits the amount of CPU that is requested by the guest OS. Consider a scenario in which ASAv is reporting CPU usage at 40% and the hypervisor is reporting CPU usage at 90%. Now, if the ASAv needs more processing power, the CPU usage may go above 80% and the hypervisor may then report CPU usage at more than 95%. This leads to the hypervisor throttling the ASAv CPU even though the ASAv is just running a loop of lightweight commands in poll mode exhibiting idling behavior.

Hyper-V CPU Usage Reporting

In addition to viewing CPU, RAM, and disk space configuration information for available Cloud Servers, you can also view disk, I/O, and networking information. Use this information to help you decide which Cloud Server is right for your needs. You can view the available servers through either the command-line nova client or the **Cloud Control Panel** interface.

On the command line, run the following command:

```
nova flavor-list
```

All available server configurations are displayed. The list contains the following information:

- ID - The server configuration ID
- Name - The configuration name, labeled by RAM size and performance type
- Memory_MB - The amount of RAM for the configuration
- Disk - The size of the disk in GB (for general purpose Cloud Servers, the size of the system disk)
- Ephemeral - The size of the data disk
- Swap - The size of the swap space
- VCPUs - The number of virtual CPUs associated with the configuration
- RXTX_Factor - The amount of bandwidth, in Mbps, allocated to the PublicNet ports, ServiceNet ports, and isolated networks (cloud networks) attached to a server
- Is_Public - Not used

ASA Virtual and Hyper-V Graphs

There are differences in the CPU % numbers between the ASA Virtual and Hyper-V:

- The Hyper-V graph numbers are always higher than the ASA Virtual numbers.
- Hyper-V calls it %CPU usage; the ASA Virtual calls it %CPU utilization.

The terms “%CPU utilization” and “%CPU usage” mean different things:

- CPU utilization provides statistics for physical CPUs.
- CPU usage provides statistics for logical CPUs, which is based on CPU hyperthreading. But because only one vCPU is used, hyperthreading is not turned on.

Hyper-V calculates the CPU % usage as follows:

Amount of actively used virtual CPUs, specified as a percentage of the total available CPUs

This calculation is the host view of the CPU usage, not the guest operating system view, and is the average CPU utilization over all available virtual CPUs in the virtual machine.

For example, if a virtual machine with one virtual CPU is running on a host that has four physical CPUs and the CPU usage is 100%, the virtual machine is using one physical CPU completely. The virtual CPU usage calculation is Usage in MHz / number of virtual CPUs x core frequency

Note It is recommended to look into ASA Virtual reports to get an accurate CPU usage percentage.

OCI CPU Usage Reporting

You can view the CPU utilization % in OCI by using the compute instance metric: **oci_computeagent**. The CpuUtilization metric displays the activity level from the CPU and is expressed as a percentage of total time. Perform the following steps to view metric charts for a single compute instance:

Procedure

Step 1 Open the navigation menu and click **Instances** under **Compute**.

Step 2 Click an instance and click **Metrics** under **Resources**.

Step 3 Select **oci_computeagent** in the Metric namespace list.

See Compute Instance Metrics for more information.

ASAv and OCI Graphs

The OCI graph numbers are always higher than the ASAv numbers because OCI calculates the CPU % usage as the amount of actively used virtual CPUs, specified as a percentage of the total available CPUs.

This calculation is the host view of the CPU usage, not the guest operating system view, and is the average CPU utilization over all available virtual CPUs in the virtual machine.

For example, if a virtual machine with one virtual CPU is running on a host that has four physical CPUs and the CPU usage is 100%, the virtual machine is using one physical CPU completely. The virtual CPU usage calculation is Usage in MHz / number of virtual CPUs x core frequency.

Test Your Configuration

This section describes how to test connectivity for the single mode ASA or for each security context, how to ping the ASA interfaces, and how to allow hosts on one interface to ping through to hosts on another interface.

Test Basic Connectivity: Pinging Addresses

Ping is a simple command that lets you determine if a particular address is alive and responsive. The following topics explain more about the command and what types of testing you can accomplish with it.

What You Can Test Using Ping

When you ping a device, a packet is sent to the device and the device returns a reply. This process enables network devices to discover, identify, and test each other.

You can use ping to do the following tests:

- Loopback testing of two interfaces—You can initiate a ping from one interface to another on the same ASA, as an external loopback test to verify basic "up" status and operation of each interface.
- Pinging to an ASA—You can ping an interface on another ASA to verify that it is up and responding.
- Pinging through an ASA—You can ping through an intermediate ASA by pinging a device on the other side of the ASA. The packets will pass through two of the intermediate ASA's interfaces as they go in each direction. This action performs a basic test of the interfaces, operation, and response time of the intermediate unit.
- Pinging to test questionable operation of a network device—You can ping from an ASA interface to a network device that you suspect is functioning incorrectly. If the interface is configured correctly and an echo is not received, there might be problems with the device.
- Pinging to test intermediate communications—You can ping from an ASA interface to a network device that is known to be functioning correctly. If the echo is received, the correct operation of any intermediate devices and physical connectivity is confirmed.

Choosing Between ICMP and TCP Ping

The ASA includes the traditional ping, which sends ICMP Echo Request packets and gets Echo Reply packets in return. This is the standard tool and works well if all intervening network devices allow ICMP traffic. With ICMP ping, you can ping IPv4 or IPv6 addresses, or host names.

However, some networks prohibit ICMP. If this is true of your network, you can instead use TCP ping to test network connectivity. With TCP ping, the ping sends TCP SYN packets, and considers the ping a success if it receives a SYN-ACK in response. With TCP ping, you can ping IPv4 addresses or host names, but you cannot ping IPv6 addresses.

Keep in mind that a successful ICMP or TCP ping simply means that the address you are using is alive and responding to that specific type of traffic. This means that basic connectivity is working. Other policies running on a device could prevent specific types of traffic from successfully getting through a device.

Enable ICMP

By default, you can ping from a high security interface to a low security interface. You just need to enable ICMP inspection to allow returning traffic through. If you want to ping from low to high, then you need to apply an ACL to allow traffic.

When pinging an ASA interface, any ICMP rules applied to the interface must allow Echo Request and Echo Response packets. ICMP rules are optional: if you do not configure them, all ICMP traffic to an interface is allowed.

This procedure explains all of ICMP configuration you might need to complete to enable ICMP pinging of ASA interfaces, or for pinging through an ASA.

Procedure

Step 1 Ensure ICMP rules allow Echo Request/Echo Response.

ICMP rules are optional and apply to ICMP packets sent directly to an interface. If you do not apply ICMP rules, all ICMP access is allowed. In this case, no action is required.

However, if you do implement ICMP rules, ensure that you include at least the following on each interface, replacing “inside” with the name of an interface on your device.

```
ciscoasa(config)# icmp permit 0.0.0.0 0.0.0.0 echo inside
ciscoasa(config)# icmp permit 0.0.0.0 0.0.0.0 echo-reply inside
```

Step 2 Ensure access rules allow ICMP.

When pinging a host through an ASA, access rules must allow ICMP traffic to leave and return. The access rule must at least allow Echo Request/Echo Reply ICMP packets. You can add these rules as global rules.

Assuming you already have access rules applied to interfaces or applied globally, simply add these rules to the relevant ACL, for example:

```
ciscoasa(config)# access-list outside_access_in extendedpermit icmp any anyecho

ciscoasa(config)# access-list outside_access_in extendedpermit icmp any
anyecho-reply
```

Alternatively, just allow all ICMP:

```
ciscoasa(config)# access-list outside_access_in extendedpermit icmp any any
```

If you do not have access rules, you will need to also allow the other type of traffic you want, because applying any access rules to an interface adds an implicit deny, so all other traffic will be dropped. Use the **access-group** command to apply the ACL to an interface or globally.

If you are simply adding the rule for testing purposes, you can use the **no** form of the **access-list** command to remove the rule from the ACL. If the entire ACL is simply for testing purposes, use the **no access-group** command to remove the ACL from the interface.

Step 3 Enable ICMP inspection.

ICMP inspection is needed when pinging through the ASA, as opposed to pinging an interface. Inspection allows returning traffic (that is, the Echo Reply packet) to return to the host that initiated the ping, and also ensures there is one response per packet, which prevents certain types of attack.

You can simply enable ICMP inspection in the default global inspection policy.

```
ciscoasa(config)# policy-map  global_policy
ciscoasa(config-pmap)# class inspection_default
ciscoasa(config-pmap-c)# inspect icmp
```

Ping Hosts

To ping any device, you simply enter **ping** with the IP address or host name, such as **ping 10.1.1.1** or **ping www.example.com**. For TCP ping, you include the **tcp** keyword and the destination port, such as **ping tcp www.example.com 80**. That is usually the extent of any test you need to run.

Example output for a successful ping:

```
Sending 5, 100-byte ICMP Echos to out-pc, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/1 ms
```

If the ping fails, the output indicates ? for each failed attempt, and the success rate is less than 100 percent (complete failure is 0 percent):

```
Sending 5, 100-byte ICMP Echos to 10.132.80.101, timeout is 2 seconds:
?????
Success rate is 0 percent (0/5)
```

However, you can also add parameters to control some aspects of the ping. Following are your basic options:

- ICMP ping.

  ping [*if_name*] *host* [**repeat** *count*] [**timeout** *seconds*] [**data** *pattern*] [**size** *bytes*] [**validate**]

  Where:

  - *if_name* is the interface through which the source sends the ping. If you do not include a name, the data routing table is used.
  - *host* is the IPv4, IPv6, or host name of the host you are pinging.
  - **repeat** *count* is how many packets to send. The default is 5.
  - **timeout** *seconds* is the number of seconds for each packet to time out if no response occurs. The default is 2.
  - **data** *pattern* is the hexadecimal pattern to use in the packets sent. The default is 0xabcd.
  - **size** *bytes* is the length of the packet sent. The default is 100 bytes.
  - **validate** indicates that you want reply data validated.

- TCP ping.

  ping tcp [*if_name*] *host* [*port*] [**repeat** *count*] [**timeout** *seconds*] [**source** *host* [*ports*]

  Where:

  - *if_name* is the interface through which the source sends the ping. If you do not include a name, the data routing table is used.
  - *host* is the IPv4 address or host name of the destination you are pinging. You cannot use TCP ping with IPv6 addresses.
  - *port* is the TCP port on the host you are pinging.
  - **repeat** and **timeout** have the same meaning as above.
  - **source** *host port* indicates the source IP address and port for the ping. Use port 0 to get a random port.

- Interactive ping.

  ping

By entering ping without parameters, you are prompted for interface, destination, and other parameters, including extended parameters not available as keywords. Use this method if you have need for extensive control over the ping packets.

Test ASA Connectivity Systematically

If you want to do a more systematic test of ASA connectivity, you can use the following general procedure.

Before you begin

If you want to see the syslog messages mentioned in the procedure, enable logging (the **logging enable** command, or **Configuration > Device Management > Logging > Logging Setup** in ASDM).

Although unnecessary, you can also enable ICMP debug to see messages on the ASA console as you ping ASA interfaces from external devices (you will not see debug messages for pings that go through the ASA). We recommend that you only enable pinging and debugging messages during troubleshooting, as they can affect performance. The following example enables ICMP debugging, sets syslog messages to be sent to Telnet or SSH sessions and sends them to those sessions, and enables logging. Instead of using the **logging monitor debug** command, you can alternately use the **logging buffer debug** command to send log messages to a buffer, and then view them later using the **show logging**command.

```
ciscoasa(config)# debug icmp trace
ciscoasa(config)# logging monitor debug
ciscoasa(config)# terminal monitor
ciscoasa(config)# logging enable
```

With this configuration, you would see something like the following for a successful ping from an external host (209.165.201.2) to the ASA outside interface (209.165.201.1):

```
ciscoasa(config)# debug icmp trace
Inbound ICMP echo reply (len 32 id 1 seq 256) 209.165.201.1 > 209.165.201.2
Outbound ICMP echo request (len 32 id 1 seq 512) 209.165.201.2 > 209.165.201.1
Inbound ICMP echo reply (len 32 id 1 seq 512) 209.165.201.1 > 209.165.201.2
Outbound ICMP echo request (len 32 id 1 seq 768) 209.165.201.2 > 209.165.201.1
Inbound ICMP echo reply (len 32 id 1 seq 768) 209.165.201.1 > 209.165.201.2
Outbound ICMP echo request (len 32 id 1 seq 1024) 209.165.201.2 > 209.165.201.1
Inbound ICMP echo reply (len 32 id 1 seq 1024) 209.165.201.1 > 209.165.201.2
```

The output shows the ICMP packet length (32 bytes), the ICMP packet identifier (1), and the ICMP sequence number (the ICMP sequence number starts at 0, and is incremented each time that a request is sent).

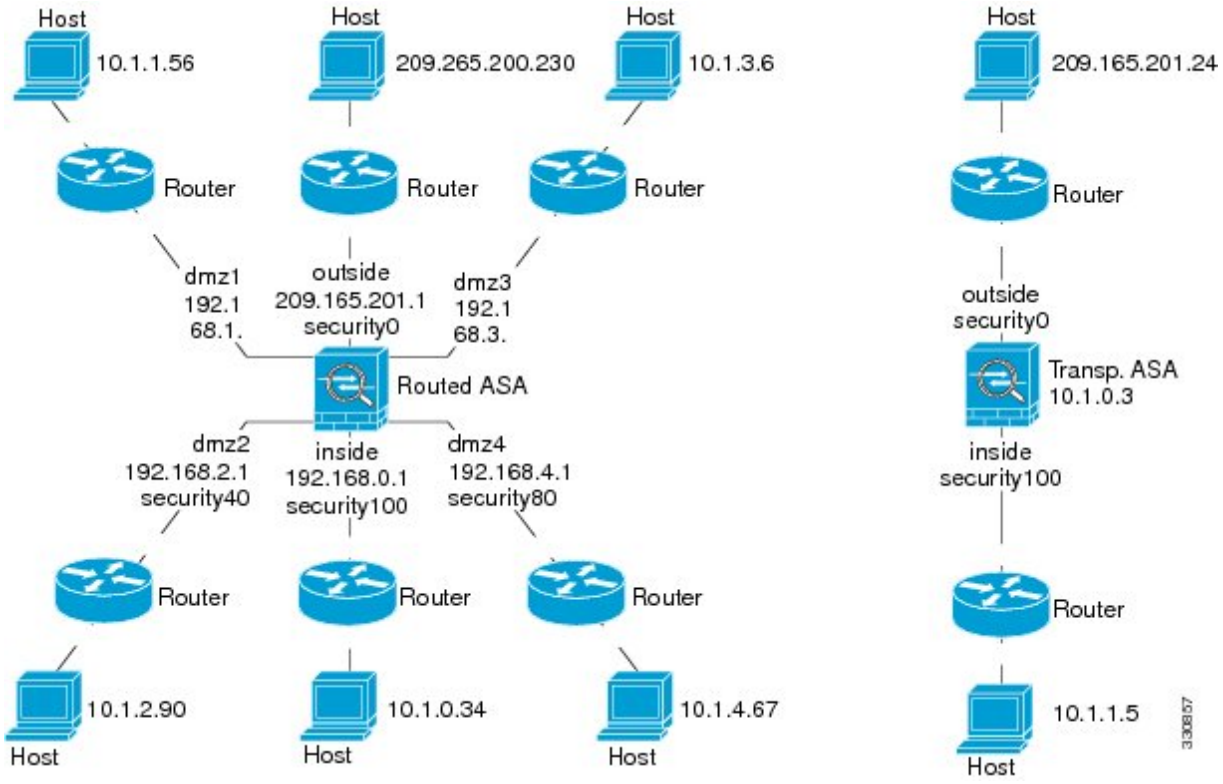
When you are finished testing, disable debugging. Leaving the configuration in place can pose performance and security risks. If you enabled logging just for testing, you can disable it also.

```
ciscoasa(config)# no debug icmp trace
ciscoasa(config)# no logging monitor debug
ciscoasa(config)# no terminal monitor
ciscoasa(config)# no logging enable
```

Procedure

Step 1 Draw a diagram of your single-mode ASA or security context that shows the interface names, security levels, and IP addresses. The diagram should also include any directly connected routers and a host on the other side of the router from which you will ping the ASA.

Figure 69: Network Diagram with Interfaces, Routers, and Hosts



Step 2 Ping each ASA interface from the directly connected routers. For transparent mode, ping the BVI IP address. This test ensures that the ASA interfaces are active and that the interface configuration is correct.

A ping might fail if the ASA interface is not active, the interface configuration is incorrect, or if a switch between the ASA and a router is down (see the following figure). In this case, no debugging messages or syslog messages appear, because the packet never reaches the ASA.

Figure 70: Ping Failure at the ASA Interface

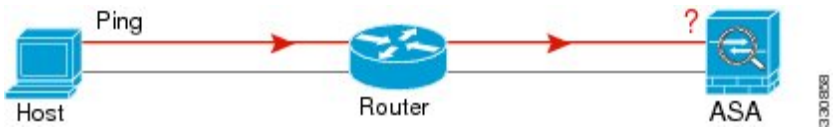
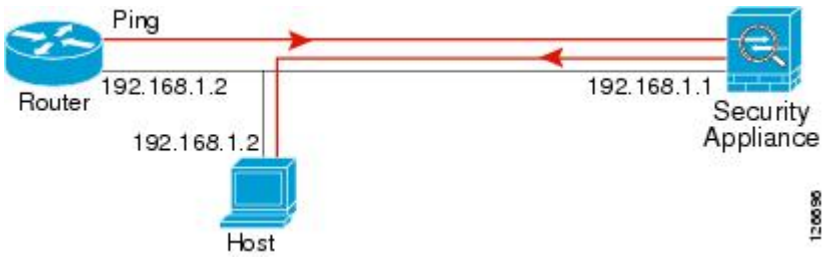


Figure 71: Ping Failure Because of IP Addressing Problems



If the ping reply does not return to the router, then a switch loop or redundant IP addresses might exist (see the following figure).

Step 3 Ping each ASA interface from a remote host. For transparent mode, ping the BVI IP address. This test checks whether the directly connected router can route the packet between the host and the ASA, and whether the ASA can correctly route the packet back to the host.

A ping might fail if the ASA does not have a return route to the host through the intermediate router (see the following figure). In this case, the debugging messages show that the ping was successful, but syslog message 110001 appears, indicating a routing failure has occurred.

Figure 72: Ping Failure Because the ASA Has No Return Route



Step 4 Ping from an ASA interface to a network device that you know is functioning correctly.

- If the ping is not received, a problem with the transmitting hardware or interface configuration may exist.
- If the ASA interface is configured correctly and it does not receive an echo reply from the "known good" device, problems with the interface hardware receiving function may exist. If a different interface with "known good" receiving capability can receive an echo after pinging the same "known good" device, the hardware receiving problem of the first interface is confirmed.

Step 5 Ping from the host or router through the source interface to another host or router on another interface. Repeat this step for as many interface pairs as you want to check. If you use NAT, this test shows that NAT is operating correctly.

If the ping succeeds, a syslog message appears to confirm the address translation for routed mode (305009 or 305011) and that an ICMP connection was established (302020). You can also enter either the **show xlate** or **show conns** command to view this information.

The ping might fail because NAT is not configured correctly. In this case, a syslog message appears, showing that the NAT failed (305005 or 305006). If the ping is from an outside host to an inside host, and you do not have a static translation, you get message 106010.

Figure 73: Ping Failure Because the ASA is Not Translating Addresses



Trace Routes to Hosts

If you are having problems sending traffic to an IP address, you can trace the route to the host to determine if there is a problem on the network path.

Procedure

Step 1 Make the ASA Visible on Trace Routes, on page 1289.

Step 2 Determine Packet Routes, on page 1291.

Make the ASA Visible on Trace Routes

By default, the ASA does not appear on traceroutes as a hop. To make it appear, you need to decrement the time-to-live on packets that pass through the ASA, and increase the rate limit on ICMP unreachable messages.

Procedure

Step 1 Create an L3/L4 class map to identify the traffic for which you want to customize connection settings.

class-map *name*

match *parameter*

Example:

```
ciscoasa(config)# class-map CONNS
ciscoasa(config-cmap)# match any
```

For information on matching statements, see the Service Policy chapter in the firewall configuration guide.

Step 2 Add or edit a policy map that sets the actions to take with the class map traffic, and identify the class map.

policy-map *name* **class** *name*

Example:

```
ciscoasa(config)# policy-map global_policy
```

```
ciscoasa(config-pmap)# class CONNS
```

In the default configuration, the global_policy policy map is assigned globally to all interfaces. If you want to edit the global_policy, enter global_policy as the policy name. For the class map, specify the class you created earlier in this procedure.

Step 3 Decrement time-to-live (TTL) on packets that match the class.

```
set connection decrement-ttl
```

Step 4 If you are editing an existing service policy (such as the default global policy called global_policy), you can skip this step. Otherwise, activate the policy map on one or more interfaces.

service-policy *policymap_name* {**global** | **interface** *interface_name* }

Example:

```
ciscoasa(config)# service-policy global_policy global
```

The **global** keyword applies the policy map to all interfaces, and **interface** applies the policy to one interface. Only one global policy is allowed. You can override the global policy on an interface by applying a service policy to that interface. You can only apply one policy map to each interface.

Step 5 Increase the rate limit on ICMP Unreachable messages so that the ASA will appear on trace route output.

icmp unreachable rate-limit *rate* **burst-size** *size*

Example:

```
ciscoasa(config)# icmp unreachable rate-limit 50 burst-size 1
```

The rate limit can be 1-100, with 1 being the default. The burst size is meaningless, but must be 1-10.

Example

The following example decrements TTL for all traffic globally and increase the ICMP unreachable limit to 50.

```
ciscoasa(config)# class-map global-policy
ciscoasa(config-cmap)# match any
ciscoasa(config-cmap)# exit
ciscoasa(config)# policy-map global_policy
ciscoasa(config-pmap)# class global-policy
ciscoasa(config-pmap-c)# set connection decrement-ttl
ciscoasa(config-pmap-c)# exit
ciscoasa(config)# icmp unreachable rate-limit 50 burst-size 6
```

Determine Packet Routes

Use Traceroute to help you to determine the route that packets will take to their destination. A traceroute works by sending UDP packets or ICMPv6 echo to a destination on an invalid port. Because the port is not valid, the routers along the way to the destination respond with an ICMP or ICMPv6 Time Exceeded Message, and report that error to the ASA.

The traceroute shows the result of each probe sent. Every line of output corresponds to a TTL value in increasing order. The following table explains the output symbols.

Output Symbol	Description
*	No response was received for the probe within the timeout period.
U	No route to the destination.
nn msec	For each node, the round-trip time (in milliseconds) for the specified number of probes.
!N.	ICMP network unreachable. For ICMPv6, address is out of scope.
!H	ICMP host unreachable.
!P	ICMP unreachable. For ICMPv6, port not reachable.
!A	ICMP administratively prohibited.
?	Unknown ICMP error.

Procedure

Trace the route to a destination:

traceroute [*destination_ip* | *hostname*} [**source** {*source_ip* | *source-interface*}] [**numeric**] [**timeout** *timeout_value*] [**probe** *probe_num*] [**ttl** *min_ttl max_ttl*] [**port** *port_value*] [**use-icmp**]

Example:

```
ciscoasa# traceroute 209.165.200.225

Type escape sequence to abort.
Tracing the route to 209.165.200.225

1 10.83.194.1 0 msec 10 msec 0 msec
2 10.83.193.65 0 msec 0 msec 0 msec
3 10.88.193.101 0 msec 10 msec 0 msec
4 10.88.193.97 0 msec 0 msec 10 msec
5 10.88.239.9 0 msec 10 msec 0 msec
6 10.88.238.65 10 msec 10 msec 0 msec
7 172.16.7.221 70 msec 70 msec 80 msec
8 209.165.200.225 70 msec 70 msec 70 msec

ciscoasa# traceroute 2002::130

Type escape sequence to abort.
Tracing the route to 2002::130
```

```
1  5000::2 0 msec 0 msec 0 msec
2  2002::130 10 msec 0 msec 0 msec
```

Normally, you simply include the destination IP address or hostname, such as **traceroute www.example.com**. However, you can adjust the characteristics of the trace if desired:

- **source** {*source_ip* | *source-interface*}—Specifies the interface to use as the source of the trace. You can specify the interface by name or by IP address. For IPv6, you cannot specify the source interface; you can only specify the source IP address. An IPv6 address is valid only if you enabled IPv6 on an ASA interface. In transparent mode, you must use the management address.
- **numeric**—Indicates that only the IP addresses should be shown in the trace route. Without this keyword, the trace route does DNS lookups for addresses and includes DNS names, assuming that you configure DNS.
- **timeout** *timeout_value*—How long to wait for a response before timing out. The default is 3 seconds.
- **probe** *probe_num*—How many probes to send at each TTL level. The default is 3.
- **ttl** *min_ttl max_ttl*—The minimum and maximum time-to-live values for the probes. The minimum default is one, but you can set it to a higher value to suppress the display of known hops. The maximum default is 30. The traceroute terminates when the packet reaches the destination or when the maximum value is reached.
- **port** *port_value*—The UDP port to use. The default is 33434.
- **use-icmp**—Send ICMP packets instead of UDP packets for probes.

Using the Packet Tracer to Test Policy Configuration

You can test your policy configuration by modeling a packet based on source and destination addressing and protocol characteristics. The trace does policy lookup to test access rules, NAT, routing, and so forth, to see if the packet would be permitted or denied.

By testing packets this way, you can see the results of your policies and test whether the types of traffic you want to allow or deny are handled as desired. Besides verifying your configuration, you can use the tracer to debug unexpected behavior, such as packets being denied when they should be allowed.

Procedure

Step 1 The command is complicated, so we divided it into parts. Start by choosing the interface and protocol for the trace:

packet-tracer input *ifc_name* [**vlan-id***vlan_id*] {**icmp** | **tcp** | **udp** | **rawip** | **sctp**} [**inline-tag** *tag*] ...

Where:

- **input** *ifc_name*—The name of the interface from which to start the trace. For a bridge group, specify the bridge group member interface name.

- **vlan-id** *vlan_id*—(Optional.) The virtual LAN, where packet tracer enters a parent interface, which is later redirected to a sub-interface. VLAN identity is available only when the input interface is not a sub-interface. Valid values range from 1 - 4096.
- **icmp**, **tcp**, **udp**, **rawip**, **sctp**—The protocol to use. “rawip” is raw IP, that is, IP packets that are not TCP/UDP.
- **inline-tag** *tag*—(Optional.) The security group tag value embedded in the Layer 2 CMD header. Valid values range from 0 - 65533.

Step 2 Next, type in the source address and protocol criteria.

...{*src_ip* | **user** *username* | **security-group** {**name** *name* | **tag** *tag*} | **fqdn** *fqdn-string*}*...*

Where:

- *src_ip*—The source IPv4 or IPv6 address for the packet trace.
- **user** *username*—The user identity in the format of domain\user. The most recently mapped address for the user (if any) is used in the trace.
- **security-group** {**name** *name* | **tag** *tag*}—The source security group based on the IP-SGT lookup for Trustsec. You can specify a security group name or a tag number.
- **fqdn** *fqdn-string*—The fully qualified domain name of the source host, IPv4 only.

Step 3 Next, type in the protocol characteristics.

- ICMP—Enter the ICMP type (1-255), ICMP code (0-255), and optionally, the ICMP identifier. You must use numbers for each variable, for example, 8 for echo.

  type code... [*ident*]...
- TCP/UDP/SCTP—Enter the source port number.

  ...src_port ...
- Raw IP—Enter the protocol number, 0-255.

  ... protocol ...

Step 4 Finally, type in the destination address criteria, destination port for TCP/UDP traces, and optional keywords, and press **Enter**.

...dmac {*dst_ip* | **security-group** {**name** *name* | **tag** *tag*} | **fqdn** *fqdn-string*} *dst_port* [**detailed**] [**xml**]

Where:

- *dst_ip*—The destination IPv4 or IPv6 address for the packet trace.
- **security-group** {**name** *name* **| tag** *tag*}—The destination security group based on the IP-SGT lookup for Trustsec. You can specify a security group name or a tag number.
- **fqdn** *fqdn-string*—The fully qualified domain name of the destination host, IPv4 only.
- *dst_port*—The destination port for TCP/UDP/SCTP traces. Do not include this value for ICMP or raw IP traces.
- *dmac*—(Transparent mode) The Destination MAC address.

- **detailed**—Provides detailed trace results information in addition to the normal output.
- **xml**—Displays the trace results in XML format.

Step 5 Type in the **persist** option for packet tracer to debug packets across cluster units.

- You can allow simulated packets to egress the ASA by using the **transmit** option.
- To skip security checks like ACL, VPN filters, IPsec spoof, and uRPF, use the **bypass-checks** option.
- Using the **decrypted** option, you can inject a decrypted packet in a VPN tunnel and also simulate a packet that comes across a VPN tunnel.

Step 6 Type in the **id** and **origin** for tracking a specific packet in the cluster units.

- **id**—The identity number assigned by the unit that starts the trace.
- **origin**—Indicates the cluster unit that commences the trace.

Example

The following example traces a TCP packet for the HTTP port from 10.100.10.10 to 10.100.11.11. The result indicates that the packet will be dropped by the implicit deny access rule.

```
ciscoasa(config)# packet-tracer input outside tcp 10.100.10.10 80 10.100.11.11
 80

Phase: 1
Type: ROUTE-LOOKUP
Subtype: Resolve Egress Interface
Result: ALLOW
Config:
Additional Information:
found next-hop 10.86.116.1 using egress ifc  outside

Phase: 2
Type: ACCESS-LIST
Subtype:
Result: DROP
Config:
Implicit Rule
Additional Information:

Result:
input-interface: outside
input-status: up
input-line-status: up
output-interface: NP Identity Ifc
output-status: up
output-line-status: up
Action: drop
Drop-reason: (acl-drop) Flow is denied by configured rule
```

Monitoring Connections

To view current connections with information about source, destination, protocol, and so forth, use the **show conn all detail** command.

History for Testing and Troubleshooting

Feature Name	Platform Releases	Description
IPv6 support for traceroute	9.7(1)	The **traceroute** command was modified to accep address. We modified the following command: **tracerout**
Support for the packet tracer for bridge group member interfaces	9.7(1)	You can now use the packet tracer for bridge gro interfaces. We added two new options to the **packet-tracer vlan-id** and **dmac**
Manually start and stop packet captures	9.7(1)	You can now manually stop and start the capture Added/Modified commands: **capture stop**
Enhanced packet tracer and packet capture capabilities	9.9(1)	The packet tracer has been enhanced with the follo • Trace a packet when it passes between clust • Allow simulated packets to egress the ASA • Bypass security checks for a similated pack • Treat a simulated packet as an IPsec/SSL dec The packet capture has been enhanced with the f features: • Capture packets after they are decrypted. • Capture traces and retain them in the persist New or modified commands: **cluster exec captu include-decrypted, cluster exec capture test tr cluster exec clear packet-tracer, cluster exec s packet-tracer id, cluster exec show packet-tra packet-tracer persist, packet-tracer transmit, p decrypted, packet-tracer bypass-checks**

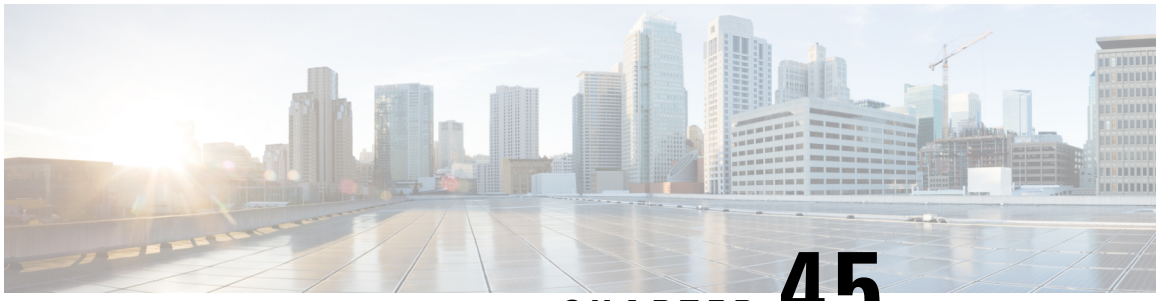
Feature Name	Platform Releases	Description
Packet capture support for matching IPv6 traffic without using an ACL	9.10(1)	If you use the **match** keyword for the **capture** comm **any** keyword only matches IPv4 traffic. You can no **any4** and **any6** keywords to capture either IPv4 or IP The **any** keyword continues to match only IPv4 traf New/Modified commands: **capture match**
New **debug telemetry** command for Forepower 9300/4100.	9.14(1)	If you use the **debug telemetry** command, debug m related to telemetry are displayed. The debugs help t the cause for errors when generating the telemetry re New/Modified commands: **[no] debug telemetry**,**sh telemetry**



PART VIII

Monitoring

- Logging, on page 1299
- SNMP, on page 1327
- Cisco Success Network and Telemetry Data, on page 1365
- Alarms for the Cisco ISA 3000, on page 1375
- Anonymous Reporting and Smart Call Home, on page 1383



CHAPTER 45

Logging

This chapter describes how to log system messages and use them for troubleshooting.

- About Logging, on page 1299
- Guidelines for logging, on page 1306
- Configure Logging, on page 1308
- Monitoring the Logs, on page 1323
- Examples for Logging, on page 1324
- History for Logging, on page 1325

About Logging

System logging is a method of collecting messages from devices to a server running a syslog daemon. Logging to a central syslog server helps in aggregation of logs and alerts. Cisco devices can send their log messages to a UNIX-style syslog service. A syslog service accepts messages and stores them in files, or prints them according to a simple configuration file. This form of logging provides protected long-term storage for logs. Logs are useful both in routine troubleshooting and in incident handling.

The ASA system logs provide you with information for monitoring and troubleshooting the ASA. With the logging feature, you can do the following:

- Specify which syslog messages should be logged.
- Disable or change the severity level of a syslog message.
- Specify one or more locations where syslog messages should be sent, including:
  - An internal buffer
  - One or more syslog servers
  - ASDM
  - An SNMP management station
  - Specified e-mail addresses
  - Console
  - Telnet and SSH sessions.

- Configure and manage syslog messages in groups, such as by severity level or class of message.
- Specify whether or not a rate-limit is applied to syslog generation.
- Specify what happens to the contents of the internal log buffer when it becomes full: overwrite the buffer, send the buffer contents to an FTP server, or save the contents to internal flash memory.
- Filter syslog messages by locations, severity level, class, or a custom message list.

Logging in Multiple Context Mode

Each security context includes its own logging configuration and generates its own messages. If you log in to the system or admin context, and then change to another context, messages you view in your session are only those messages that are related to the current context.

Syslog messages that are generated in the system execution space, including failover messages, are viewed in the admin context along with messages generated in the admin context. You cannot configure logging or view any logging information in the system execution space.

You can configure the ASA to include the context name with each message, which helps you differentiate context messages that are sent to a single syslog server. This feature also helps you to determine which messages are from the admin context and which are from the system; messages that originate in the system execution space use a device ID of **system**, and messages that originate in the admin context use the name of the admin context as the device ID.

Syslog Message Analysis

The following are some examples of the type of information you can obtain from a review of various syslog messages:

- Connections that are allowed by ASA security policies. These messages help you spot holes that remain open in your security policies.
- Connections that are denied by ASA security policies. These messages show what types of activity are being directed toward your secured inside network.
- Using the ACE deny rate logging feature shows attacks that are occurring on your ASA.
- IDS activity messages can show attacks that have occurred.
- User authentication and command usage provide an audit trail of security policy changes.
- Bandwidth usage messages show each connection that was built and torn down as well as the duration and traffic volume used.
- Protocol usage messages show the protocols and port numbers used for each connection.
- Address translation audit trail messages record NAT or PAT connections being built or torn down, which are useful if you receive a report of malicious activity coming from inside your network to the outside world.

Syslog Message Format

Syslog messages are structured as follows:

```
[<PRI>]: [Timestamp] [Device-ID] : %ASA-Level-Message_number: Message_text
```

Field descriptions are as follows:

<PRI>	Priority value. When the logging EMBLEM is enabled, this value is displayed in the syslog message. Logging EMBLEM is compatible with UDP and not with TCP.
Timestamp	Date and time of the event is displayed. When logging of timestamps is enabled, and if the timestamp is configured to be in the RFC 5424 format, all timestamp in syslog messages display the time in UTC, as indicated by the RFC 5424 standard.
Device-ID	The device identifier string that was configured while enabling the logging device-id option through the user interface. If enabled, the device ID does not appear in EMBLEM-formatted syslog messages.
ASA	The syslog message facility code for messages that are generated by the ASA. This value is always `ASA`.
Level	0 through 7. The level reflects the severity of the condition described by the syslog message—the lower the number, the more severe the condition.
Message_number	A unique six-digit number that identifies the syslog message.
Message_text	A text string that describes the condition. This portion of the syslog message sometimes includes IP addresses, port numbers, or usernames.

All syslog messages that are generated by the device are documented in the Cisco Secure Firewall ASA Series Syslog Messages guide.

The EMBLEM syslog format is a Cisco-specific convention that is built upon the RFC 3164 and RFC 5424 standards. Hence, when EMBLEM is enabled, the syslog message prints colon (:) after <PRI> field.

Example of a syslog message with logging EMBLEM, logging timestamp rfc5424, and device-id enabled. Note the colon (:) after the <PRI> field (*<166>*).

```
<166>:2018-06-27T12:17:46Z: %ASA-6-110002: Failed to locate egress interface for protocol
from src interface :src IP/src port to dest IP/dest port
```

Example of a syslog message with logging timestamp rfc5424 and device-id enabled. No colon (:) is present before the timestamp.

```
2018-06-27T12:17:46Z asa : %ASA-6-110002: Failed to locate egress interface for protocol
from src interface :src IP/src port to dest IP/dest port
```

Syslog message severity levels

The syslog message severity levels define the importance of system messages. You can assign custom colors to each severity level to make them easier to distinguish in the ASDM log viewers. To configure syslog message color settings, either choose the **Tools > Preferences > Syslog** tab or, in the log viewer itself, click **Color Settings** on the toolbar.

Table 58: Syslog message severity levels

Level Number	Severity Level	Description
0	**emergencies**	You cannot use the system.

Level Number	Severity Level	Description
1	**alert**	You need to take immediate action.
2	**critical**	The system has critical conditions to be addressed.
3	**error**	Error conditions.
4	**warning**	The system has warnings you should review.
5	**notification**	The system reports significant events you should monitor.
6	**informational**	Informational messages only.
7	**debugging**	The system generates only debugging messages Log at this level only temporarily, when debugging issues. This log level can generate many messages, which might affect system performance.



Note ASA does not generate syslog messages with a severity level of zero (emergencies).

Syslog message filters

Syslog message filter criteria

You can direct syslog messages to an output destination according to these criteria:

- Syslog message ID number
- Syslog message severity level
- Syslog message class (equivalent to a functional area)

You can create a message list with specific criteria, which you can then specify as the output destination is set. Alternatively, you can configure the ASA to send a particular message class to each type of output destination independently of the message list.

Syslog message classes and associated message ID numbers

You can use syslog message classes in two ways:

- Specify an output location for an entire category of syslog messages. Use the **logging class** command.
- Create a message list that specifies the message class. Use the **logging list** command.

Use the syslog message class to categorize syslog messages by type, such as features or functions of the device. For example, use the RIP class for RIP routing.

Syslog messages in the same class have the same three initial digits in their message ID numbers. For example, the syslog messages for the VPN client feature use ID numbers 611101 to 611323.

Most ISAKMP syslog messages include prepended objects that identify the tunnel. When available, these objects appear before the descriptive text of a syslog message. If the object is unknown when the syslog message is generated, the heading and value combination is omitted.

The objects are prefixed in this manner:

Group = *groupname*, Username = *user*, IP = *IP_address*

The term 'group' here refers to the tunnel-group configuration. The username is from the local database or AAA server. The IP address is the public IP address of the remote access client or Layer 2 peer.

See the message classes and the range of message ID numbers in each class in this table.

Table 59: Syslog message classes and associated message ID numbers

Class	Definition	Syslog Message ID Numbers
auth	User Authentication	109, 113
—	Access Lists	106
—	Application Firewall	415
—	Botnet Traffic Filtering	338
bridge	Transparent Firewall	110, 220
ca	PKI Certification Authority	717
citrix	Citrix Client	723
—	Clustering	747
—	Card Management	323
config	Command Interface	111, 112, 208, 308
csd	Secure Desktop	724
cts	Cisco TrustSec	776
dap	Dynamic Access Policies	734
EAP, eapoudp	EAP or EAPoUDP for Network Admission Control	333, 334
EIGRP	EIGRP Routing	336
email	E-mail Proxy	719
—	Environment Monitoring	735
ha	Failover	101, 102, 103, 104, 105, 210, 311, 709
—	Identity-based Firewall	746
ids	Intrusion Detection System	400, 733

Class	Definition	Syslog Message ID Numbers
—	IKEv2 Toolkit	750, 751, 752
IP	IP Stack	209, 215, 313, 317, 408
ipaa	IP Address Assignment	735
ips	Intrusion Protection System	400, 401, 420
—	IPv6	325
—	Licensing	444
MDM-proxy	MDM Proxy	802
NAC	Network Admission Control	731, 732
nacpolicy	NAC Policy	731
nacsettings	NAC Settings to apply NAC Policy	732
—	NAT and PAT	305
—	Network Access Point	713
NP	Network Processor	319
—	NP SSL	725
OSPF	OSPF Routing	318, 409, 503, 613
—	Password Encryption	742
—	Phone Proxy	337
RIP	RIP Routing	107, 312
rm	Resource Manager	321
—	Smart Call Home	120
session	User Session	106, 108, 201, 202, 204, 302, 303, 304, 305, 314, 405, 406, 407, 500, 502, 607, 608, 609, 616, 620, 703, 710
SNMP	SNMP	212
—	ScanSafe	775
SSL	SSL Stack	725
svc	SSL VPN Client	722

Class	Definition	Syslog Message ID Numbers
sys	System	199, 211, 214, 216, 306, 307, 315, 414, 604, 605, 606, 610, 612, 614, 615,701, 711, 741
—	Threat Detection	733
tag-switching	Service Tag Switching	779
transactional-rule-engine-tre	Transactional Rule Engine	780
uc-IMS	UC-IMS	339
vm	VLAN Mapping	730
vpdn	PPTP and L2TP Sessions	213, 403, 603
VPN	IKE and IPsec	316, 320, 402, 404, 501, 602, 702, 713, 714, 715
vpnc	VPN Client	611
vpnfo	VPN Failover	720
vpnlb	VPN Load Balancing	718
—	VXLAN	778
webfo	WebVPN Failover	721
webvpn	WebVPN and AnyConnect Client	716

Custom Message Lists

Creating a custom message list is a flexible way to exercise control over which syslog messages are sent to which output destination. In a custom syslog message list, you specify groups of syslog messages using any or all of the following criteria:

- Severity level
- Message IDs
- Ranges of syslog message IDs
- Message class.

For example, you can use message lists to do the following:

- Select syslog messages with the severity levels of 1 and 2 and send them to one or more e-mail addresses.
- Select all syslog messages associated with a message class (such as ha) and save them to the internal buffer.

A message list can include multiple criteria for selecting messages. However, you must add each message selection criterion with a new command entry. It is possible to create a message list that includes overlapping message selection criteria. If two criteria in a message list select the same message, the message is logged only once.

Clustering

Syslog messages are an invaluable tool for accounting, monitoring, and troubleshooting in a clustering environment. Each ASA unit in the cluster (up to eight units are allowed) generates syslog messages independently; certain **logging** commands then enable you to control header fields, which include a time stamp and device ID. The syslog server uses the device ID to identify the syslog generator. You can use the **logging device-id** command to generate syslog messages with identical or different device IDs to make messages appear to come from the same or different units in the cluster.

Guidelines for logging

Restrictions for IPv6 logging

- IPv6 is supported. Syslog message can be sent using TCP or UDP.
- Ensure that the interface configured for sending syslogs is enabled, IPv6 capable, and the syslog server is reachable through the designated interface.
- You cannot use IPv6 for secure logging.

Syslog server configuration guidelines

- The syslog server must run a server program called syslogd. Windows includes a syslog server as part of its operating system.
- The syslog server operates based on the syslog-ng process of the firewall system. Do not use external configuration files, such as the *scwx.conf* file from SecureWorks. Because these files are not compatible with the device, using them causes parsing errors and the syslog-ng process fails.
- You can configure up to 16 syslog servers. However, in multiple context mode, the limitation is four servers per context.
- Assign only one list or class to each syslog server and location.
- The syslog server should be reachable through the ASA. Configure the device to deny ICMP unreachable messages on the interface through which the syslog server is reachable, and to send syslogs to the same server. Enable logging for all severity levels. To prevent the syslog server from crashing, suppress the generation of syslogs 313001, 313004, and 313005.
- To view logs generated by the ASA, you must specify a logging output destination. If you enable logging without specifying a logging output destination, the ASA generates messages but does not save them to a location from which you can view them. You must specify each different logging output destination separately. For example, to designate more than one syslog server as an output destination, enter a new command for each syslog server.
- The Firewall Threat Defense syslog supports one-way TLS for secure logging. It does not support client certificate authentication, trustpoint configuration for syslog, or mutual TLS (X.509 certificate authentication).

- The server certificate received from a Syslog Server must contain "ServAuth" in the Extended Key Usage field. This check will be done on non self-signed certificates only, self-signed certificates do not provide any value in this field.

Interfaces for syslog configuration

- Determining the egress interface for the syslog:
  - If the specified management-only interface has management-access enabled, the management center will perform route table lookups and determine the egress interface (could be data or management) based on best routing logic.
  - If you configure a management-only interface as logging host, that does not have management-access enabled, the management center will use the interface regardless of routing table entries.

    Thus, for the management center to always uses a dedicated management path for syslog traffic, configure the management interface without management-access, and then specify the interface in the logging host:

    ```
    interface <management-interface>
    management-only ----->Do not include management-access
    logging host <management-interface> <syslog-server-ip>
    ```

- When the syslog rate exceeds 50,000 messages per second, ensure that a data interface is used as egress interface rather than a management interface.
- When you use a custom message list to match only access list hits, the access list logs are not generated for access lists that have had their logging severity level increased to debugging (level 7). The default logging severity level is set to 6 for the **logging list** command. This default behavior is by design. If you explicitly change the logging severity level of the access list configuration to debugging, you must also change the logging configuration itself.

  This is sample output from the **show running-config logging** command that does not include access list hits, because their logging severity level has been changed to debugging:

  ```
  ciscoasa# show running-config logging
  logging enable
  logging timestamp
  logging list test message 106100
  logging buffered test
  ```

  This is sample output from the **show running-config logging** command that does include access list hits:

  ```
  ciscoasa# show running-config logging
  logging enable
  logging timestamp
  logging buffered debugging
  ```

  In this case, the access list configuration does not change, and the number of access list hits appears, as shown in this example:

  ```
  ciscoasa(config)# access-list global line 1 extended
  permit icmp any host 4.2.2.2 log debugging interval 1 (hitcnt=7) 0xf36b5386
  ciscoasa(config)# access-list global line 2 extended
  permit tcp host 10.1.1.2 any eq www log informational interval 1 (hitcnt=18) 0xe7e7c3b8
  ```

```
ciscoasa(config)# access-list global line 3 extended
permit ip any any (hitcnt=543) 0x25f9e609
```

TCP and UDP connections for syslog

- Sending syslogs over TCP is not supported on a standby device.
- If you use TCP as the transport protocol, the system opens 4 connections to the syslog server to ensure that messages are not lost. If you collect messages from many devices and the combined connection overhead is too much for the server, use UDP instead.
- The number of UDP connections for syslog is directly related to the number of CPUs on the hardware platform and the number of syslog servers you configure. At any point in time, there can be as many UDP syslog connections as there are CPUs times the number of configured syslog servers. The global UDP connection idle timeout applies to these sessions, and the default duration is two minutes. You can adjust this setting to close sessions more quickly, but the timeout applies to all UDP connections, not just syslog. This is the intended behavior.
- When the ASA sends syslogs via TCP, the connection takes about one minute to initiate after the syslogd service restarts.
- When the TCP logging host goes down, it takes approximately six minutes to change its connection status from *Connected* to *Not connected*. Logging relies on TCP to detect the channel state; until then, logging sends the logs through the channel. During this time, when you execute the **show log**, the output would display the TCP logging host as connected. Once the TCP channel is closed, the TCP logging host state is updated to *Not connected*.

Configure Logging

This section describes how to configure logging.

Enable Logging

To enable logging, perform the following steps:

Procedure

Enable logging.

logging enable

Example:

```
ciscoasa(config)# logging enable
```

Configure an Output Destination

To optimize syslog message usage for troubleshooting and performance monitoring, we recommend that you specify one or more locations where syslog messages should be sent, including an internal log buffer, one or more external syslog servers, ASDM, an SNMP management station, the console port, specified e-mail addresses, or Telnet and SSH sessions.

When you configure syslog logging on an interface with management-only access enabled, the dataplane related logs (syslog IDs 302015, 302014, 106023, and 304001) are dropped and does not reach the syslog server. The syslog messages are dropped because the datapath routing table does not have the management interface routing. Hence, ensure the interface that you are configuring has management-only access disabled.

Send Syslog Messages to an External Syslog Server

You can archive messages according to the available disk space on the external syslog server, and manipulate logging data after it is saved. For example, you could specify actions to be executed when certain types of syslog messages are logged, extract data from the log and save the records to another file for reporting, or track statistics using a site-specific script.

To send syslog messages to an external syslog server, perform the following steps:

Procedure

Step 1 Configure the ASA to send messages to syslog servers.

You can configure the ASA to send messages to IPv4 or IPv6 syslog servers.

logging host *interface_name syslog_ip* [**tcp**[/*port*] | **udp** [/*port*] [**format emblem**]]

Example:

```
ciscoasa(config)# logging host dmz1 192.168.1.5 udp/1026
ciscoasa(config)# logging host dmz1 2002::1:1 udp/2020
```

The **format emblem** keyword enables EMBLEM format logging for the syslog server with UDP only. The *interface_name* argument specifies the interface through which you access the syslog server. The *syslog_ip* argument specifies the IP address of the syslog server. The **tcp**[/*port*] or **udp**[/*port*] keyword-argument pair specify that the ASA should use TCP or UDP to send syslog messages to the syslog server.

You can configure the ASA to send data to a syslog server using either UDP or TCP, but not both. The default protocol is UDP if you do not specify a protocol.

Warning
If you specify TCP, when the ASA discovers syslog server failures, for security reasons, new connections through the ASA are blocked. To allow new connections regardless of connectivity to a TCP syslog server, see Step 3.

If you specify UDP, the ASA continues to allow new connections whether or not the syslog server is operational. Valid port values for either protocol are 1025 through 65535. The default UDP port is 514. The default TCP port is 1470.

Step 2 Specify which syslog messages should be sent to the syslog server.

logging trap {*severity_level* | *message_list*}

Example:

```
ciscoasa(config)# logging trap errors
```

You can specify the severity level number (1 through 7) or name. For example, if you set the severity level to 3, then the ASA sends syslog messages for severity levels 3, 2, and 1. You can specify a custom message list that identifies the syslog messages to send to the syslog server.

Step 3 (Optional) Disable the feature to block new connections when a TCP-connected syslog server is down.

logging permit-hostdown

Example:

```
ciscoasa(config)# logging permit-hostdown
```

When the ASA is configured to send syslog messages to a TCP-based syslog server, and if either the syslog server is down or the log queue is full, then new connections to ASA are blocked. New connections are allowed again after the syslog server is back up and the log queue is no longer full. Using this command, you can permit new connections even if the syslog server is not operational.

Step 4 (Optional) Set the logging facility to a value other than 20, which is what most UNIX systems expect.

logging facility *number*

Example:

```
ciscoasa(config)# logging facility 21
```

Enable Secure Logging

Procedure

Enable secure logging by specifying the **secure** keyword in the logging host command. Also, optionally enter the **reference-identity**.

logging host *interface_name syslog_ip* [**tcp**/*port* | **udp**/*port*] [**format emblem**] [**secure**[**reference-identity** *reference_identity_name*]]

Where:

- **logging host** *interface_name syslog_ip* specifies the interface on which the syslog server resides and the IP address of the syslog server.
- [**tcp**/*port* | **udp**/*port*] specifies the port (TCP or UDP) that the syslog server listens to for syslog messages. The **tcp** keyword specifies that the ASA should use TCP to send syslog messages to the syslog server. The **udp** keyword specifies that the ASA should use UDP to send syslog messages to the syslog server.
- **format emblem** keyword enables EMBLEM format logging for the syslog server.
- **secure** keyword specifies that the connection to the remote logging host should use SSL/TLS for TCP only. Secure logging does not support UDP; an error occurs if you try to use this protocol.

- [**reference-identity** *reference_identity_name*] enables RFC 6125 reference identity checks on the certificate based on the previously configured reference identity object. See Configure Reference Identities, on page 811 for details on the reference identity object.

Example:

```
ciscoasa(config)# logging host inside 10.0.0.1 TCP/1500 secure  reference-identity
syslogServer
```

Generate Syslog Messages in EMBLEM Format to a Syslog Server

To generate syslog messages in EMBLEM format to a syslog server, perform the following steps:

Procedure

Send syslog messages in EMBLEM format to a syslog server over UDP using port 514.

logging host *interface_name ip_address*{**tcp** [/*port*] | **udp** [/ *port*]] [**format emblem**]

Example:

```
ciscoasa(config)# logging host interface_1 127.0.0.1 udp format emblem
ciscoasa(config)# logging host interface_1 2001::1 udp format emblem
```

You can configure IPv4 or IPv6 syslog servers.

The **format emblem** keyword enables EMBLEM format logging for the syslog server (UDP only). The *interface_name* argument specifies the interface through which you access the syslog server. The *ip_address* argument specifies the IP address of the syslog server. The **tcp**[/*port*] or **udp**[/*port*] keyword and argument pair specify that the ASA should use TCP or UDP to send syslog messages to the syslog server.

You can configure the ASA to send data to a syslog server using either UDP or TCP. The default protocol is UDP if you do not specify a protocol.

You can use multiple **logging host** commands to specify additional servers that would all receive syslog messages. If you configure two or more logging servers, make sure that you limit the logging severity level to warnings for all logging servers.

Warning
If you specify TCP, when the ASA discovers syslog server failures, for security reasons, new connections through the ASA are blocked. To permit new connections despite syslog server failures, see Step 3 of Send Syslog Messages to an External Syslog Server, on page 1309.

If you specify UDP, the ASA continues to allow new connections whether or not the syslog server is operational. Valid port values for either protocol are 1025 through 65535. The default UDP port is 514. The default TCP port is 1470.

Note
Sending syslogs over TCP is not supported on a standby ASA.

Generate Syslog Messages in EMBLEM Format to Other Output Destinations

To generate syslog messages in EMBLEM format to other output destinations, perform the following steps:

Procedure

Send syslog messages in EMBLEM format to output destinations other than a syslog server, such as Telnet or SSH sessions.

logging emblem

Example:

```
ciscoasa(config)# logging emblem
```

Send Syslog Messages to the Internal Log Buffer

You need to specify which syslog messages should be sent to the internal log buffer, which serves as a temporary storage location. New messages are appended to the end of the list. When the buffer is full, that is, when the buffer wraps, old messages are overwritten as new messages are generated, unless you configure the ASA to save the full buffer to another location.

To send syslog messages to the internal log buffer, perform the following steps:

Procedure

Step 1 Specify which syslog messages should be sent to the internal log buffer, which serves as a temporary storage location.

logging buffered {*severity_level* | *message_list*}

Example:

```
ciscoasa(config)# logging buffered critical
ciscoasa(config)# logging buffered level 2
ciscoasa(config)# logging buffered notif-list
```

New messages are appended to the end of the list. When the buffer is full, that is, when the buffer wraps, old messages are overwritten as new messages are generated, unless you configure the ASA to save the full buffer to another location. To empty the internal log buffer, enter the **clear logging buffer** command.

Step 2 Change the size of the internal log buffer. The default buffer size is 4 KB.

logging buffer-size *bytes*

Example:

```
ciscoasa(config)# logging buffer-size 16384
```

Note
When you change the logging buffer size, the existing logs in the buffer are purged, and a new buffer is created with the newly configured size.

Step 3 Choose one of the following options:

- Save new messages to the internal log buffer and save the full log buffer content to the internal flash memory.

  logging flash-bufferwrap

  Example:

  ```
  ciscoasa(config)# logging flash-bufferwrap
  ```

  Note
  If the buffer-size is over 2 MB, this command will stop writing data in flash, without any warning.

- Save new messages to the internal log buffer and save the full log buffer content to an FTP server.

  logging ftp-bufferwrap

  Example:

  ```
  ciscoasa(config)# logging ftp-bufferwrap
  ```

  When saving the buffer content to another location, the ASA create log files with names that use the following time-stamp format:

  ```
  LOG-YYYY-MM-DD-HHMMSS.TXT
  ```

  where *YYYY* is the year, *MM* is the month, *DD* is the day of the month, and *HHMMSS* is the time in hours, minutes, and seconds.

- Identify the FTP server on which you want to store log buffer content.

  logging ftp-server *server pathusername password*

  Example:

  ```
  ciscoasa(config)# logging ftp-server 10.1.1.1 /syslogs logsupervisor 1luvMy10gs
  ```

  The *server* argument specifies the IP address of the external FTP server. The *path* argument specifies the directory path on the FTP server where the log buffer data is to be saved. This path is relative to the FTP root directory. The *username* argument specifies a username that is valid for logging into the FTP server. The *password* argument indicates the password for the username specified.

- Save the current log buffer content to the internal flash memory.

  logging savelog [*savefile*]

  Example:

```
ciscoasa(config)# logging savelog latest-logfile.txt
```

Change the Amount of Internal Flash Memory Available for Logs

To change the amount of internal flash memory available for logs, perform the following steps:

Procedure

Step 1 Specify the maximum amount of internal flash memory available for saving log files.

logging flash-maximum-allocation *kbytes*

Example:

```
ciscoasa(config)# logging flash-maximum-allocation 1200
```

By default, the ASA can use up to 50 MB of internal flash memory for log data. The minimum amount of internal flash memory that must be free for the ASA to save log data is 3 MB. The maximum limit of the flash-maximum-allocation value is 2 GB.

If a log file being saved to internal flash memory would cause the amount of free internal flash memory to fall below the configured minimum limit, the ASA deletes the oldest log files to ensure that the minimum amount of memory remains free after saving the new log file. If there are no files to delete or if, after all old files have been deleted, free memory is still below the limit, the ASA fails to save the new log file.

Step 2 Specify the minimum amount of internal flash memory that must be free for the ASA to save a log file.

logging flash-minimum-free *kbytes*

Example:

```
ciscoasa(config)# logging flash-minimum-free 4000
```

Send Syslog Messages to an E-mail Address

To send syslog messages to an e-mail address, perform the following steps:

Procedure

Step 1 Specify which syslog messages should be sent to an e-mail address.

logging mail {*severity_level* | *message_list*}

Example:

```
ciscoasa(config)# logging mail high-priority
```

When sent by e-mail, a syslog message appears in the subject line of the e-mail message. For this reason, we recommend configuring this option to notify administrators of syslog messages with high severity levels, such as critical, alert, and emergency.

Step 2 Specify the source e-mail address to be used when sending syslog messages to an e-mail address.

logging from-address *email_address*

Example:

```
ciscoasa(config)# logging from-address xxx-001@example.com
```

Step 3 Specify the recipient e-mail address to be used when sending syslog messages to an e-mail address.

logging recipient-address *e-mail_address*[*severity_level*]

Example:

```
ciscoasa(config)# logging recipient-address admin@example.com
```

Step 4 Specify the SMTP server to be used when sending syslog messages to an e-mail address. You may provide a primary and secondary server address to ensure unfailing log messaging service. Optionally, you can also associate an interface along with the server to identify the routing table to be used for logging. If no interface is provided, ASA would refer to the management routing table , and in absence of a route entry it would look at the data routing table.

smtp-server [*primary-interface*] *primary-smpt-server-ip-address* [[*backup-interface*]*backup-smpt-server-ip-address*]

Example:

```
ciscoasa(config)# smtp-server 10.1.1.24 10.1.1.34
ciscoasa(config)# smtp-server 10.1.1.24
ciscoasa(config)# smtp-server management 10.1.1.24 outside 10.1.1.34
ciscoasa(config)# smtp-server management 10.1.1.24
```

Send Syslog Messages to ASDM

To send syslog messages to ASDM, perform the following steps:

Procedure

Step 1 Specify which syslog messages should be sent to ASDM.

logging asdm {*severity_level* | *message_list*}

Example:

```
ciscoasa(config)# logging asdm 2
```

The ASA sets aside a buffer area for syslog messages waiting to be sent to ASDM and saves messages in the buffer as they occur. The ASDM log buffer is a different buffer than the internal log buffer. When the ASDM log buffer is full, the ASA deletes the oldest syslog message to make room in the buffer for new ones. Deletion of the oldest syslog message to make room for new ones is the default setting in ASDM. To control the number of syslog messages retained in the ASDM log buffer, you can change the size of the buffer.

Step 2 Specify the number of syslog messages to be retained in the ASDM log buffer.

logging asdm-buffer-size *num_of_msgs*

Example:

```
ciscoasa(config)# logging asdm-buffer-size 200
```

Enter the **clear logging asdm** command to empty the current content of the ASDM log buffer.

Configure the Logging Queue

To configure the logging queue, perform the following steps:

Procedure

Specify the number of syslog messages that the ASA can hold in its queue before sending them to the configured output destination.

logging queue *message_count*

Example:

```
ciscoasa(config)# logging queue 300
```

The ASA have a fixed number of blocks in memory that can be allocated for buffering syslog messages while they are waiting to be sent to the configured output destination. The number of blocks required depends on the length of the syslog message queue and the number of syslog servers specified. The default queue size is 512 syslog messages. The queue size is limited only by block memory availability. Valid values are from 0 to 8192 messages, depending on the platform. If the logging queue is set to zero, the queue is the maximum configurable size (8192 messages).

Send Syslog Messages to the Console Port

To send syslog messages to the console port, perform the following steps:

Procedure

Specify which syslog messages should be sent to the console port.

logging console { *severity_level* | *message_list*}

Example:

```
ciscoasa(config)# logging console errors
```

Send Syslog Messages to an SNMP Server

To enable logging to an SNMP server, perform the following steps.

Procedure

Enable SNMP logging, and specify which messages are to be sent to SNMP servers.

logging history [*logging_list* | *level*]

Example:

```
ciscoasa(config)# logging history errors
```

Send Syslog Messages to a Telnet or SSH Session

To send syslog messages to a Telnet or SSH session, perform the following steps:

Procedure

Step 1 Specify which syslog messages should be sent to a Telnet or SSH session.

logging monitor {*severity_level* | *message_list*}

Example:

```
ciscoasa(config)# logging monitor 6
```

Step 2 Enable logging to the current session only.

terminal monitor

Example:

```
ciscoasa(config)# terminal monitor
```

If you log out and then log in again, you need to reenter this command. Enter the **terminal no monitor** command to disable logging to the current session.

Configure Syslog Messages

Show or Hide Invalid Usernames in Syslogs

You can show or hide invalid usernames in syslog messages for unsuccessful login attempts. The default setting is to hide usernames when the username is invalid or if the validity is unknown. If a user accidentally types a password instead of a username, for example, then it is more secure to hide the "username" in the resultant syslog message. You might want to show invalid usernames to help with troubleshooting login issues.

Procedure

Step 1 Show invalid usernames:

no logging hide username

Step 2 Hide invalid usernames:

logging hide username

Include the Date and Time in Syslog Messages

To include the date and time in syslog messages, perform the following steps:

Procedure

Specify that syslog messages should include the date and time that they were generated.

logging timestamp

Example:

```
ciscoasa(config)# logging timestamp
LOG-2008-10-24-081856.TXT
```

To remove the date and time from syslog messages, enter the **no logging timestamp** command.

Disable a Syslog Message

To disable a specified syslog message, perform the following steps:

Procedure

Prevent the ASA from generating a particular syslog message.

no logging message *syslog_id*

Example:

```
ciscoasa(config)# no logging message 113019
```

To reenable a disabled syslog message, enter the **logging message** *syslog_id* command (for example, **logging message 113019**). To reenable logging of all disabled syslog messages, enter the **clear configure logging disabled** command.

Change the Severity Level of a Syslog Message

To change the severity level of a syslog message, perform the following steps:

Procedure

Specify the severity level of a syslog message.

logging message *syslog_id* **level** *severity_level*

Example:

```
ciscoasa(config)# logging message 113019 level 5
```

To reset the severity level of a syslog message to its setting, enter the **no logging message** *syslog_id* **level** *severity_level* command (for example, **no logging message 113019 level 5**). To reset the severity level of all modified syslog messages to their settings, enter the **clear configure logging level** command.

Block Syslog Messages on a Standby Unit

Procedure

Use the following command to block a specific syslog message from being generated on a standby unit.

no logging message *syslog-id* **standby**

Example:

```
ciscoasa(config)# no logging message 403503 standby
```

Unblock a specific syslog message to ensure that the syslog messages of the failover standby ASA stay synchronized if failover occurs. Use the **logging standby** command to unblock a specific syslog message that was previously blocked from being generated on a standby unit.

Note
During a steady state when both active and standby ASAs are logging, the traffic doubles on the shared logging destinations, such as syslog servers, SNMP servers, and FTP servers. However, at times of a failover, during the switchover phase, the standby ASA generates more events including switchover intrusion and connection events of the active unit.

Include the Device ID in Non-EMBLEM Format Syslog Messages

To include the device ID in non-EMBLEM format syslog messages, perform the following steps:

Procedure

Configure the ASA to include a device ID in non-EMBLEM-format syslog messages. You can specify only one type of device ID for syslog messages.

logging device-id {**cluster-id** | **context-name** | **hostname** | **ipaddress** *interface_name* [**system**] | **string** *text*}

Example:

```
ciscoasa(config)# logging device-id hostname
ciscoasa(config)# logging device-id context-name
```

The **context-name** keyword indicates that the name of the current context should be used as the device ID (applies to multiple context mode only). If you enable the logging device ID for the admin context in multiple context mode, messages that originate in the system execution space use a device ID of **system**, and messages that originate in the admin context use the name of the admin context as the device ID.

Note
In an ASA cluster, always use the control unit IP address for the selected interface.

The **cluster-id** keyword specifies the unique name in the boot configuration of an individual ASA unit in the cluster as the device ID. The **hostname** keyword specifies that the hostname of the ASA should be used as the device ID. The **ipaddress** *interface_name* keyword-argument pair specifies that the interface IP address specified as *interface_name* should be used as the device ID. If you use the **ipaddress** keyword, the device ID becomes the specified ASA interface IP address, regardless of the interface from which the syslog message is sent. In the cluster environment, the **system** keyword dictates that the device ID becomes the system IP address on the interface. This keyword provides a single, consistent device ID for all syslog messages that are sent from the device. The **string** *text* keyword-argument pair specifies that the text string should be used as the device ID. The string can include as many as 16 characters.

You cannot use blank spaces or any of the following characters:

- & (ampersand)
- ‘ (single quote)
- “ (double quote)
- < (less than)
- > (greater than)
- ? (question mark)

Note
If enabled, the device ID does not appear in EMBLEM-formatted syslog messages nor in SNMP traps.

Create a Custom Event List

You use the following three criteria to define an event list:

- Event Class
- Severity
- Message ID

To create a custom event list to send to a specific logging destination (for example, an SNMP server), perform the following steps:

Procedure

Step 1 Specify criteria for selecting messages to be saved in the internal log buffer. For example, if you set the severity level to 3, then the ASA sends syslog messages for severity levels 3, 2, and 1.

logging list *name* {**level** *level* [**class** *message_class*] | **message** *start_id*[*-end_id*]}

Example:

```
ciscoasa(config)# logging list list-notif level 3
```

The *name* argument specifies the name of the list. The **level** *level* keyword and argument pair specify the severity level. The **class** *message_class* keyword-argument pair specify a particular message class. The **message** *start_id* [*-end_id*] keyword-argument pair specify an individual syslog message number or a range of numbers.

Note
Do not use the names of severity levels as the name of a syslog message list. Prohibited names include emergencies, alert, critical, error, warning, notification, informational, and debugging. Similarly, do not use the first three characters of these words at the beginning of an event list name. For example, do not use an event list name that starts with the characters “err.”

Step 2 (Optional) Add more criteria for message selection to the list.

logging list *name* {**level** *level* [**class** *message_class*] | **message** *start_id*[*-end_id*]}

Example:

```
ciscoasa(config)# logging list list-notif message 104024-105999
ciscoasa(config)# logging list list-notif level critical
ciscoasa(config)# logging list list-notif level warning class ha
```

Enter the same command as in the previous step, specifying the name of the existing message list and the additional criterion. Enter a new command for each criterion that you want to add to the list. For example, you can specify criteria for syslog messages to be included in the list as the following:

- Syslog message IDs that fall into the range of 104024 to 105999.
- All syslog messages with the critical severity level or higher (emergency, alert, or critical).
- All ha class syslog messages with the warning severity level or higher (emergency, alert, critical, error, or warning).

Note
A syslog message is logged if it satisfies any of these conditions. If a syslog message satisfies more than one of the conditions, the message is logged only once.

Configure Logging Filters

Send All Syslog Messages in a Class to a Specified Output Destination

To send all syslog messages in a class to a specified output destination, perform the following steps:

Procedure

Override the configuration in the specified output destination command. For example, if you specify that messages at severity level 7 should go to the internal log buffer and that ha class messages at severity level 3 should go to the internal log buffer, then the latter configuration takes precedence.

logging class *message_class* {**buffered** | **console** | **history** | **mail** | **monitor** | **trap**} [*severity_level*]

Example:

```
ciscoasa(config)# logging class ha buffered alerts
```

The **buffered**, **history**, **mail**, **monitor**, and **trap** keywords specify the output destination to which syslog messages in this class should be sent. The **history** keyword enables SNMP logging. The **monitor** keyword enables Telnet and SSH logging. The **trap** keyword enables syslog server logging. Select one destination per command line entry. To specify that a class should go to more than one destination, enter a new command for each output destination.

Limit the Rate of Syslog Message Generation

To limit the rate of syslog message generation, perform the following steps:

Procedure

Apply a specified severity level (1 through 7) to a set of messages or to an individual message (not the destination) within a specified time period.

logging rate-limit {**unlimited** | {*num* [*interval*]}} **message** *syslog_id* | **level** *severity_level*

Example:

```
ciscoasa(config)# logging rate-limit 1000 600 level 6
```

Rate limits affect the volume of messages being sent to all configured destinations. To reset the logging rate limit to the default value, enter the **clear running-config logging rate-limit** command. To reset the logging rate limit, enter the **clear configure logging rate-limit** command.

Monitoring the Logs

See the following commands for monitoring logging status.

- **show logging**

  This command shows syslog messages, including the severity level.

Note

- The maximum number of syslog messages that are available to view is 1000, which is the default setting. The maximum number of syslog messages that are available to view is 2000.
- The Host TX counter in the output for **show logging** will be displayed even after executing the **clear logging counters all** command.

- **show logging message**

  This command shows a list of syslog messages with modified severity levels and disabled syslog messages.

- **show logging message** *message_ID*

  This command shows the severity level of a specific syslog message.

- **show logging queue**

  This command shows the logging queue and queue statistics.

- **show running-config logging rate-limit**

This command shows the current logging rate-limit setting.

- **Configuration** > **Firewall** > **Access Rules**

  This pane allows you to filter the live viewer of logging to that specific logs based on the search criteria (Rule Hex Id). To view the results, select the rule and click **Show Log**.

Examples for Logging

The following examples show the logging information, that displays for the **show logging** command:

```
ciscoasa(config)# show logging
Syslog logging: enabled
    Facility: 16
    Timestamp logging: disabled
    Standby logging: disabled
    Deny Conn when Queue Full: disabled
    Console logging: disabled
    Monitor logging: disabled
    Buffer logging: disabled
    Trap logging: level errors, facility 16, 3607 messages logged
        Logging to infrastructure 10.1.2.3
    History logging: disabled
    Device ID: 'inside' interface IP address “10.1.1.1”
    Mail logging: disabled
    ASDM logging: disabled
```

```
ciscoasa (config)# show logging
Syslog logging: enabled
    Facility: 20
    Timestamp logging: disabled
    Hide Username logging: enabled
    Standby logging: disabled
    Debug-trace logging: enabled
    Console logging: disabled
    Monitor logging: disabled
    Buffer logging: level debugging, 330272 messages logged
    Trap logging: level debugging, facility 20, 325464 messages logged
        Logging to inside 2001:164:5:1::123
    Permit-hostdown logging: disabled
    History logging: disabled
    Device ID: disabled
    Mail logging: disabled
    ASDM logging: disabled
```

The following examples show how to control both whether a syslog message is enabled and the severity level of the specified syslog message:

```
ciscoasa(config)# show logging message 403503
syslog 403503: -level errors (enabled)

ciscoasa(config)# logging message 403503 level 1
ciscoasa(config)# show logging message 403503
syslog 403503: -level errors, current-level alerts (enabled)

ciscoasa(config)# no logging message 403503
ciscoasa(config)# show logging message 403503
```

```
syslog 403503: -level errors, current-level alerts (disabled)

ciscoasa(config)# logging message 403503
ciscoasa(config)# show logging message 403503
syslog 403503: -level errors, current-level alerts (enabled)

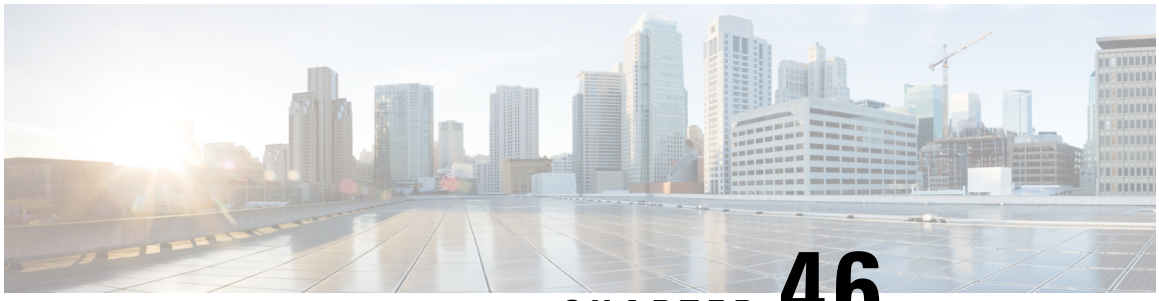
ciscoasa(config)# no logging message 403503 level 3
ciscoasa(config)# show logging message 403503
syslog 403503: -level errors (enabled)
```

History for Logging

Table 60: History for Logging

Feature Name	Platform Releases	Description
Logging	7.0(1)	Provides ASA network logging information through various output destinations, and includes the option to view and save log files.
Rate limit	7.0(4)	Limits the rate at which syslog messages are generated. We introduced the following command: **logging rate-limit**.
Logging list	7.2(1)	Creates a logging list to use in other commands to specify messages by various criteria (logging level, event class, and message IDs). We introduced the following command: **logging list**.
Secure logging	8.0(2)	Specifies that the connection to the remote logging host should use SSL/TLS. This option is valid only if the protocol selected is TCP. We modified the following command: **logging host**.
Logging class	8.0(4), 8.1(1)	Added support for the ipaa event class of logging messages. We modified the following command: **logging class**.
Logging class and saved logging buffers	8.2(1)	Added support for the dap event class of logging messages. We modified the following command: **logging class**. Added support to clear the saved logging buffers (ASDM, internal, FTP, and flash). We introduced the following command: **clear logging queue bufferwrap**.
Password encryption	8.3(1)	Added support for password encryption. We modified the following command: **logging ftp server**.
Log viewers	8.3(1)	The source and destination IP addresses were added to the log viewers.

Feature Name	Platform Releases	Description
Enhanced logging and connection blocking	8.3(2)	When you configure a syslog server to use TCP, and the syslog server is unavailable, the ASA blocks new connections that generate syslog messages until the server becomes available again (for example, VPN, firewall, and cut-through-proxy connections). This feature has been enhanced to also block new connections when the logging queue on the ASA is full; connections resume when the logging queue is cleared. This feature was added for compliance with Common Criteria EAL4+. Unless required, we recommended allowing connections when syslog messages cannot be sent or received. To allow connections, continue to use the **logging permit-hostdown** command. We introduced the following syslog messages: 414005, 414006, 414007, and 414008. We modified the following command: **show logging**.
Syslog message filtering and sorting	8.4(1)	Support has been added for the following: • Syslog message filtering based on multiple text strings that correspond to various columns • Creation of custom filters • Column sorting of messages. For detailed information, see the ASDM configuration guide. This feature interoperates with all ASA versions.
Clustering	9.0(1)	Added support for syslog message generation in a clustering environment on the ASA 5580 and 5585-X. We modified the following command: **logging device-id**.
Blocking syslogs on a standby unit	9.4(1)	We added support for blocking the generation of specific syslog messages on the standby unit in a failover configuration. We introduced the following command: **logging message** *syslog-id* **standby**.
Reference Identities for Secure Syslog Server connections	9.6(2)	TLS client processing now supports rules for verification of a server identity defined in RFC 6125, Section 6. Identity verification will be done during PKI validation for TLS connections to the Syslog Server. If the presented identity cannot be matched against the configured reference identity, the connection is not established. We added or modified the following commands: **[no] crypto ca reference-identity**, **logging host**.
IPv6 address support for syslog servers	9.7(1)	You can now configure syslog servers with IPv6 addresses to record, send, and receive syslogs over TCP and UDP. We modified the following command: **logging host**
Logging class	9.12(1)	Added support for the BFD, BGP, interface, IPv6, Muticast, Object-Group-Search, PBR, routing, SLA class of logging messages. We modified the following command: **logging class**.



CHAPTER 46

SNMP

This chapter describes how to configure Simple Network Management Protocol (SNMP) to monitor ASA.

- About SNMP, on page 1327
- Guidelines for SNMP, on page 1343
- Configure SNMP, on page 1348
- Monitoring SNMP, on page 1357
- Examples for SNMP, on page 1358
- History for SNMP, on page 1359

About SNMP

SNMP is an application-layer protocol that facilitates the exchange of management information between network devices and is part of the TCP/IP protocol suite. The ASA provides support for network monitoring using SNMP Versions 1, 2c, and 3, and support the use of all three versions simultaneously. The SNMP agent running on the ASA interface lets you monitor the network devices through network management systems (NMSes), such as HP OpenView. The ASA support SNMP read-only access through issuance of a GET request. SNMP write access is not allowed, so you cannot make changes with SNMP. In addition, the SNMP SET request is not supported.

You can configure the ASA to send traps, which are unsolicited messages from the managed device to the management station for certain events (event notifications) to an NMS, or you can use the NMS to browse the Management Information Bases (MIBs) on the security devices. MIBs are a collection of definitions, and the ASA maintain a database of values for each definition. Browsing a MIB means issuing a series of GET-NEXT or GET-BULK requests of the MIB tree from the NMS to determine values.

Note With intense workloads, deploying more than 10 NMS can impact the device's performance. To ensure device's stability and responsiveness, we recommend that you cautiously utilize NMS in conducting SNMP walk polling and in managing the trap traffic.

The ASA have an SNMP agent that notifies designated management stations if events occur that are predefined to require a notification, for example, when a link in the network goes up or down. The notification it sends includes an SNMP OID, which identifies itself to the management stations. The ASA agent also replies when a management station asks for information.

SNMP Terminology

The following table lists the terms that are commonly used when working with SNMP.

Table 61: SNMP Terminology

Term	Description
Agent	The SNMP server running on the ASA. The SNMP agent has the following features: • Responds to requests for information and actions from the network management station. • Controls access to its Management Information Base, the collection of objects that the SNMP manager can view or change. • Does not allow SET operations.
Browsing	Monitoring the health of a device from the network management station by polling required information from the SNMP agent on the device. This activity may include issuing a series of GET-NEXT or GET-BULK requests of the MIB tree from the network management station to determine values.
Management Information Bases (MIBs)	Standardized data structures for collecting information about packets, connections, buffers, failovers, and so on. MIBs are defined by the product, protocols, and hardware standards used by most network devices. SNMP network management stations can browse MIBs and request specific data or events be sent as they occur.
Network management stations (NMSs)	The PCs or workstations set up to monitor SNMP events and manage devices, such as the ASA.
Object identifier (OID)	The system that identifies a device to its NMS and indicates to users the source of information monitored and displayed.
Trap	Predefined events that generate a message from the SNMP agent to the NMS. Events include alarm conditions such as linkup, linkdown, coldstart, warmstart, authentication, or syslog messages.

MIBs and Traps

MIBs are either standard or enterprise-specific. Standard MIBs are created by the IETF and documented in various RFCs. A trap reports significant events occurring on a network device, most often errors or failures. SNMP traps are defined in either standard or enterprise-specific MIBs. Standard traps are created by the IETF and documented in various RFCs. SNMP traps are compiled into the ASA software.

If needed, you can also download RFCs, standard MIBs, and standard traps from the following locations:

http://www.ietf.org/

Browse the complete list of Cisco MIBs, traps, and OIDs from the following location:

ftp://ftp.cisco.com/pub/mibs/supportlists/asa/asa-supportlist.html

In addition, download Cisco OIDs by FTP from the following location:

ftp://ftp.cisco.com/pub/mibs/oid/oid.tar.gz

Note In software versions 7.2(1), 8.0(2), and later, the interface information accessed through SNMP refreshes about every 5 seconds. As a result, we recommend that you wait for at least 5 seconds between consecutive polls.

Not all OIDs in MIBs are supported. To obtain a list of the supported SNMP MIBs and OIDs for a specific ASA, enter the following command:

```
ciscoasa(config)# show snmp-server oidlist
```

Note Although the **oidlist** keyword does not appear in the options list for the **show snmp-server** command help, it is available. However, this command is for Cisco TAC use only. Contact the Cisco TAC before using this command.

This sample displays the output from the **show snmp-server oidlist** command:

```
ciscoasa(config)# show snmp-server oidlist
[0]     1.3.6.1.2.1.1.1.        sysDescr
[1]     1.3.6.1.2.1.1.2.        sysObjectID
[2]     1.3.6.1.2.1.1.3.        sysUpTime
[3]     1.3.6.1.2.1.1.4.        sysContact
[4]     1.3.6.1.2.1.1.5.        sysName
[5]     1.3.6.1.2.1.1.6.        sysLocation
[6]     1.3.6.1.2.1.1.7.        sysServices
[7]     1.3.6.1.2.1.2.1.        ifNumber
[8]     1.3.6.1.2.1.2.2.1.1.    ifIndex
[9]     1.3.6.1.2.1.2.2.1.2.    ifDescr
[10]    1.3.6.1.2.1.2.2.1.3.    ifType
[11]    1.3.6.1.2.1.2.2.1.4.    ifMtu
[12]    1.3.6.1.2.1.2.2.1.5.    ifSpeed
[13]    1.3.6.1.2.1.2.2.1.6.    ifPhysAddress
[14]    1.3.6.1.2.1.2.2.1.7.    ifAdminStatus
[15]    1.3.6.1.2.1.2.2.1.8.    ifOperStatus
[16]    1.3.6.1.2.1.2.2.1.9.    ifLastChange
[17]    1.3.6.1.2.1.2.2.1.10.   ifInOctets
[18]    1.3.6.1.2.1.2.2.1.11.   ifInUcastPkts
[19]    1.3.6.1.2.1.2.2.1.12.   ifInNUcastPkts
[20]    1.3.6.1.2.1.2.2.1.13.   ifInDiscards
[21]    1.3.6.1.2.1.2.2.1.14.   ifInErrors
[22]    1.3.6.1.2.1.2.2.1.16.   ifOutOctets
[23]    1.3.6.1.2.1.2.2.1.17.   ifOutUcastPkts
[24]    1.3.6.1.2.1.2.2.1.18.   ifOutNUcastPkts
[25]    1.3.6.1.2.1.2.2.1.19.   ifOutDiscards
[26]    1.3.6.1.2.1.2.2.1.20.   ifOutErrors
[27]    1.3.6.1.2.1.2.2.1.21.   ifOutQLen
[28]    1.3.6.1.2.1.2.2.1.22.   ifSpecific
[29]    1.3.6.1.2.1.4.1.        ipForwarding
[30]    1.3.6.1.2.1.4.20.1.1.   ipAdEntAddr
[31]    1.3.6.1.2.1.4.20.1.2.   ipAdEntIfIndex
[32]    1.3.6.1.2.1.4.20.1.3.   ipAdEntNetMask
[33]    1.3.6.1.2.1.4.20.1.4.   ipAdEntBcastAddr
[34]    1.3.6.1.2.1.4.20.1.5.   ipAdEntReasmMaxSize
[35]    1.3.6.1.2.1.11.1.       snmpInPkts
[36]    1.3.6.1.2.1.11.2.       snmpOutPkts
```

```
[37]    1.3.6.1.2.1.11.3.       snmpInBadVersions
[38]    1.3.6.1.2.1.11.4.       snmpInBadCommunityNames
[39]    1.3.6.1.2.1.11.5.       snmpInBadCommunityUses
[40]    1.3.6.1.2.1.11.6.       snmpInASNParseErrs
[41]    1.3.6.1.2.1.11.8.       snmpInTooBigs
[42]    1.3.6.1.2.1.11.9.       snmpInNoSuchNames
[43]    1.3.6.1.2.1.11.10.      snmpInBadValues
[44]    1.3.6.1.2.1.11.11.      snmpInReadOnlys
[45]    1.3.6.1.2.1.11.12.      snmpInGenErrs
[46]    1.3.6.1.2.1.11.13.      snmpInTotalReqVars
[47]    1.3.6.1.2.1.11.14.      snmpInTotalSetVars
[48]    1.3.6.1.2.1.11.15.      snmpInGetRequests
[49]    1.3.6.1.2.1.11.16.      snmpInGetNexts
[50]    1.3.6.1.2.1.11.17.      snmpInSetRequests
[51]    1.3.6.1.2.1.11.18.      snmpInGetResponses
[52]    1.3.6.1.2.1.11.19.      snmpInTraps
[53]    1.3.6.1.2.1.11.20.      snmpOutTooBigs
[54]    1.3.6.1.2.1.11.21.      snmpOutNoSuchNames
[55]    1.3.6.1.2.1.11.22.      snmpOutBadValues
[56]    1.3.6.1.2.1.11.24.      snmpOutGenErrs
[57]    1.3.6.1.2.1.11.25.      snmpOutGetRequests
[58]    1.3.6.1.2.1.11.26.      snmpOutGetNexts
[59]    1.3.6.1.2.1.11.27.      snmpOutSetRequests
[60]    1.3.6.1.2.1.11.28.      snmpOutGetResponses
[61]    1.3.6.1.2.1.11.29.      snmpOutTraps
[62]    1.3.6.1.2.1.11.30.      snmpEnableAuthenTraps
[63]    1.3.6.1.2.1.11.31.      snmpSilentDrops
[64]    1.3.6.1.2.1.11.32.      snmpProxyDrops
[65]    1.3.6.1.2.1.31.1.1.1.1. ifName
[66]    1.3.6.1.2.1.31.1.1.1.2. ifInMulticastPkts
[67]    1.3.6.1.2.1.31.1.1.1.3. ifInBroadcastPkts
[68]    1.3.6.1.2.1.31.1.1.1.4. ifOutMulticastPkts
[69]    1.3.6.1.2.1.31.1.1.1.5. ifOutBroadcastPkts
[70]    1.3.6.1.2.1.31.1.1.1.6. ifHCInOctets
--More--
```

SNMP Object Identifiers

Each Cisco system-level product has an SNMP object identifier (OID) for use as a MIB-II sysObjectID. The CISCO-PRODUCTS-MIB and the CISCO-ENTITY-VENDORTYPE-OID-MIB includes the OIDs that can be reported in the sysObjectID object in the SNMPv2-MIB, Entity Sensor MIB and Entity Sensor Threshold Ext MIB. You can use this value to identify the model type. The following table lists the sysObjectID OIDs for ASA and ISA models.

Table 62: SNMP Object Identifiers

Product Identifier	sysObjectID	Model Number
ASAv	ciscoASAv (ciscoProducts 1902)	Cisco Adaptive Security Virtual Appliance (ASAv)
ASAv System Context	ciscoASAvsy (ciscoProducts 1903)	Cisco Adaptive Security Virtual Appliance (ASAv) System Context
ASAv Security Context	ciscoASAvsc (ciscoProducts 1904)	Cisco Adaptive Security Virtual Appliance (ASAv) Security Context.
ISA 30004C Industrial Security Appliance	ciscoProducts 2268	ciscoISA30004C

Product Identifier	sysObjectID	Model Number
CISCO ISA30004C with 4 GE Copper Security Context	ciscoProducts 2139	ciscoISA30004Csc
CISCO ISA30004C with 4 GE Copper System Context	ciscoProducts 2140	ciscoISA30004Csy
ISA 30002C2F Industrial Security Appliance	ciscoProducts 2267	ciscoISA30002C2F
CISCO ISA30002C2F with 2 GE Copper ports + 2 GE Fiber Security Context	ciscoProducts 2142	ciscoISA30002C2Fsc
CISCO ISA30002C2F with 2 GE Copper ports + 2 GE Fiber System Context	ciscoProducts 2143	ciscoISA30002C2Fsy
Cisco Industrial Security Appliance (ISA) 30004C Chassis	cevChassis 1677	cevChassisISA30004C
Cisco Industrial Security Appliance (ISA) 30002C2F Chassis	cevChassis 1678	cevChassisISA30002C2F
Central Processing Unit Temperature Sensor for ISA30004C Copper SKU	cevSensor 187	cevSensorISA30004CCpuTempSensor
Central Processing Unit Temperature Sensor for ISA30002C2F Fiber	cevSensor 189	cevSensorISA30002C2FCpuTempSensor
Processor Card Temperature Sensor for ISA30004C Copper SKU	cevSensor 192	cevSensorISA30004CPTS
Processor Card Temperature Sensor for ISA30002C2F Fiber SKU	cevSensor 193	cevSensorISA30002C2FPTS
Power Card Temperature Sensor for ISA30004C Copper SKU	cevSensor 197	cevSensorISA30004CPowercardTS
Power Card Temperature Sensor for ISA30002C2F Fiber SKU	cevSensor 198	cevSensorISA30002C2FPowercardTS
Port Card Temperature Sensor for ISA30004C	cevSensor 199	cevSensorISA30004CPortcardTS
Port Card Temperature Sensor for ISA30002C2F	cevSensor 200	cevSensorISA30002C2FPortcardTS
Central Processing Unit for ISA30004C Copper SKU	cevModuleCpuType 329	cevCpuISA30004C
Central Processing Unit for ISA30002C2F Fiber SKU	cevModuleCpuType 330	cevCpuISA30002C2F
Modules ISA30004C, ISA30002C2F	cevModule 111	cevModuleISA3000Type

Product Identifier	sysObjectID	Model Number
30004C Industrial Security Appliance Solid State Drive	cevModuleISA3000Type 1	cevModuleISA30004CSSD64
30002C2F Industrial Security Appliance Solid State Drive	cevModuleISA3000Type 2	cevModuleISA30002C2FSSD64
Cisco ISA30004C/ISA30002C2F Hardware Bypass	cevModuleISA3000Type 5	cevModuleISA3000HardwareBypass
Cisco Secure Firewall 240P	ciscoCsf240P (ciscoProducts 3424)	ciscoCsf240P
FirePOWER 4140 Security Appliance, 1U with embedded security module 36	ciscoFpr4140K9 (ciscoProducts 2293)	FirePOWER 4140
FirePOWER 4120 Security Appliance, 1U with embedded security module 24	ciscoFpr4120K9 (ciscoProducts 2294)	FirePOWER 4120
FirePOWER 4110 Security Appliance, 1U with embedded security module 12	ciscoFpr4110K9 (ciscoProducts 2295)	FirePOWER 4110
FirePOWER 4110 Security Module 12	ciscoFpr4110SM12 (ciscoProducts 2313)	FirePOWER 4110 Security Module 12
FirePOWER 4120 Security Module 24	ciscoFpr4120SM24 (ciscoProducts 2314)	FirePOWER 4110 Security Module 24
FirePOWER 4140 Security Module 36	ciscoFpr4140SM36 (ciscoProducts 2315)	FirePOWER 4110 Security Module 36
FirePOWER 4110 Chassis	cevChassis 1714	cevChassisFPR4110
FirePOWER 4120 Chassis	cevChassis 1715	cevChassisFPR4120
FirePOWER 4140 Chassis	cevChassis 1716	cevChassisFPR4140
FirePOWER 4K Fan Bay	cevContainer 363	cevContainerFPR4KFanBay
FirePOWER 4K Power Supply Bay	cevContainer 364	cevContainerFPR4KPowerSupplyBay
FirePOWER 4120 Supervisor Module	cevModuleFPRType 4	cevFPR4120SUPFixedModule
FirePOWER 4140 Supervisor Module	cevModuleFPRType 5	cevFPR4140SUPFixedModule
FirePOWER 4110 Supervisor Module	cevModuleFPRType 7	cevFPR4110SUPFixedModule
Cisco FirePOWER 4110 Security Appliance, Threat Defense	cevChassis 1787	cevChassisCiscoFpr4110td
Cisco FirePOWER 4120 Security Appliance, Threat Defense	cevChassis 1788	cevChassisCiscoFpr4120td
Cisco FirePOWER 4140 Security Appliance, Threat Defense	cevChassis 1789	cevChassisCiscoFpr4140td
Cisco Firepower 9000 Security Module 24, Threat Defense	cevChassis 1791	cevChassisCiscoFpr9000SM24td

Product Identifier	sysObjectID	Model Number
Cisco Firepower 9000 Security Module 24 NEBS, Threat Defense	cevChassis 1792	cevChassisCiscoFpr9000SM24Ntd
Cisco Firepower 9000 Security Module 36, Threat Defense	cevChassis 1793	cevChassisCiscoFpr9000SM36td
Cisco Secure Firewall Threat Defense Virtual, VMware	cevChassis 1795	cevChassisCiscoFTDVVMW
Cisco Firewall Threat Defense Virtual, AWS	cevChassis 1796	cevChassisCiscoFTDVAWS

Physical Vendor Type Values

Each Cisco chassis or standalone system has a unique type number for SNMP use. The entPhysicalVendorType OIDs are defined in the CISCO-ENTITY-VENDORTYPE-OID-MIB. This value is returned in the entPhysicalVendorType object from the ASA, ASAv, or ASASM SNMP agent. You can use this value to identify the type of component (module, power supply, fan, sensors, CPU, and so on). The following table lists the physical vendor type values for the ASA models.

Table 63: Physical Vendor Type Values

Item	entPhysicalVendorType OID Description
Gigabit Ethernet port	cevPortGe (cevPort 109)
Cisco Adaptive Security Virtual Appliance	cevChassisASAv (cevChassis 1451)

Supported Tables and Objects in MIBs

The following table lists the supported tables and objects for the specified MIBs.

In multi-context mode, these tables and objects provide information for a single context. If you want data across contexts, you need to sum them. For example, to get overall memory usage, sum the cempMemPoolHCUsed values for each context.

Table 64: Supported Tables and Objects in MIBs

MIB Name and OID	Supported Tables and Objects
CISCO-ENHANCED-MEMPOOL-MIB; OID:1.3.6.1.4.1.9.9.221	cempMemPoolTable, cempMemPoolIndex, cempMemPoolType, cempMemPoolName, cempMemPoolAlternate, cempMemPoolValid. For a 32-bit memory system, poll using the 32-bit memory counters—cempMemPoolUsed, cempMemPoolFree,cempMemPoolUsedOvrflw, cempMemPoolFreeOvrflw, cempMemPoolLargestFree, cempMemPoolLowestFree, cempMemPoolUsedLowWaterMark, cempMemPoolAllocHit, cempMemPoolAllocMiss, cempMemPoolFreeHit, cempMemPoolFreeMiss, cempMemPoolLargestFreeOvrflw, cempMemPoolLowestFreeOvrflw, cempMemPoolUsedLowWaterMarkOvrflw, cempMemPoolSharedOvrflw. For a 64-bit memory system, poll using the 64-bit memory counters—cempMemPoolHCUsed, cempMemPoolHCFree, cempMemPoolHCLargestFree, cempMemPoolHCLowestFree, cempMemPoolHCUsedLowWaterMark, cempMemPoolHCShared
CISCO-REMOTE-ACCESS-MONITOR-MIB; OID:1.3.6.1.4.1.9.9.392 **Note** These three MIB OIDs can be used to track why remote access connections fail.	crasNumTotalFailures, crasNumSetupFailInsufResources, crasNumAbortedSessions
CISCO-ENTITY-SENSOR-EXT-MIB; OID:1.3.6.1.4.1.9.9.745	ceSensorExtThresholdTable
CISCO-L4L7MODULE-RESOURCE-LIMIT-MIB; OID:1.3.6.1.4.1.9.9.480	ciscoL4L7ResourceLimitTable
CISCO-TRUSTSEC-SXP-MIB; OID:1.3.6.1.4.1.9.9.720 **Note** Not supported on the ASAv.	ctsxSxpGlobalObjects, ctsxSxpConnectionObjects, ctsxSxpSgtObjects
DISMAN-EVENT-MIB; OID:1.3.6.1.2.1.88	mteTriggerTable, mteTriggerThresholdTable, mteObjectsTable, mteEventTable, mteEventNotificationTable
DISMAN-EXPRESSION-MIB; OID:1.3.6.1.2.1.90	expExpressionTable, expObjectTable, expValueTable
ENTITY-SENSOR-MIB; OID: 1.3.6.1.2.1.99 **Note** Provides information related to physical sensors, such as chassis temperature, fan RPM, power supply voltage, etc. Not supported on the ASAv platform.	entPhySensorTable

MIB Name and OID	Supported Tables and Objects
NAT-MIB; OID:1.3.6.1.2.1.123	natAddrMapTable, natAddrMapIndex, natAddrMapName, natAddrMapGlobalAddrType, natAddrMapGlobalAddrFrom, natAddrMapGlobalAddrTo, natAddrMapGlobalPortFrom, natAddrMapGlobalPortTo, natAddrMapProtocol, natAddrMapAddrUsed, natAddrMapRowStatus
CISCO-PTP-MIB; OID:1.3.6.1.4.1.9.9.760 **Note** Only MIBs corresponding to E2E Transparent Clock mode are supported.	ciscoPtpMIBSystemInfo, cPtpClockDefaultDSTable, cPtpClockTransDefaultDSTable, cPtpClockPortTransDSTable
CISCO-PROCESS-MIB; 1.3.6.1.4.1.9.9.109.1.1.1.1.7.1 1.3.6.1.4.1.9.9.109.1.1.1.1.7.2 to 1.3.6.1.4.1.9.9.109.1.1.1.1.7.(n+1)	cpmCPUTotal1minRev Associated parameters and values of cpmCPUTotal1minRev Examples: • .3.6.1.4.1.9.9.109.1.1.1.1.7.(n+2) - Aggregate system CPU utilization % (This value is same as the system cpu usage from .3.6.1.4.1.9.9.109.1.1.1.1.7.1 in single context mode). • .3.6.1.4.1.9.9.109.1.1.1.1.7.(n+3) - Snort average CPU utilization % (total aggregate value of all snort instances) • .3.6.1.4.1.9.9.109.1.1.1.1.7.(n+4) - System process average % (average of "Sysproc" cores)

Supported Traps (Notifications)

The following table lists the supported traps (notifications) and their associated MIBs.

Table 65: Supported Traps (Notifications)

Trap and MIB Name	Varbind List	Description
authenticationFailure (SNMPv2-MIB)	—	For SNMP Version 1 or 2, the community string provided in the SNMP request is incorrect. For SNMP Version 3, a report PDU is generated instead of a trap if the auth or priv passwords or usernames are incorrect. The **snmp-server enable traps snmp authentication** command is used to enable and disable transmission of these traps.
bgpBackwardTransition	bgpPeerLastError, bgpPeerState	The **snmp-server enable traps peer-flap** command is used to enable transmission of BGP peer-flap related trap.

Trap and MIB Name	Varbind List	Description
ccmCLIRunningConfigChanged (CISCO-CONFIG-MAN-MIB)	ccmHistoryRunningLastChanged, ccmHistoryEventTerminalType	The **snmp-server enable traps config** command is used to enable transmission of this trap.
cefcFRUInserted (CISCO-ENTITY-FRU-CONTROL-MIB)	entPhysicalContainedIn	The **snmp-server enable traps entity fru-insert** command is used to enable this notification.
cefcFRURemoved (CISCO-ENTITY-FRU-CONTROL-MIB)	entPhysicalContainedIn	The **snmp-server enable traps entity fru-remove** command is used to enable this notification.

Trap and MIB Name	Varbind List	Description
ceSensorExtThresholdNotification (CISCO-ENTITY-SENSOR-EXT-MIB)	entPhysicalName, entPhysicalDescr, entPhySensorValue, entPhySensorType, ceSensorExtThresholdValue	The **snmp-server enable traps entity** [**power-supply-failure** | **fan-failure** | **cpu-temperature**] command is used to enable transmission of the entity threshold notifications. This notification is sent for a power supply failure. The objects sent identify the fan and CPU temperature. The **snmp-server enable traps entity fan-failure** command is used to enable transmission of the fan failure trap.This trap does not apply to the Firepower 2100 series. The **snmp-server enable traps entity power-supply-failure** command is used to enable transmission of the power supply failure trap.This trap does not apply to the Firepower 2100 series. The **snmp-server enable traps entity chassis-fan-failure** command is used to enable transmission of the chassis fan failure trap. The **snmp-server enable traps entity cpu-temperature** command is used to enable transmission of the high CPU temperature trap. This trap does not apply to the Firepower 2100 series. The **snmp-server enable traps entity power-supply-presence** command is used to enable transmission of the power supply presence failure trap. The **snmp-server enable traps entity power-supply-temperature** command is used to enable transmission of the power supply temperature threshold trap. The **snmp-server enable traps entity chassis-temperature** command is used to enable transmission of the chassis ambient temperature trap. This trap does not apply to the Firepower 2100 series. The **snmp-server enable traps entity accelerator-temperature** command is used to enable transmission of the chassis accelerator temperature trap.
cikeTunnelStart (CISCO-IPSEC-FLOW-MONITOR-MIB)	cikePeerLocalAddr, cikePeerRemoteAddr, cikeTunLifeTime	The **snmp-server enable traps ikev2 start** command is used to enable transmission of ikev2 start trap.
cikeTunnelStop (CISCO-IPSEC-FLOW-MONITOR-MIB)	cikePeerLocalAddr, cikePeerRemoteAddr, cikeTunActiveTime	The **snmp-server enable traps ikev2 stop** command is used to enable transmission of ikev2 stop trap.

Trap and MIB Name	Varbind List	Description
cipSecTunnelStart (CISCO-IPSEC-FLOW-MONITOR-MIB)	cipSecTunLifeTime, cipSecTunLifeSize	The **snmp-server enable traps ipsec start** command is used to enable transmission of this trap.
cipSecTunnelStop (CISCO-IPSEC-FLOW-MONITOR-MIB)	cipSecTunActiveTime	The **snmp-server enable traps ipsec stop** command is used to enable transmission of this trap.
ciscoConfigManEvent (CISCO-CONFIG-MAN-MIB)	ccmHistoryEventCommandSource, ccmHistoryEventConfigSource, ccmHistoryEventConfigDestination	The **snmp-server enable traps config** command is used to enable transmission of this trap.
ciscoRasTooManySessions (CISCO-REMOTE-ACCESS-MONITOR-MIB)	crasNumSessions, crasNumUsers, crasMaxSessionsSupportable, crasMaxUsersSupportable, crasThrMaxSessions	The **snmp-server enable traps remote-access session-threshold-exceeded** command is used to enable transmission of these traps.
ciscoUFwFailoverStateChanged (CISCO-UNIFIED-FIREWALL-MIB)	gid, FOStatus	The **snmp-server enable traps failover-state** command is used to enable transmission of failover-state trap.
clogMessageGenerated (CISCO-SYSLOG-MIB)	clogHistFacility, clogHistSeverity, clogHistMsgName, clogHistMsgText, clogHistTimestamp	Syslog messages are generated. The value of the clogMaxSeverity object is used to decide which syslog messages are sent as traps. The **snmp-server enable traps syslog** command is used to enable and disable transmission of these traps.
clrResourceLimitReached (CISCO-L4L7MODULE-RESOURCE-LIMIT-MIB)	crlResourceLimitValueType, crlResourceLimitMax, clogOriginIDType, clogOriginID	The **snmp-server enable traps connection-limit-reached** command is used to enable transmission of the connection-limit-reached notification. The clogOriginID object includes the context name from which the trap originated.
coldStart (SNMPv2-MIB)	—	The coldStart trap that occurs when the SNMP agent starts after the SNMP configuration. This trap also occurs when the agent starts after a system reboot. **Note** For cluster and HA nodes, post a reload, if the interfaces reboot time exceeds 5 minutes (preset threshold), the trap is dropped. When the cluster and HA nodes have rebooted sucessfully, all other traps are sent as expected. The **snmp-server enable traps snmp coldstart** command is used to enable and disable transmission of these traps.

Trap and MIB Name	Varbind List	Description
cpmCPURisingThreshold (CISCO-PROCESS-MIB)	cpmCPURisingThresholdValue, cpmCPUTotalMonIntervalValue, cpmCPUInterruptMonIntervalValue, cpmCPURisingThresholdPeriod, cpmProcessTimeCreated, cpmProcExtUtil5SecRev	The **snmp-server enable traps cpu threshold rising** command is used to enable transmission of the CPU threshold rising notification. The cpmCPURisingThresholdPeriod object is sent with the other objects.
cufwClusterStateChanged (CISCO-UNIFIED-FIREWALL-MIB)	status	The **snmp-server enable traps cluster-state** command is used to enable transmission of cluster-state trap.
entConfigChange (ENTITY-MIB)	—	The **snmp-server enable traps entity config-change fru-insert fru-remove** command is used to enable this notification. **Note** This notification is only sent in multimode when a security context is created or removed.
linkDown (IF-MIB)	ifIndex, ifAdminStatus, ifOperStatus	The linkdown trap for interfaces. The **snmp-server enable traps snmp linkdown** command is used to enable and disable transmission of these traps.
linkUp (IF-MIB)	ifIndex, ifAdminStatus, ifOperStatus	The linkup trap for interfaces. The **snmp-server enable traps snmp linkup** command is used to enable and disable transmission of these traps.
mteTriggerFired (DISMAN-EVENT-MIB)	mteHotTrigger, mteHotTargetName, mteHotContextName, mteHotOID, mteHotValue, cempMemPoolName, cempMemPoolHCUsed	The **snmp-server enable traps memory-threshold** command is used to enable the memory threshold notification. The mteHotOID is set to cempMemPoolHCUsed. The cempMemPoolName and cempMemPoolHCUsed objects are sent with the other objects.
mteTriggerFired (DISMAN-EVENT-MIB)	mteHotTrigger, mteHotTargetName, mteHotContextName, mteHotOID, mteHotValue, ifHCInOctets, ifHCOutOctets, ifHighSpeed, entPhysicalName	The **snmp-server enable traps interface-threshold** command is used to enable the interface threshold notification. The entPhysicalName objects are sent with the other objects.
natPacketDiscard (NAT-MIB)	ifIndex	The **snmp-server enable traps nat packet-discard** command is used to enable the NAT packet discard notification. This notification is rate limited for 5 minutes and is generated when IP packets are discarded by NAT because mapping space is not available. The ifIndex gives the ID of the mapped interface.

Trap and MIB Name	Varbind List	Description
ospfNbrStateChange	ospfRouterId, ospfNbrIpAddr, ospfNbrAddressLessIndex, ospfNbrRtrId, ospfNbrState	The **snmp-server enable traps peer-flap** command is used to enable transmission of OSPF peer-flap related trap. **Note** For ASA5585 models, the SNMP engine is changed to use the netsnmp version 5.8 library and the following OIDs are not available in the library: • ospfIfStateChange 1.3.6.1.2.1.14.16.2.16 • ospfVirtIfStateChange 1.3.6.1.2.1.14.16.2.1 • ospfVirtNbrStateChange 1.3.6.1.2.1.14.16.2.3
warmStart (SNMPv2-MIB)	—	The warmStart trap that occurs when the SNMP agent restarts for the first time. This trap also occurs when the agent restarts after a SNMP configuration change where, all the SNMP host configuration are removed and a fresh SNMP configuration is done. The **snmp-server enable traps snmp warmstart** command is used to enable and disable transmission of these traps.

Interface Types and Examples

The interface types that produce SNMP traffic statistics include the following:

- Logical—Statistics collected by the software driver, which are a subset of physical statistics.
- Physical—Statistics collected by the hardware driver. Each physical named interface has a set of logical and physical statistics associated with it. Each physical interface may have more than one VLAN interface associated with it. VLAN interfaces only have logical statistics.

Note For a physical interface that has multiple VLAN interfaces associated with it, be aware that SNMP counters for ifInOctets and ifOutoctets OIDs match the aggregate traffic counters for that physical interface.

- VLAN-only—SNMP uses logical statistics for ifInOctets and ifOutOctets.

The examples in the following table show the differences in SNMP traffic statistics. Example 1 shows the difference in physical and logical output statistics for the **show interface** command and the **show traffic** command. Example 2 shows output statistics for a VLAN-only interface for the **show interface** command and the **show traffic** command. The example shows that the statistics are close to the output that appears for the **show traffic** command.

Table 66: SNMP Traffic Statistics for Physical and VLAN Interfaces

Example 1

```
ciscoasa# show interface GigabitEthernet3/2
interface GigabitEthernet3/2
description fullt-mgmt
nameif mgmt
security-level 10
ip address 10.7.14.201 255.255.255.0
management-only

ciscoasa# show traffic
(Condensed output)

Physical Statistics
GigabitEthernet3/2:
received (in 121.760 secs)
36 packets       3428 bytes
0 pkts/sec      28 bytes/sec

Logical Statistics
mgmt:
received (in 117.780 secs)
36 packets       2780 bytes
0 pkts/sec      23 bytes/sec
```

The following examples show the SNMP output statistics for the management interface and the physical interface. The ifInOctets value is close to the physical statistics output that appears in the **show traffic** command output but not to the logical statistics output.

ifIndex of the mgmt interface:

```
IF_MIB::ifDescr.6 = Adaptive Security Appliance 'mgmt' interface
```

ifInOctets that corresponds to the physical interface statistics:

```
IF-MIB::ifInOctets.6 = Counter32:3246
```

Example 2

```
ciscoasa# show interface GigabitEthernet
interface GigabitEthernet0/0.100
vlan 100
nameif inside
security-level 100
ip address 10.7.1.101 255.255.255.0 stan

ciscoasa# show traffic
inside
received (in 9921.450 secs)
1977 packets       126528 bytes
0 pkts/sec      12 bytes/sec
transmitted (in 9921.450 secs)
1978 packets       126556 bytes
0 pkts/sec      12 bytes/sec
```

ifIndex of VLAN inside:

```
IF-MIB::ifDescr.9 = Adaptive Security Ap
IF-MIB::ifInOctets.9 = Counter32: 126318
```

SNMP Version 3 Overview

SNMP Version 3 provides security enhancements that are not available in SNMP Version 1 or Version 2c. SNMP Versions 1 and 2c transmit data between the SNMP server and SNMP agent in clear text. SNMP Version 3 adds authentication and privacy options to secure protocol operations. In addition, this version controls access to the SNMP agent and MIB objects through the User-based Security Model (USM) and View-based Access Control Model (VACM). The ASA also supports the creation of SNMP groups and users, as well as hosts, which is required to enable transport authentication and encryption for secure SNMP communications.

Security Models

For configuration purposes, the authentication and privacy options are grouped together into security models. Security models apply to users and groups, which are divided into the following three types:

- NoAuthPriv—No Authentication and No Privacy, which means that no security is applied to messages.
- AuthNoPriv—Authentication but No Privacy, which means that messages are authenticated.
- AuthPriv—Authentication and Privacy, which means that messages are authenticated and encrypted.

SNMP Groups

An SNMP group is an access control policy to which users can be added. Each SNMP group is configured with a security model, and is associated with an SNMP view. A user within an SNMP group must match the security model of the SNMP group. These parameters specify what type of authentication and privacy a user within an SNMP group uses. Each SNMP group name and security model pair must be unique.

SNMP Users

SNMP users have a specified username, a group to which the user belongs, authentication password, encryption password, and authentication and encryption algorithms to use. The authentication algorithm options are SHA-1, SHA-224, SHA-256 HMAC, and SHA-384. The encryption algorithm options are 3DES and AES (which is available in 128, 192, and 256 versions). When you create a user, you must associate it with an SNMP group. The user then inherits the security model of the group.



Note When configuring an SNMP v3 user account, ensure that the length of authentication algorithm is equal to or greater than the length of encryption algorithm.

SNMP Hosts

An SNMP host is an IP address to which SNMP notifications and traps are sent. To configure SNMP Version 3 hosts, along with the target IP address, you must configure a username, because traps are only sent to a configured user. SNMP target IP addresses and target parameter names must be unique on the ASA. Each SNMP host can have only one username associated with it. To receive SNMP traps, after you have added the **snmp-server host** command, make sure that you configure the user credentials on the NMS to match the credentials for the ASA.



Note You can add up to 4000 hosts. However, only 128 of this number can be for traps.

Implementation Differences Between the ASA and Cisco IOS Software

The SNMP Version 3 implementation in the ASA differs from the SNMP Version 3 implementation in the Cisco IOS software in the following ways:

- The local-engine and remote-engine IDs are not configurable. The local engine ID is generated when the ASA starts or when a context is created.

- No support exists for view-based access control, which results in unrestricted MIB browsing.
- Support is restricted to the following MIBs: USM, VACM, FRAMEWORK, and TARGET.
- You must create users and groups with the correct security model.
- You must remove users, groups, and hosts in the correct sequence.
- Use of the snmp-server host command creates an ASA rule to allow incoming SNMP traffic.

SNMP Syslog Messaging

SNMP generates detailed syslog messages that are numbered 212*nnn*. Syslog messages indicate the status of SNMP requests, SNMP traps, SNMP channels, and SNMP responses from the ASA or ASASM to a specified host on a specified interface.

For detailed information about syslog messages, see the syslog messages guide.

Note SNMP polling fails if SNMP syslog messages exceed a high rate (approximately 4000 per second).

Application Services and Third-Party Tools

For information about SNMP support, see the following URL:

http://www.cisco.com/en/US/tech/tk648/tk362/tk605/tsd_technology_support_sub-protocol_home.html

For information Refer to SNMP Chapter in Cisco ASA Series General Operations ASDM Configuration Guide, see the following URL:

https://www.cisco.com/c/en/us/support/security/adaptive-security-appliance-asa-software/products-installation-and-configuration-guides-list.html

Guidelines for SNMP

This section includes the guidelines and limitations that you should review before configuring SNMP.

Failover and Clustering Guidelines

- When using SNMPv3 with clustering or failover, if you add a new cluster unit after the initial cluster formation or you replace a failover unit, then SNMPv3 users are not replicated to the new unit. You must re-add the SNMPv3 users to the control/active unit to force the users to replicate to the new unit; or you can add the users directly on the new unit (SNMPv3 users and groups are an exception to the rule that you cannot enter configuration commands on a cluster data unit). Reconfigure each user by entering the **snmp-server user** *username group-name* **v3** command on the control/active unit or directly to the data/standby unit with the *priv-password* option and *auth-password* option in their unencrypted forms.

IPv6 Guidelines (All ASA Models)

SNMP can be configured over IPv6 transport so that an IPv6 host can perform SNMP queries and receive SNMP notifications from a device running IPv6 software. The SNMP agent and related MIBs have been enhanced to support IPv6 addressing.

IPv6 Guidelines for the Firepower 2100

The Firepower 2100 runs an underlying operating system called the FXOS, and supports both Appliance mode (the default) and Platform mode; see Set the Firepower 2100 to Appliance or Platform Mode, on page 39.

When in Platform mode, you must configure an IPv6 management IP address in FXOS. The following example configures an IPv6 management interface and gateway:

```
Firepower-chassis# scope fabric-interconnect a
Firepower-chassis /fabric-interconnect # scope ipv6-config
Firepower-chassis /fabric-interconnect/ipv6-config # show ipv6-if
Management IPv6 Interface:
IPv6 Address Prefix IPv6 Gateway
---------------------------------- ---------- ------------
2001::8998 64 2001::1
Firepower-chassis /fabric-interconnect/ipv6-config # set out-of-band ipv6 2001::8999
ipv6-prefix 64 ipv6-gw 2001::1
Firepower-chassis /fabric-interconnect/ipv6-config* # commit-buffer
Firepower-chassis /fabric-interconnect/ipv6-config #
```

Additional Guidelines

- Power supply traps are not issued for systems operating in Appliance mode.
- For the Firepower 2100 in Platform mode, you cannot poll member interfaces of an EtherChannel, and traps for member interfaces are not generated. This functionality is supported if you enable SNMP directly in FXOS. Appliance mode is not affected.
- Does not support ASA traps for individual individual port members for the Firepower 2100 in Platform mode; see Cisco Firepower 2100 FXOS MIB Reference Guide.
- For the Firepower 2100 in Appliance mode, you cannot poll the hardware model and serial, and in ASA, traps are not generated for these details. Hence, configure SNMP on FXOS or Chassis Manager to poll Chassis Management IP instead of the interfaces on ASA instance.
- You must have Cisco Works for Windows or another SNMP MIB-II compliant browser to receive SNMP traps or browse a MIB.
- Management-access over a VPN tunnel is not supported with SNMP (the **management-access** command).
- Does not support view-based access control, but the VACM MIB is available for browsing to determine default view settings.
- The ENTITY-MIB is not available in the non-admin context. Use the IF-MIB instead to perform queries in the non-admin context.
- The ENTITY-MIB is not available for the Firepower 9300. Instead, use CISCO-FIREPOWER-EQUIPMENT-MIB and CISCO-FIREPOWER-SM-MIB.
- On some devices, the order of interfaces (ifDescr) in the output of **snmpwalk** has been observed to change after a reboot. The ASA uses an algorithm to determine the ifIndex table that SNMP queries. When the ASA is booted up, the interfaces are added to the ifIndex table in the order loaded as the ASA

reads the configuration. New interfaces added to the ASA are appended to the list of interfaces in the ifIndex table. As interfaces are added, removed, or renamed, it can affect the order of interfaces on reboot.

- When you provide an OID in the **snmpwalk** command, the snmpwalk tool queries all variables in the subtree that is below the specified OID and displays their values. Thus, to view a comprehensive output of the objects on the device, ensure to provide the OID in the **snmpwalk** command.
- Does not support SNMP Version 3 for the AIP SSM or AIP SSC.
- Does not support SNMP debugging.
- Does not support retrieval of ARP information.
- Does not support SNMP SET commands.
- When using NET-SNMP Version 5.4.2.1, only supports the encryption algorithm version of AES128. Does not support the encryption algorithm versions of AES256 or AES192.
- Changes to the existing configuration are rejected if the result places the SNMP feature in an inconsistent state.
- For SNMP Version 3, configuration must occur in the following order: group, user, host.
- For Firepower 2100, when SNMPv3 is configured over the device management interface, all SNMPv3 users can poll the device even when they are not mapped in the Host configuration.
- Before a group is deleted, you must ensure that all users associated with that group are deleted.
- Before a user is deleted, you must ensure that no hosts are configured that are associated with that username.
- If users have been configured to belong to a particular group with a certain security model, and if the security level of that group is changed, you must do the following in this sequence:
  - Remove the users from that group.
  - Change the group security level.
  - Add users that belong to the new group.
- The creation of custom views to restrict user access to a subset of MIB objects is not supported.
- All requests and traps are available in the default Read/Notify View only.
- The connection-limit-reached trap is generated in the admin context. To generate this trap. you must have at least one SNMP server host configured in the user context in which the connection limit has been reached.
- You can add up to 4000 hosts. However, only 128 of this number can be for traps.
- The total number of supported active polling destinations is 128.
- You can specify a network object to indicate the individual hosts that you want to add as a host group.
- You can associate more than one user with one host.
- You can specify overlapping network objects in different **host-group** commands. The values that you specify for the last host group take effect for the common set of hosts in the different network objects.

- If you delete a host group or hosts that overlap with other host groups, the hosts are set up again using the values that have been specified in the configured host groups.
- The values that the hosts acquire depend on the specified sequence that you use to run the commands.
- The limit on the message size that SNMP sends is 1472 bytes.
- The ASA supports an unlimited number of SNMP server trap hosts per context. The **show snmp-server host** command output displays only the active hosts that are polling the ASA, as well as the statically configured hosts.

Troubleshooting Tips

- To ensure that the SNMP process that receives incoming packets from the NMS is running, enter the following command:

```
ciscoasa(config)# show process | grep snmp
```

- To capture syslog messages from SNMP and have them appear on the ASA console, enter the following commands:

```
ciscoasa(config)# logging list snmp message 212001-212015
ciscoasa(config)# logging console snmp
```

- To make sure that the SNMP process is sending and receiving packets, enter the following commands:

```
ciscoasa(config)# clear snmp-server statistics
ciscoasa(config)# show snmp-server statistics
```

  The output is based on the SNMP group of the SNMPv2-MIB.

- To make sure that SNMP packets are going through the ASA and to the SNMP process, enter the following commands:

```
ciscoasa(config)# clear asp drop
ciscoasa(config)# show asp drop
```

- If the NMS cannot request objects successfully or is not handing incoming traps from the ASA correctly, use a packet capture to isolate the problem, by entering the following commands:

```
ciscoasa (config)# access-list snmp permit udp any eq snmptrap any
ciscoasa (config)# access-list snmp permit udp any any eq snmp
ciscoasa (config)# capture snmp type raw-data access-list snmp interface mgmt
ciscoasa (config)# copy /pcap capture:snmp tftp://192.0.2.5/exampledir/snmp.pcap
```

- If the ASA is not performing as expected, obtain information about network topology and traffic by doing the following:
  - For the NMS configuration, obtain the following information:

    Number of timeouts

Retry count

Engine ID caching

Username and password used

- Issue the following commands:

  show block

  show interface

  show process

  show cpu

  show vm

- If a fatal error occurs, to help in reproducing the error, send a traceback file and the output of the **show tech-support** command to Cisco TAC.
- If SNMP traffic is not being allowed through the ASA interfaces, you might also need to permit ICMP traffic from the remote SNMP server using the **icmp permit** command.
- If you have configured the device with **snmp-server enable oid**, when doing SNMP walk operations, the the ASA will query memory information from the MEMPOOL_DMA and MEMPOOL_GLOBAL_SHARED pools. This can result in SNMP-related CPU hogs causing packet drops. To mitigate this issue, avoid polling the OIDs that relate to the Global Shared pool using the **no snmp-server enable oid** command. When disabled, the mempool OIDs would return 0 bytes.
- When you use SNMPGET with a large number of OID's in a single request for polling ASP drop counters requires repeated polling of ASP drop counters that results in higher CPU usage. Hence, we recommended that you identify important counters to monitor and use SNMPGET on each counter to get these values such that there is limited cpu impact.
- When SNMP is configured on more than one context in a multiple-context ASA, poll the contexts sequentially and use SNMPBULKGET instead of snmpwalk to reduce the number of connections to the platform. This approach will avoid delays or timeouts from SNMP when a large number of contexts are being polled at the same time.
- The ASA consistently sets the Don't Fragment (DF) bit when responding to SNMP polls with SNMP get-response messages, such as the SNMPBULKGET, that requires the entire network path to support at least the Maximum Transmission Unit (MTU) configured in the ASA. If a lower MTU is configured on the network path, other devices may send Internet Control Message Protocol (ICMP) packets requesting for fragmentation. However, since the DF bit is set, the ASA does not reply and does not fragment the packets, resulting in a lack of response from the ASA.

  To address this issue, you can modify the MTU on the ASA or throughout the entire network path, use multiple get requests instead of SNMPBULKGET, or reduce the bulk size in SNMPBULKGET requests.

- For additional troubleshooting information, see the following URL:
  http://www.cisco.com/c/en/us/support/docs/security/asa-5500-x-series-next-generation-firewalls/116423-troubleshoot-asa-snmp.html

Configure SNMP

This section describes how to configure SNMP.

Procedure

Step 1 Enable the SNMP Agent and SNMP server.
Step 2 Configure SNMP traps.
Step 3 Configure SNMP Version 1 and 2c parameters or SNMP Version 3 parameters.

Enable the SNMP Agent and SNMP Server

To enable the SNMP agent and SNMP server, perform the following steps:

Procedure

Enable the SNMP agent and SNMP server on the ASA. By default, the SNMP server is enabled.

snmp-server enable

Example:

```
ciscoasa(config)# snmp-server enable
```

Configure SNMP Traps

To designate which traps that the SNMP agent generates and how they are collected and sent to NMSs, perform the following steps:

Note When you enable all SNMP or syslog traps, it is possible for the SNMP process to consume excess resources in the agent and in the network, causing the system to hang. If you notice system delays, unfinished requests, or timeouts, you can selectively enable SNMP and syslog traps. For example, you can skip *Informational* syslog trap severity level.

Procedure

Send individual traps, sets of traps, or all traps to the NMS.

snmp-server enable traps [**all** | **syslog** | **snmp** [**authentication** | **linkup** | **linkdown** | **coldstart** | **warmstart**] | **config** | **entity** [**config-change** | **fru-insert** | **fru-remove** | **fan-failure** | **cpu-temperature** | **chassis-fan-failure** | **power-supply**] | **chassis-temperature** | **power-supply-presence** | **power-supply-temperature** **l1-bypass-status**] | **ikev2** [**start** | **stop**] | **cluster-state** | **failover-state** | **peer-flap** | **ipsec** [**start** | **stop**] | **remote-access** [**session-threshold-exceeded**] | **connection-limit-reached** | **cpu threshold rising** | **interface-threshold** | **memory-threshold** | **nat** [**packet-discard**]

Example:

```
ciscoasa(config)# snmp-server enable traps snmp authentication
linkup linkdown coldstart warmstart
```

This command enables syslog messages to be sent as traps to the NMS. The default configuration has all SNMP standard traps enabled, as shown in the example. To disable these traps, use the **no snmp-server enable traps snmp** command.

If you enter this command and do not specify a trap type, the default is the **syslog** trap. By default, the **syslog** trap is enabled. The default SNMP traps continue to be enabled with the **syslog** trap.

You need to configure both the **logging history** command and the **snmp-server enable traps syslog** command to generate traps from the syslog MIB.

To restore the default enabling of SNMP traps, use the **clear configure snmp-server** command. All other traps are disabled by default.

Traps available in the admin context only:

- **connection-limit-reached**
- **entity**
- **memory-threshold**

Traps generated through the admin context only for physically connected interfaces in the system context:

- **interface-threshold**

All other traps are available in the admin and user contexts in single mode.

The **config** trap enables the ciscoConfigManEvent notification and the ccmCLIRunningConfigChanged notification, which are generated after you have exited configuration mode.

If the CPU usage is greater than the configured threshold value for the configured monitoring period, the **cpu threshold rising** trap is generated.

When the used system context memory reaches 80 percent of the total system memory, the **memory-threshold** trap is generated from the admin context. For all other user contexts, this trap is generated when the used memory reaches 80 percent of the total system memory in that particular context.

Some traps are not applicable to certain hardware models. Use ? in place of a trap keyword to determine which traps are available for your device. For example:

- The Firepower 1000 series supports the following entity traps only: **chassis-temperature**, **config-change**, and **cpu-temperature**.

Note

SNMP does not monitor voltage sensors.

Configure a CPU Usage Threshold

To configure a CPU usage threshold, perform the following steps:

Procedure

Configure the threshold value for a high CPU threshold and the threshold monitoring period.

snmp cpu threshold rising *threshold_value monitoring_period*

Example:

```
ciscoasa(config)# snmp cpu threshold rising 75% 30 minutes
```

To clear the threshold value and monitoring period of the CPU utilization, use the **no** form of this command. If the **snmp cpu threshold rising** command is not configured, the default for the high threshold level is over 70 percent, and the default for the critical threshold level is over 95 percent. The default monitoring period is set to 1 minute.

You cannot configure the critical CPU threshold level, which is maintained at a constant 95 percent. Valid threshold values for a high CPU threshold range from 10 to 94 percent. Valid values for the monitoring period range from 1 to 60 minutes.

Configure a Physical Interface Threshold

To configure the physical interface threshold, perform the following steps:

Procedure

Configure the threshold value for an SNMP physical interface.

snmp interface threshold *threshold_value*

Example:

```
ciscoasa(config)# snmp interface threshold 75%
```

To clear the threshold value for an SNMP physical interface, use the **no** form of this command. The threshold value is defined as a percentage of interface bandwidth utilization. Valid threshold values range from 30 to 99 percent. The default value is 70 percent.

The **snmp interface threshold** command is available only in the admin context.

Physical interface usage is monitored in single mode and multimode, and traps for physical interfaces in the system context are sent through the admin context. Only physical interfaces are used to compute threshold usage.

Configure Parameters for SNMP Version 1 or 2c

To configure parameters for SNMP Version 1 or 2c, perform the following steps:

Procedure

Step 1 Specify the recipient of an SNMP notification, indicate the interface from which traps are sent, and identify the name and IP address of the NMS or SNMP manager that can connect to the ASA.

snmp-server host{*interface hostname* | *ip_address*} [**trap**| **poll**] [**community** *community-string*] [**version** {**1 2c**| *username*}] [**udp-port** *port*]

Example:

```
ciscoasa(config)# snmp-server host mgmt 10.7.14.90 version 2c
ciscoasa(config)# snmp-server host corp 172.18.154.159 community public

ciscoasa(config)# snmp-server host mgmt 12:ab:56:ce::11 version 2c
```

The **trap** keyword limits the NMS to receiving traps only. The **poll** keyword limits the NMS to sending requests (polling) only. By default, SNMP traps are enabled. By default, the UDP port is 162. The community string is a shared secret key between the ASA and the NMS. The key is a case-sensitive value up to 32 alphanumeric characters long. Spaces are not permitted. The default community string is public. The ASA uses this key to determine whether or not the incoming SNMP request is valid. For example, you could designate a site with a community string and then configure the ASA and the management station with the same string. The ASA uses the specified string and do not respond to requests with an invalid community string. However, if SNMP monitoring is through the management interface instead of the diagnostic interface, polling takes place without ASA validating the community string. After you have used an encrypted community string, only the encrypted form is visible to all systems (for example, CLI, ASDM, CSM, and so on). The clear text password is not visible. The encrypted community string is always generated by the ASA; you normally enter the clear text form.

The **version** keyword specifies the SNMP version to use for traps and requests (polling). Communication with the server is allowed using the selected version only.

To receive traps after you have added the **snmp-server host** command, make sure that you configure the user on the NMS with the same credentials as the credentials configured on the ASA.

Step 2 Set the community string, which is for use *only* with SNMP Version 1 or 2c.

snmp-server community *community-string*

Example:

```
ciscoasa(config)# snmp-server community onceuponatime
```

Note
You should avoid the use of special characters (!, @, #, $, %, ^, &, *, \) in community strings. In general, using any special characters reserved for functions used by the operating system can cause unexpected results. For example, the backslash (\) is interpreted as an escape character and should not be used in the community string.

Step 3 Set the SNMP server location or contact information.

snmp-server [**contact** | **location**] *text*

Example:

```
ciscoasa(config)# snmp-server location building 42
ciscoasa(config)# snmp-server contact EmployeeA
```

The *text* argument specifies the name of the contact person or the ASA system administrator. The name is case sensitive and can be up to 127 characters. Spaces are accepted, but multiple spaces are shortened to a single space.

Step 4 Set the listening port for SNMP requests.

snmp-server listen-port *lport*

Example:

```
ciscoasa(config)# snmp-server lport 192
```

The *lport* argument is the port on which incoming requests are accepted. The default listening port is 161. The **snmp-server listen-port** command is only available in admin context, and is not available in the system context. If you configure the **snmp-server listen-port** command on a port that is currently in use, the following message appears:

```
The UDP port port is in use by another feature. SNMP requests to the device
will fail until the snmp-server listen-port command is configured to use a different port.
```

The existing SNMP thread continues to poll every 60 seconds until the port is available, and issues syslog message %ASA-1-212001 if the port is still in use.

Configure Parameters for SNMP Version 3

To configure parameters for SNMP Version 3, perform the following steps:

Procedure

Step 1 Specify a new SNMP group, which is for use *only* with SNMP Version 3.

snmp-server group *group-name* **v3** [**auth** | **noauth** | **priv**]

Example:

```
ciscoasa(config)# snmp-server group testgroup1 v3 auth
```

When a community string is configured, two additional groups with the name that matches the community string are autogenerated: one for the Version 1 security model and one for the Version 2 security model. The **auth** keyword enables packet authentication. The **noauth** keyword indicates no packet authentication or encryption is being used. The **priv** keyword enables packet encryption and authentication. No default values exist for the **auth** or **priv** keywords.

Step 2 Configure a new user for an SNMP group, which is for use only with SNMP Version 3.

snmp-server user *username group_name* **v3** [**engineID** *engineID*] [**encrypted**] [**auth** {**sha** | **sha224** | **sha256** | **sha384**} *auth_password* [**priv** {**3des** | **aes** {**128** | **192** | **256**}} *priv_password*]]

Example:

```
ciscoasa(config)# snmp-server user testuser1 testgroup1 v3 auth md5 testpassword
aes 128 mypassword
ciscoasa(config)# snmp-server user testuser1 public v3 encrypted auth md5
00:11:22:33:44:55:66:77:88:99:AA:BB:CC:DD:EE:FF
```

The username argument is the name of the user on the host that belongs to the SNMP agent. Enter up to 32 characters for the username. The name must begin with a letter. Valid characters include letters, numbers, _ (underscore), . (period), @ (at sign), and - (hyphen).

The group-name argument is the name of the group to which the user belongs. The **v3** keyword specifies that the SNMP Version 3 security model should be used and enables the use of the **encrypted**, **priv**, and the **auth** keywords. The **engineID** keyword is optional and specifies the engineID of the ASA which was used to localize the user's authentication and encryption information. The engineID argument must specify a valid ASA engineID.

The **encrypted** keyword specifies the password in encrypted format. Encrypted passwords must meet the following requirements.

- Must be in hexadecimal format.
- Must contain a minimum of 8 characters and a maximum of 80 characters.
- Must contain only letters, numbers, and the following characters: ~`!@#%^&*()_-+{}[]|\:;"'<,>./
- Must not contain the following symbols: $ (dollar sign), ? (question mark), or = (equals sign).
- Must contain at least five different characters.
- Must not contain too many consecutively incrementing or decrementing numbers or letters. For example, the string "12345" has four such characters, and the string "ZYXW" has three. If the total number of such characters exceeds a certain limit (typically more than around 4-6 such occurrences), the simplicity check will fail.

  Note
  The consecutively incrementing or decrementing character count is not reset when non-incrementing or decrementing characters are used in between. For example, abcd&!21 will fail the password check, but abcd&!25, will not.

The **auth** keyword specifies which authentication level (**sha**, **sha224**, **sha256**, or **sha384**) should be used. The **priv** keyword specifies the encryption level. No default values for the **auth** or **priv** keywords, or default passwords exist.

For the encryption algorithm, you can specify the **3des** or **aes** keyword. You can also specify which version of the AES encryption algorithm to use: **128**, **192**, or **256**. The auth-password argument specifies the authentication user password. The priv-password argument specifies the encryption user password.

If you forget a password, you cannot recover it and you must reconfigure the user. You can specify a plain-text password or a localized digest. The localized digest must match the authentication algorithm selected for the user, which can be SHA, SHA-224, SHA-256, or SHA-384. When the user configuration is displayed on the console or is written to a file (for example, the startup-configuration file), the localized authentication and privacy digests are always displayed instead of a plain-text password (see the second example). The minimum length for a password is 1 alphanumeric character; however, we recommend that you use at least 8 alphanumeric characters for security.

When using SNMPv3 with clustering or failover, if you add a new cluster unit after the initial cluster formation or you replace a failover unit, then SNMPv3 users are not replicated to the new unit. You must re-add the SNMPv3 users to the control/active unit to force the users to replicate to the new unit; or you can add the users directly on the new unit (SNMPv3 users and groups are an exception to the rule that you cannot enter configuration commands on a cluster data unit). Reconfigure each user by entering the **snmp-server user** *username group-name* **v3** command on the control/active unit or directly to the data/standby unit with the *priv-password* option and *auth-password* option in their unencrypted forms.

If you enter a user on the control/active unit with the **encrypted** keyword, an error message appears to inform you that the SNMPv3 user commands will not be replicated. This behavior also means that existing SNMPv3 user and group commands are not cleared during replication.

For example, a control/active unit using commands entered with encrypted keys:

```
ciscoasa(config)# snmp-server user defe abc v3 encrypted auth sha
c0:e7:08:50:47:eb:2e:e4:3f:a3:bc:45:f6:dd:c3:46:25:a0:22:9a
priv aes 256 cf:ad:85:5b:e9:14:26:ae:8f:92:51:12:91:16:a3:ed:de:91:6b:f7:
f6:86:cf:18:c0:f0:47:d6:94:e5:da:01
ERROR: This command cannot be replicated because it contains localized keys.
```

For example, a data unit during cluster replication (appears only if an **snmp-server user** commands exist in the configuration):

```
ciscoasa(cfg-cluster)#
Detected Cluster Master.
Beginning configuration replication from Master.
WARNING: existing snmp-server user CLI will not be cleared.
```

Step 3 Specify the recipient of an SNMP notification. Indicate the interface from which traps are sent. Identify the name and IP address of the NMS or SNMP manager that can connect to the ASA.

snmp-server host *interface* {*hostname* | *ip_address*} [**trap**| **poll**] [**community** *community-string*] [**version** {**1** | **2c** | **3** *username*}] [**udp-port** *port*]

Example:

```
ciscoasa(config)# snmp-server host mgmt 10.7.14.90 version 3 testuser1
ciscoasa(config)# snmp-server host mgmt 10.7.26.5 version 3 testuser2
ciscoasa(config)# snmp-server host mgmt 12:ab:56:ce::11 version 3 testuser3
```

The **trap** keyword limits the NMS to receiving traps only. The **poll** keyword limits the NMS to sending requests (polling) only. By default, SNMP traps are enabled. By default, the UDP port is 162. The community string is a shared secret key between the ASA and the NMS. The key is a case-sensitive value up to 32 alphanumeric characters. Spaces are not permitted. The default community-string is public. The ASA uses this key to determine whether the incoming SNMP request is valid. For example, you could designate a site with a community string and then configure the ASA and the NMS with the same string. The ASA uses the specified string and do not respond to requests with an invalid community string. After you have used an encrypted community string, only the encrypted form is visible to all systems (for example, CLI, ASDM, CSM, and so on). The clear text password is not visible. The encrypted community string is always generated by the ASA; you normally enter the clear text form.

The **version** keyword specifies the SNMP version to use for traps and requests (polling). Communication with the server is allowed using the selected version only.

When SNMP Version 3 hosts are configured on the ASA, a user must be associated with that host.

To receive traps after you have added the **snmp-server host** command, make sure that you configure the user on the NMS with the same credentials as the credentials configured on the ASA.

Step 4 Set the SNMP server location or contact information.

snmp-server [**contact** | **location**] *text*

Example:

```
ciscoasa(config)# snmp-server location building 42
ciscoasa(config)# snmp-server contact EmployeeA
```

The *text* argument specifies the name of the contact person or the ASA system administrator. The name is case sensitive and can be up to 127 characters. Spaces are accepted, but multiple spaces are shortened to a single space.

Step 5 Set the listening port for SNMP requests.

snmp-server listen-port *lport*

Example:

```
ciscoasa(config)# snmp-server lport 192
```

The *lport* argument is the port on which incoming requests are accepted. The default listening port is 161. The **snmp-server listen-port** command is only available in admin context, and is not available in the system context. If you configure the **snmp-server listen-port** command on a port that is currently in use, the following message appears:

```
The UDP port port is in use by another feature. SNMP requests to the device
will fail until the snmp-server listen-port command is configured to use a different port.
```

The existing SNMP thread continues to poll every 60 seconds until the port is available, and issues syslog message %ASA-1-212001 if the port is still in use.

Configure a Group of Users

To configure an SNMP user list with a group of specified users in it, perform the following steps:

Procedure

Configure an SNMP user list.

snmp-server user-list *list_name* **username** *user_name*

Example:

```
ciscoasa(config)# snmp-server user-list engineering username user1
```

The *listname* argument specifies the name of the user list, which may be up to 33 characters long. The **username** *user_name* keyword-argument pair specifies the users who may be configured in the user list. You configure the users in the user list with the **snmp-server user** *username* command, which is available only if you are using SNMP Version 3. The user list must have more than one user in it and can be associated with a hostname or a range of IP addresses.

Associate Users with a Network Object

To associate a single user or a group of users in a user list with a network object, perform the following steps:

Procedure

Associate a single user or a group of users in a user list with a network object.

snmp-server host-group *net_obj_name* [**trap**| **poll**] [**community** *community-string*] [**version** {**1** | **2c** | **3** {*username* | **user-list** *list_name*}] [**udp-port** *port*]

Example:

```
ciscoasa(config)# snmp-server host-group inside net1 trap community public version 1
ciscoasa(config)# snmp-server host-group inside net1 trap community public version 2c
ciscoasa(config)# snmp-server host-group inside net1 trap version 3 user1
ciscoasa(config)# snmp-server host-group inside net1 trap version 3 user-list engineering
```

The *net_obj_name* argument specifies the interface network object name with which a user or group of users is associated.

The **trap** keyword specifies that only traps can be sent, and that this host is not allowed to browse (poll). SNMP traps are enabled by default.

The **poll** keyword specifies that the host is allowed to browse (poll), but no traps can be sent.

The **community** keyword specifies that a non-default string is required for requests from the NMS, or when generating traps sent to the NMS. You can use this keyword only for SNMP Version 1 or 2c. The

community-string argument specifies the password-like community string that is sent with the notification or in a request from the NMS. The community string can have a maximum of 32 characters.

The **version** keyword sets the SNMP notification version to Version 1, 2c, or 3 to use for sending traps and accepting requests (polling). The default version is 1.

The *username* argument specifies the name of the user if you are using SNMP Version 3.

The **user-list** *list_name* keyword-argument pair specifies the name of the user list.

The **udp-port** *port* keyword-argument pair specifies that SNMP traps must be sent to an NMS host on a non-default port and sets the UDP port number of the NMS host. The default UDP port is 162.

Monitoring SNMP

See the following commands for monitoring SNMP.

- **show running-config snmp-server** [**default**]

  This command shows all SNMP server configuration information.

- **show running-config snmp-server group**

  This command shows SNMP group configuration settings.

- **show running-config snmp-server host**

  This command shows configuration settings used by SNMP to control messages and notifications sent to remote hosts.

- **show running-config snmp-server host-group**

  This command shows SNMP host group configurations.

- **show running-config snmp-server user**

  This command shows SNMP user-based configuration settings.

- **show running-config snmp-server user-list**

  This command shows SNMP user list configurations.

- **show snmp-server engineid**

  This command shows the ID of the SNMP engine configured.

- **show snmp-server group**

  This command shows the names of configured SNMP groups. If the community string has already been configured, two extra groups appear by default in the output. This behavior is normal.

- **show snmp-server statistics**

  This command shows the configured characteristics of the SNMP server. To reset all SNMP counters to zero, use the **clear snmp-server statistics** command.

- **show snmp-server user**

  This command shows the configured characteristics of users.

Examples

The following example shows how to display SNMP server statistics:

```
ciscoasa(config)# show snmp-server statistics
0 SNMP packets input
    0 Bad SNMP version errors
    0 Unknown community name
    0 Illegal operation for community name supplied
    0 Encoding errors
    0 Number of requested variables
    0 Number of altered variables
    0 Get-request PDUs
    0 Get-next PDUs
    0 Get-bulk PDUs
    0 Set-request PDUs (Not supported)
0 SNMP packets output
    0 Too big errors (Maximum packet size 512)
    0 No such name errors
    0 Bad values errors
    0 General errors
    0 Response PDUs
    0 Trap PDUs
```

The following example shows how to display the SNMP server running configuration:

```
ciscoasa(config)# show running-config snmp-server
no snmp-server location
no snmp-server contact
snmp-server enable traps snmp authentication linkup linkdown coldstart
```

Examples for SNMP

The following section provides examples that you can use as reference for all SNMP versions.

SNMP Versions 1 and 2c

The following example shows how the ASA can receive SNMP requests from host 192.0.2.5 on the inside interface but does not send any SNMP syslog requests to any host:

```
ciscoasa(config)# snmp-server host 192.0.2.5
ciscoasa(config)# snmp-server location building 42
ciscoasa(config)# snmp-server contact EmployeeA
ciscoasa(config)# snmp-server community ohwhatakeyisthee
```

SNMP Version 3

The following example shows how the ASA can receive SNMP requests using the SNMP Version 3 security model, which requires that the configuration follow this specific order: group, followed by user, followed by host:

```
ciscoasa(config)# snmp-server group v3 vpn-group priv
ciscoasa(config)# snmp-server user admin vpn group v3 auth sha letmein priv 3des cisco123
ciscoasa(config)# snmp-server host mgmt 10.0.0.1 version 3 priv admin
```

History for SNMP

Table 67: History for SNMP

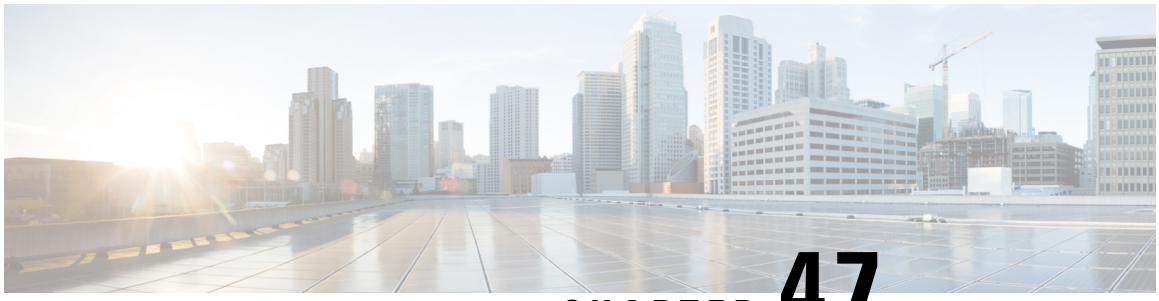
Feature Name	Version	Description
SNMP Versions 1 and 2c	7.0(1)	Provides ASA network monitoring and event information by transmitting data between the SNMP server and SNMP agent through the clear text community string.
SNMP Version 3	8.2(1)	Provides 3DES or AES encryption and support for SNMP Version 3, the most secure form of the supported security models. This version allows you to configure users, groups, and hosts, as well as authentication characteristics by using the USM. In addition, this version allows access control to the agent and MIB objects and includes additional MIB support. We introduced or modified the following commands: **show snmp-server engineid**, **show snmp-server group**, **show snmp-server user**, **snmp-server group**, **snmp-server user**, **snmp-server host**.
Password encryption	8.3(1)	Supports password encryption. We modified the following commands: **snmp-server community**, **snmp-server host**.
SNMP traps and MIBs	8.4(1)	Supports the following additional keywords: **connection-limit-reached**, **cpu threshold rising**, **entity cpu-temperature**, **entity fan-failure**, **entity power-supply**, **ikev2 stop | start**, **interface-threshold**, **memory-threshold**, **nat packet-discard**, **warmstart**. The entPhysicalTable reports entries for sensors, fans, power supplies, and related components. Supports the following additional MIBs: CISCO-ENTITY-SENSOR-EXT-MIB, CISCO-ENTITY-FRU-CONTROL-MIB, CISCO-PROCESS-MIB, CISCO-ENHANCED-MEMPOOL-MIB, CISCO-L4L7MODULE-RESOURCE-LIMIT-MIB, DISMAN-EVENT-MIB, DISMAN-EXPRESSION-MIB, ENTITY-SENSOR-MIB, NAT-MIB. Supports the following additional traps: ceSensorExtThresholdNotification, clrResourceLimitReached, cpmCPURisingThreshold, mteTriggerFired, natPacketDiscard, warmStart. We introduced or modified the following commands: **snmp cpu threshold rising**, **snmp interface threshold**, **snmp-server enable traps**.
IF-MIB ifAlias OID support	8.2(5) / 8.4(2)	The ASA now supports the ifAlias OID. When you browse the IF-MIB, the ifAlias OID will be set to the value that has been set for the interface description.

Feature Name	Version	Description
ASA Services Module (ASASM)	8.5(1)	The ASASM supports all MIBs and traps that are present in 8.4(1), except for the following: Unsupported MIBs in 8.5(1): • CISCO-ENTITY-SENSOR-EXT-MIB (Only objects under the entPhySensorTable group are supported). • ENTITY-SENSOR-MIB (Only objects in the entPhySensorTable group are supported). • DISMAN-EXPRESSION-MIB (Only objects in the expExpressionTable, expObjectTable, and expValueTable groups are supported). Unsupported traps in 8.5(1): • ceSensorExtThresholdNotification (CISCO-ENTITY-SENSOR-EXT-MIB). This trap is only used for power supply failure, fan failure, and high CPU temperature events. • InterfacesBandwidthUtilization.
SNMP traps	8.6(1)	Supports the following additional keywords for the ASA 5512-X, 5515-X, 5525-X, 5545-X, and 5555-X: **entity power-supply-presence**, **entity power-supply-failure**, **entity chassis-temperature**, **entity chassis-fan-failure, entity power-supply-temperature**. We modified the following command: **snmp-server enable traps**.
VPN-related MIBs	9.0(1)	An updated version of the CISCO-IPSEC-FLOW-MONITOR-MIB.my MIB has been implemented to support the next generation encryption feature. The following MIBs have been enabled for the ASASM: • ALTIGA-GLOBAL-REG.my • ALTIGA-LBSSF-STATS-MIB.my • ALTIGA-MIB.my • ALTIGA-SSL-STATS-MIB.my • CISCO-IPSEC-FLOW-MONITOR-MIB.my • CISCO-REMOTE-ACCESS-MONITOR-MIB.my
Cisco TrustSec MIB	9.0(1)	Support for the following MIB was added: CISCO-TRUSTSEC-SXP-MIB.
SNMP OIDs	9.1(1)	Five new SNMP Physical Vendor Type OIDs have been added to support the ASA 5512-X, 5515-X, 5525-X, 5545-X, and 5555-X.
NAT MIB	9.1(2)	Added the cnatAddrBindNumberOfEntries and cnatAddrBindSessionCount OIDs to support the xlate_count and max_xlate_count entries, which are the equivalent to allowing polling using the **show xlate count** command.

Feature Name	Version	Description
SNMP hosts, host groups, and user lists	9.1(5)	You can now add up to 4000 hosts. The number of supported active polling destinations is 128. You can specify a network object to indicate the individual hosts that you want to add as a host group. You can associate more than one user with one host. We introduced or modified the following commands: **snmp-server host-group**, **snmp-server user-list**, **show running-config snmp-server**, **clear configure snmp-server**.
SNMP message size	9.2(1)	The limit on the message size that SNMP sends has been increased to 1472 bytes.
SNMP OIDs and MIBs	9.2(1)	The ASA now supports the cpmCPUTotal5minRev OID. The ASAv has been added as a new product to the SNMP sysObjectID OID and entPhysicalVendorType OID. The CISCO-PRODUCTS-MIB and CISCO-ENTITY-VENDORTYPE-OID-MIB have been updated to support the new ASAv platform. A new SNMP MIB for monitoring VPN shared license usage has been added.
SNMP OIDs and MIBs	9.3(1)	CISCO-REMOTE-ACCESS-MONITOR-MIB (OID 1.3.6.1.4.1.9.9.392) support has been added for the ASASM.
SNMP MIBs and traps	9.3(2)	The CISCO-PRODUCTS-MIB and CISCO-ENTITY-VENDORTYPE-OID-MIB have been updated to support the ASA 5506-X. The ASA 5506-X has been added as new products to the SNMP sysObjectID OID and entPhysicalVendorType OID tables. The ASA now supports the CISCO-CONFIG-MAN-MIB, which enables you to do the following: • Know which commands have been entered for a specific configuration. • Notify the NMS when a change has occurred in the running configuration. • Track the time stamps associated with the last time that the running configuration was changed or saved. • Track other changes to commands, such as terminal details and command sources. We modified the following command: **snmp-server enable traps**.
SNMP MIBs and traps	9.4(1)	The ASA 5506W-X, ASA 5506H-X, ASA 5508-X, and ASA 5516-X have been added as a new product to the SNMP sysObjectID OID and entPhysicalVendorType OID tables.

Feature Name	Version	Description
Unlimited SNMP server trap hosts per context	9.4(1)	The ASA supports unlimited SNMP server trap hosts per context. The **show snmp-server host** command output displays only the active hosts that are polling the ASA, as well as the statically configured hosts. We modified the following command: **show snmp-server host**.
Added support for ISA 3000	9.4(1.225)	The ISA 3000 family of products is now supported for SNMP. We added new OIDs for this platform. The **snmp-server enable traps entity** command has been modified to include a new variable *l1-bypass-status*. This enables hardware bypass status change. We modified the following command: **snmp-server enable traps entity**.
Support for the cempMemPoolTable in the CISCO-ENHANCED-MEMPOOL-MIB	9.6(1)	The cempMemPoolTable of the CISCO-ENHANCED-MEMPOOL-MIB is now supported. This is a table of memory pool monitoring entries for all physical entities on a managed system. **Note** The CISCO-ENHANCED-MEMPOOL-MIB uses 64-bit counters and supports reporting of memory on platforms with more than 4GB of RAM.
Support for E2E Transparent Clock Mode MIBs for the Precision Time Protocol (PTP)	9.7(1)	MIBs corresponding to E2E Transparent Clock mode are now supported. **Note** Only SNMP get, bulkget, getnext, and walk operations are supported.
SNMP over IPv6	9.9(2)	The ASA now supports SNMP over IPv6, including communicating with SNMP servers over IPv6, allowing the execution of queries and traps over IPv6, and supporting IPv6 addresses for existing MIBs. We added the following new SNMP IPv6 MIB objects as described in RFC 8096. • ipv6InterfaceTable (OID: 1.3.6.1.2.1.4.30)—Contains per-interface IPv6-specific information. • ipAddressPrefixTable (OID:1.3.6.1.2.1.4.32)—Includes all the prefixes learned by this entity. • ipAddressTable (OID: 1.3.6.1.2.1.4.34)—Contains addressing information relevant to the entity's interfaces. • ipNetToPhysicalTable (OID: 1.3.6.1.2.1.4.35)—Contains the mapping from IP addresses to physical addresses. New or modified command: **snmp-server host** **Note** The **snmp-server host-group** command does not support IPv6.
Support to enable and disable the results for free memory and used memory statistics during SNMP walk operations	9.10(1)	To avoid overutilization of CPU resources, you can enable and disable the query of free memory and used memory statistics collected through SNMP walk operations. New or modified command: **snmp-server enable oid**

Feature Name	Version	Description
Support to enable and disable the results for free memory and used memory statistics during SNMP walk operations	9.12(1)	To avoid overutilization of CPU resources, you can enable and disable the query of free memory and used memory statistics collected through SNMP walk operations. We did not modify any commands.
SNMPv3 Authentication	9.14(1)	You can now use SHA-256 HMAC for user authentication. New/Modified commands: **snmp-server user**
For Failover pairs in 9.14(1)+, the ASA no longer shares SNMP client engine data with its peer.	9.14(1)	The ASA no longer shares SNMP client engine data with its peer.
SNMP polling over site-to-site VPN	9.14(2)	For secure SNMP polling over a site-to-site VPN, include the IP address of the outside interface in the crypto map access-list as part of the VPN configuration.
Support for the CISCO-MEMORY-POOL-MIB OIDs is deprecated	9.15(1)	The CISCO-MEMORY-POOL-MIB OIDs (ciscoMemoryPoolUsed, ciscoMemoryPoolFree) are deprecated for systems that use 64-bit counters. The cempMemPoolTable of the CISCO-ENHANCED-MEMPOOL-MIB provides memory pool monitoring entries for systems that use 64-bit counters.
SNMPv3 Authentication	9.16(1)	You can now use SHA-224 and SHA-384 for user authentication. You can no longer use MD5 for user authentication. You can no longer use DES for encryption. New/Modified commands: **snmp-server user**
SNMP over IPv6	9.17(1)	The **snmp-server host-group** command now supports IPv6 host, range, and subnet objects.



CHAPTER 47

Cisco Success Network and Telemetry Data

This chapter describes about Cisco Success Network and how to enable it on ASA. It also lists the telemetry data points that are sent to the Security Service Engine(SSE) cloud.

- About Cisco Success Network , on page 1365
- Enable or Disable Cisco Success Network , on page 1366
- View ASA Telemetry Data , on page 1367
- Cisco Success Network - Telemetry Data, on page 1367
- Debug Telemetry Data, on page 1373

About Cisco Success Network

Cisco Success Network is user-enabled cloud service that establishes a secured connection with the Security Service Exchange (SSE) cloud to stream ASA usage information and statistics. Streaming telemetry provides a mechanism to transmit ASA usage and other details in structured format (JSON) to remote management stations for the following benefits:

- To inform you of extra technical support services and monitoring that are available for your product.
- To help Cisco improve its products.

By default, the Cisco Success Network is enabled on the Firepower 4100/9300 platforms that hosts ASA devices (at the blade level). However, for the telemetry data to be transmitted, you must enable the configuration on FXOS at chassis level (see Cisco Firepower 4100/9300 FXOS CLI Configuration Guide) or enable the Cisco Success Network on the chassis manager (see Cisco Firepower 4100/9300 FXOS Firepower Chassis Manager Configuration Guide) ASA allows you to disable the telemetry service at any point in time.

The telemetry data that is collected on your ASA devices includes CPU, memory, disk, bandwidth, and license usage, configured feature list, cluster/failover information, and the alike. Refer Cisco Success Network - Telemetry Data, on page 1367.

Supported Platforms and Required Configurations

- Supported on FP9300/4100 platforms with ASA version 9.13.1 or above running on it.
- Requires FXOS version 2.7.1 or above to connect with the cloud.
- The SSE connector on FXOS must be connected to the SSE cloud. This connection is established by enabling and registering the smart license with smart licensing backend. The SSE connector on FXOS is automatically registered to the SSE cloud by registering smart license.

- The Cisco Success Network configuration must be enabled on chassis manager.
- The telemetry configuration must be enabled on ASA.

How Does ASA Telemetry Data Reach the SSE Cloud

Cisco Success Network is supported on Firepower 4100/9300 platforms in ASA 9.13(1) by default. The FXOS service manager sends telemetry request daily to the ASA application running on the platform. The ASA engine, based on the configuration and connectivity status, sends the telemetry data either in standalone mode or cluster mode to FXOS. That is, if the telemetry support is enabled in ASA and SSE connector status is connected, the telemetry thread pulls the needed information from various sources such as system or platform or device APIs, license APIs, CPU APIs, memory APIs, disk APIs, smart call home feature APIs, and so on. However, if the telemetry support is disabled in ASA or the SSE connector status is disconnected, ASA sends a response to FXOS (appAgent) indicating the telemetry configuration status and does not send any telemetry data.

FXOS has only one SSE connector instance running on it. When it gets registered with the SSE cloud, it is considered as one device and SSE infra assigns FXOS with one device ID. Any telemetry report that is sent through the SSE connector is categorized under the same device ID. Therefore, FXOS aggregates the telemetry report from each ASA into a single report. Other contents such as smart license account information are added to the report. FXOS then sends the final report to the SSE cloud. The telemetry data is saved in the SSE data exchange (DEX) and available for the Cisco IT team to use.

Enable or Disable Cisco Success Network

Before you begin

- Enable and register smart license on FXOS.
- Enable telemetry support on FXOS at the chassis level (see Cisco Firepower 4100/9300 FXOS CLI Configuration Guide) or enable the Cisco Success Network on the chassis manager (see Cisco Firepower 4100/9300 FXOS Firepower Chassis Manager Configuration Guide).

Procedure

To enable the telemetry service on ASA, in the global configuration mode, enter the following commnad. Use the no form of the command to disable the telemetry service:

[**no**] **service telemetry**

Example:

```
ciscoasa(config)# service telemetry
ciscoasa(config)# no service telemetry
```

What to do next

- You can view the telemetry configuration and activities log or the telemetry data. See View ASA Telemetry Data , on page 1367
- To view a sample of telemetry data and the data fields, see Cisco Success Network - Telemetry Data, on page 1367

View ASA Telemetry Data

Before you begin

- Enable the telemetry service on ASA. See Enable or Disable Cisco Success Network , on page 1366

Procedure

To view the telemetry data on ASA devices of your network, enter the following commnad in the privileged EXEC mode:

show telemetry [**history** | **last-report** | **sample**]

Example:

```
ciscoasa# show telemetry history
17:38:24 PDT Apr 30 2019: Telemetry support on the blade: enabled
17:38:03 PDT Apr 30 2019: Telemetry support on the blade: disabled
11:49:47 PDT Apr 29 2019: msgId 3. Telemetry support on the chassis: disabled
11:48:47 PDT Apr 29 2019: msgId 2. Telemetry request from the chassis received. SSE connector
 status: enabled. Telemetry config on the blade: enabled. Telemetry data Sent
11:47:47 PDT Apr 29 2019: msgId 1. Telemetry request from the chassis received. SSE connector
 status: enabled. Telemetry config on the blade: enabled. Telemetry data Sent.
```

Use **history** to view the past 100 events that are related to telemetry configuration and activities; **last-report** to view the latest telemetry data that are sent to FXOS in JSON format, and **sample** to view the instantly generated telemetry data in JSON format.

Cisco Success Network - Telemetry Data

Cisco Success Network is supported on Firepower 4100/9300 platforms by default. The FXOS service manager sends telemetry request daily to the ASA engine running on the platform. The ASA engine, on receiving the request, based on the connectivity status, sends the telemetry data either in standalone mode or cluster mode to FXOS. Following tables provide information on the telemetry data points, its description, and sample values.

Table 68: Device Info

Data Point	Description	Example Value
Device Model	Device model	Cisco Adaptive Security Appliance

Data Point	Description	Example Value
Serial Number	Serial number of the device	FCH183771EZ
System Time	System uptime	11658000
Platform	Hardware	FPR9K-SM-24
Deployment Mode	Deployment type	Native
Security context mode	Single/Multiple	Single

Table 69: Versions Info

Data Point	Description	Example Value
Version Global Variable	ASA version	9.13.1.5
Device Manager Version	Device manager version	7.10.1

Table 70: License Info

Data Point	Description	Example Value
Smart License Global Variable	Activated licenses	regid.2015-01.com.cisco.ASA - SSP-STRONG-ENCRYPTION, 1.0_555507e9-85f8-4e41-96de-860b59f10bbe

Table 71: Platform Info

Data Point	Description	Example Value
CPU	CPU usage in past 5 minutes	fiveSecondsPercentage: 0.2000000, oneMinutePercentage: 0, fiveMinutesPercentage: 0
Memory	Memory usage	freeMemoryInBytes: 225854966384, usedMemoryInBytes: 17798281616, totalMemoryInBytes: 243653248000
Disk	Disk usage	freeGB: 21.237285, usedGB: 0.238805, totalGB: 21.476090

Data Point	Description	Example Value
Bandwidth	Bandwidth usage	receivedPktsPerSec: 3, receivedBytesPerSec: 212, transmittedPktsPerSec: 3, transmittedBytesPerSec: 399

Table 72: Feature Info

Data Point	Description	Example Value
Feature List	Enabled feature list	name: cluster status: enabled

Table 73: Cluster Info

Data Point	Description	Example Value
Cluster Info	Cluster information	clusterGroupName : ssp-cluster interfaceMode : spanned unitName : unit-3-3 unitState : SLAVE otherMembers : items : memberName : unit-2-1 memberState : MASTER memberSerialNum : FCH183771BA

Table 74: Failover Info

Data Point	Description	Example Value
Failover	Failover information	myRole: Primary, peerRole: Secondary, myState: active, peerState: standby, peerSerialNum: FCH183770EZ

Table 75: Login Info

Data Point	Description	Example Value
Login	Login history	loginTimes: 2 times in last 2 days, lastSuccessfulLogin: 12:25:36 PDT Mar 11 2019

ASA Telemetry Data Sample

Following is an example of the telemetry data that are sent from ASA in JSON format. When service manager receives this input, it aggregates the data from all ASAs and adds necessary headers/fields before sending to the SSE connector. The headers/fields include “version”, “metadata”, “payload” with “recordedAt”, “recordType”, “recordVersion”, and ASA telemetry data, "smartLicenseProductInstanceIdentifier", "smartLicenseVirtualAccountName", and alike.


```
{
  "version": "1.0",
  "metadata": {
    "topic": "ASA.telemetry",
    "contentType": "application/json"
  },
  "payload": {
    "recordType": "CST_ASA",
    "recordVersion": "1.0",
    "recordedAt": 1557363423705,
    "SSP": {
      "SSPdeviceInfo": {
        "deviceModel": "Cisco Firepower FP9300 Security Appliance",
        "serialNumber": "JMX2235L01J",
        "smartLicenseProductInstanceIdentifier": "f85a5bb0-xxxx-xxxx-xxxx-xxxxxxxxx",
        "smartLicenseVirtualAccountName": "SSP-general",
        "systemUptime": 198599,
        "udiProductIdentifier": "FPR-C9300-AC"
      },
      "versions": {
        "items": [
          {
            "type": "package_version",
            "version": "92.7(1.342g)"
          }
        ]
      }
    },
    "asaDevices": {
      "items": [
        {
    "deviceInfo": {
        "deviceModel": "Cisco Adaptive Security Appliance",
        "serialNumber": "AANNNXXX",
        "systemUptime": 285,
        "udiProductIdentifier": "FPR9K-SM-36",
        "deploymentType": "Native",
        "securityContextMode": "Single"
    },
    "versions": {
        "items": [
            {
                "type": "asa_version",
                "version": "201.4(1)82"
```

```
            },
            {
                "type": "device_mgr_version",
                "version": "7.12(1)44"
            }
        ]
    },
    "licenseActivated": {
        "items": [
            {
                "type": "Strong encryption",
                "tag":
"regid.2015-01.com.cisco.ASA-SSP-STRONG-ENCRYPTION,1.0_xxxxxxx-xxxx-xxxx-96de-860b59f10bbe",

                "count": 1
            },
            {
                "type": "Carrier",
                "tag":
"regid.2015-01.com.cisco.ASA-SSP-MOBILE-SP,1.0_xxxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxx",
                "count": 1
            }
        ]
    },
    "CPUUsage": {
        "fiveSecondsPercentage": 0,
        "oneMinutePercentage": 0,
        "fiveMinutesPercentage": 0
    },
    "memoryUsage": {
        "freeMemoryInBytes": 99545662064,
        "usedMemoryInBytes": 20545378704,
        "totalMemoryInBytes": 120091040768
    },
    "diskUsage": {
        "freeGB": 21.237027,
        "usedGB": 0.239063,
        "totalGB": 21.476090
    },
    "bandwidthUsage": {
        "receivedPktsPerSec": 3,
        "receivedBytesPerSec": 268,
        "transmittedPktsPerSec": 4,
        "transmittedBytesPerSec": 461
    },
    "featureStatus": {
        "items": [
            {
                "name": "call-home",
                "status": "enabled"
            },
            {
                "name": "cluster",
                "status": "enabled"
            },
            {
                "name": "firewall_user_authentication",
                "status": "enabled"
            },
            {
                "name": "inspection-dns",
                "status": "enabled"
            },
            {
```

```
            "name": "inspection-esmtp",
            "status": "enabled"
        },
        {
            "name": "inspection-ftp",
            "status": "enabled"
        },
        {
            "name": "inspection-netbios",
            "status": "enabled"
        },
        {
            "name": "inspection-rsh",
            "status": "enabled"
        },
        {
            "name": "inspection-sip",
            "status": "enabled"
        },
        {
            "name": "inspection-sqlnet",
            "status": "enabled"
        },
        {
            "name": "inspection-sunrpc",
            "status": "enabled"
        },
        {
            "name": "inspection-tftp",
            "status": "enabled"
        },
        {
            "name": "inspection-xdmcp",
            "status": "enabled"
        },
        {
            "name": "logging-console",
            "status": "informational"
        },
        {
            "name": "management-mode",
            "status": "normal"
        },
        {
            "name": "sctp-engine",
            "status": "enabled"
        },
        {
            "name": "threat_detection_basic_threat",
            "status": "enabled"
        },
        {
            "name": "threat_detection_stat_access_list",
            "status": "enabled"
        },
        {
            "name": "webvpn-activex-relay",
            "status": "enabled"
        },
        {
            "name": "webvpn-dtls",
            "status": "enabled"
        }
    ]
```

```
    },
    "clusterInfo": {
        "clusterGroupName": "ssp-cluster",
        "interfaceMode": "spanned",
        "unitName": "unit-3-3",
        "unitState": "SLAVE",
        "otherMembers": {
            "items": [
                {
                    "memberName": "unit-2-1",
                    "memberState": "MASTER",
                    "memberSerialNum": "FCH183771BA"
                },
                {
                    "memberName": "unit-2-3",
                    "memberState": "SLAVE",
                    "memberSerialNum": "FLM1949C6JR"
                },
                {
                    "memberName": "unit-2-2",
                    "memberState": "SLAVE",
                    "memberSerialNum": "xxxxxxxx"
                },
                {
                    "memberName": "unit-3-2",
                    "memberState": "SLAVE",
                    "memberSerialNum": "xxxxxxxx"
                },
                {
                    "memberName": "unit-3-1",
                    "memberState": "SLAVE",
                    "memberSerialNum": "xxxxxxxx"
                }
            ]
        }
    },
    "loginHistory": {
        "loginTimes": "1 times in last 1 days",
        "lastSuccessfulLogin": "12:25:36 PDT Mar 11 2019"
    }
}
```


Debug Telemetry Data

Before you begin

- Enable the telemetry service on ASA. See Enable or Disable Cisco Success Network , on page 1366

Procedure

Step 1 To view the debug messages related to telemetry, enable the debug telemetry service using the following command in the privileged EXEC mode:

debug telemetry *<1-255>*

Example:

```
asa# debug telemetry ?
   <1-255>  Specify an optional debug level (default is 1)
```

Use the **no** form of the command to disable the debug telemetry service.

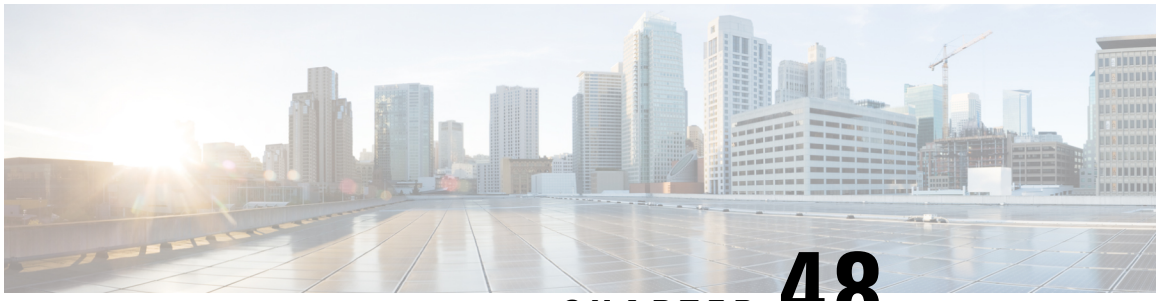
Step 2 To view the debug telemetry messages for the selected debug level, use the following command:

show debug telemetry

Example:

```
asa# show debug telemetry
debug telemetry  enabled at level 1

[telemetry_collect_device_info]: telemetry successfully collected device info
[telemetry_collect_versions]: telemetry successfully collected version info
[telemetry_collect_licenses]: no smart-lic entitlement in use
[telemetry_collect_cpu]: telemetry successfully collected cpu info
[telemetry_collect_memory]: telemetry successfully collected mem info
[telemetry_collect_disk_usage]: telemetry successfully collected disk info
[telemetry_collect_bandwidth_usage]: telemetry successfully collected bandwidth usage info
[telemetry_collect_enabled_feature_status]: telemetry successfully collected enabled feature
 info
[telemetry_collect_cluster_info]: telemetry successfully collected cluster info
[telemetry_collect_failover_info]: ha is not configured
[telemetry_get_user_login_hist]: telemetry successfully collected login history
[telemetry_collect_blocks]: telemetry successfully collected block info
[telemetry_collect_perfmon]: telemetry successfully collected perfmon stats
[telemetry_collect_resource_usage]: telemetry successfully collected res usage
[telemetry_collect_process_cpu_usage]: telemetry successfully collected res usage
[telemetry_collect_crashinfo]: telemetry successfully collected crashinfo
[telemetry_collect]: the serialized string is generated
[telemetry_collect]: successfully allocated mem for serialized string
[telemetry_history_add_record]: telemetry has a new history record: 16:23:29 PDT Oct 22
2019: Telemetry support on the blade: enabled
[telemetry_history_add_record]: telemetry has a new history record: 16:24:01 PDT Oct 22
2019: Telemetry support on the blade: disabled
```



CHAPTER 48

Alarms for the Cisco ISA 3000

This chapter gives an overview of the alarm system in the ISA 3000, and also describes how to configure and monitor alarms.

- About alarms, on page 1375
- Defaults for alarms, on page 1377
- Configure Alarms, on page 1377
- Monitoring Alarms, on page 1380
- History for Alarms, on page 1382

About alarms

You can configure the ISA 3000 to issue alarms for a variety of conditions. If any conditions do not match the configured settings, the system triggers an alarm, which is reported by way of LEDs, syslog messages, SNMP traps, and through external devices connected to the alarm output interface. By default, triggered alarms issue syslog messages only.

You can configure the alarm system to monitor the following:

- Power supply.
- Primary and secondary temperature sensors.
- Alarm input interfaces.

The ISA 3000 has internal sensors plus two alarm input interfaces and one alarm output interface. You can connect external sensors, such as door sensors, to the alarm inputs. You can connect external alarm devices, such as buzzers or lights, to the alarm output interface.

The alarm output interface is a relay mechanism. Depending on the alarm conditions, the relay is either energized or de-energized. When it is energized, any device connected to the interface is activated. A de-energized relay results in the inactive state of any connected devices. The relay remains in an energized state as long as alarms are triggered.

For information about connecting external sensors and the alarm relay, Refer to Alarms for the Cisco ISA 3000 chapter in Cisco ASA Series General Operations CLI Configuration Guide. see Cisco ISA 3000 Industrial Security Appliance Hardware Installation Guide.

Alarm input interfaces

You can connect the alarm input interfaces (or contacts) to external sensors, such as one that detects if a door is open.

Each alarm input interface has a corresponding LED. These LEDs convey the alarm status of each alarm input. You can configure the trigger and severity for each alarm input. In addition to the LED, you can configure the contact to trigger the output relay (to activate an external alarm), to send syslog messages, and to send SNMP traps.

The following table explains the statuses of the LEDs in response to alarm conditions for the alarm inputs. It also explains the behavior for the output relay, syslog messages, and SNMP traps, if you enable these responses to the alarm input.

Alarm Status	LED	Output Relay	Syslog	SNMP Trap
Alarm not configured	Off	—	—	—
No alarms triggered	Solid green	—	—	—
Alarm activated	Minor alarm—solid red Major alarm—flashing red	Relay energized	Syslog generated	SNMP trap sent
Alarm end	Solid green	Relay de-energized	Syslog generated	—

Alarm output interface

You can connect an external alarm, such as a buzzer or light, to the alarm output interface.

The alarm output interface functions as a relay and also has a corresponding LED, which conveys the alarm status of an external sensor connected to the input interface, and internal sensors such as the dual power supply and temperature sensors. You configure which alarms should activate the output relay, if any.

The following table explains the statuses of the LEDs and output relay in response to alarm conditions. It also explains the behavior for syslog messages, and SNMP traps, if you enable these responses to the alarm.

Alarm Status	LED	Output Relay	Syslog	SNMP Trap
Alarm not configured	Off	—	—	—
No alarms triggered	Solid green	—	—	—
Alarm activated	Solid red	Relay energized	Syslog generated	SNMP trap sent
Alarm end	Solid green	Relay de-energized	Syslog generated	—

Defaults for alarms

The following table specifies the defaults for alarm input interfaces (contacts), redundant power supply, and temperature.

	Alarm	Trigger	Severity	SNMP Trap	Output Relay	Syslog Message
Alarm Contact 1	Enabled	Closed State	Minor	Disabled	Disabled	Enabled
Alarm Contact 2	Enabled	Closed State	Minor	Disabled	Disabled	Enabled
Redundant Power Supply (when enabled)	Enabled	—	—	Disabled	Disabled	Enabled
Temperature	Enabled for the primary temperature alarm (default values of 92°C and -40°C for the high and low thresholds respectively) Disabled for the secondary alarm.	—	—	Enabled for primary temperature alarm	Enabled for primary temperature alarm	Enabled for primary temperature alarm

Configure Alarms

To configure alarms for the ISA 3000, perform the following steps.

Procedure

Step 1 Configure severity for one or all alarm contacts.

alarm contact {*contact_number* | **all**} **severity {major | minor | none}**

Example:

```
ciscoasa(config)# alarm contact 1 severity major
```

Enter a contact number (**1** or **2**) or enter **all** to configure all alarms. Enter **major**, **minor** or **none** as the severity. The default is minor.

Step 2 Configure triggers for one or all alarm contacts.

alarm contact {*contact_number* | **all**} **trigger {closed | open}**

Specifying **open** will trigger an alarm when a contact which is normally closed (normal electrical connectivity), is open, or when current stops flowing.

Specifying **closed** will trigger an alarm when the contact which is normally open (no electrical connectivity), is closed, or when current starts flowing.

For example, if a door sensor is connected to an alarm input, its normally open state has no electrical current flowing through the contacts. If the door is opened, electrical current flows through the contacts, activating the alarm.

Example:

```
ciscoasa(config)# alarm contact 1 trigger open
```

Enter a contact number (**1** or **2**) or enter **all** to configure all alarms. Enter **open** or **closed** to specify the trigger. The default is closed.

Step 3 Enable relay, system logger and SNMP traps for alarm contacts.

When the relay is enabled, and an alarm condition arises, the relay is energized and the device attached to the relay is activated. When the relay is energized, the alarm out LED glows solid red.

- Enable relay for the input alarm.

alarm facility input-alarm *contact_number* **relay**

Example:

```
ciscoasa(config)# alarm facility input-alarm 1 relay
```

Enter a contact number (**1** or **2**). By default, relay for alarm inputs is disabled.

- Enable system logger.

alarm facility input-alarm *contact_number* **syslog**

Example:

```
ciscoasa(config)# alarm facility input-alarm 1 syslog
```

Enter a contact number (**1** or **2**).

- Enable SNMP traps.

alarm facility input-alarm *contact_number* **notifies**

Example:

```
ciscoasa(config)# alarm facility input-alarm 1 notifies
```

Enter a contact number (**1** or **2**).

Step 4 (Optional) Specify a description for input alarm contacts.

alarm contact *contact_number* | **description** *string*

Example:

```
ciscoasa(config)# alarm contact 1 description Door_Open
```

The contact_number specifies the alarm contact for which the description is configured. The description may be up to 80 alphanumeric characters long and will be included in syslog messages.

To set the default description to the corresponding contact number use the **no alarm contact contact_number description** command.

Step 5 Configure power supply alarms.

Note
Redundant power supply has to be enabled for the power supply alarms to work.

See the following commands for configuring power supply alarms:

- **power-supply dual**

  This command enables dual power supply.

- **alarm facility power-supply rps disable**

  This command disables the power supply alarm. In its default state, this alarm is disabled. If the alarm is enabled, use this command to disable it.

- **alarm facility power-supply rps notifies**

  This command sends power supply alarm traps to an SNMP server.

- **alarm facility power-supply rps relay**

  This command associates the power supply alarm to the relay.

- **alarm facility power-supply rps syslog**

  This command sends power supply alarm traps to a syslog server.

Step 6 Configure temperature thresholds.

alarm facility temperature {**primary | secondary**} {**high | low**} *threshold*

Example:

```
ciscoasa(config)# alarm facility temperature primary high 90
ciscoasa(config)# alarm facility temperature primary low 40
ciscoasa(config)# alarm facility temperature secondary high 85
ciscoasa(config)# alarm facility temperature primary low 35
```

For the primary temperature alarm, valid threshold values range from –40°C to 92°C. For the secondary temperature alarm, valid threshold values range from –35°C to 85°C. If a temperature threshold is configured for the secondary alarm, only the secondary alarm will be enabled.

Use the **no** form of each command to disable or revert to default values. Using the **no** form of the commands for the primary alarm will not disable the alarm and will revert to the default values of 92°C for the high threshold, and –40°C for the low threshold. Using the **no** form of the command for the secondary alarm will disable it.

Step 7 Enable SNMP traps, relay and system logger for temperature alarms.

See the following commands for enabling relay, SNMP traps, and syslogs for temperature alarms:

- **alarm facility temperature** {**primary | secondary**} **notifies**

  This command sends primary or secondary temperature alarm traps to an SNMP server.

- **alarm facility temperature** {**primary | secondary**} **relay**

  This command associates the primary or secondary temperature alarm to the relay.

- **alarm facility temperature** {**primary | secondary**} **syslog**

  This command sends primary or secondary temperature alarm traps to a syslog server.

Use the no form of each command to disable relay, SNMP traps and syslogs.

Monitoring Alarms

See the following commands to monitor alarms:

Procedure

- **show alarm settings**

  This command displays all global alarm settings.

```
ciscoasa> show alarm settings
Power Supply
         Alarm                    Disabled
         Relay                    Disabled
         Notifies                 Disabled
         Syslog                   Disabled
Temperature-Primary
         Alarm                    Enabled
         Thresholds           MAX: 92C                MIN: -40C
         Relay                    Enabled
         Notifies                 Enabled
         Syslog                   Enabled
Temperature-Secondary
         Alarm                    Disabled
         Threshold
         Relay                    Disabled
         Notifies                 Disabled
         Syslog                   Disabled
Input-Alarm 1
         Alarm                    Enabled
         Relay                    Disabled
         Notifies                 Disabled
         Syslog                   Enabled
Input-Alarm 2
         Alarm                    Enabled
         Relay                    Disabled
         Notifies                 Disabled
         Syslog                   Enabled
```

- **show environment alarm-contact**

  This command displays all external alarm settings.

```
ciscoasa> show environment alarm-contact
ALARM CONTACT 1
   Status:       not asserted
   Description:  external alarm contact 1
   Severity:     minor
   Trigger:      closed
ALARM CONTACT 2
   Status:       not asserted
   Description:  external alarm contact 2
```

```
   Severity:     minor
   Trigger:      closed
```

- **show facility-alarm status** [**info** | **major** | **minor**]

  This command displays all alarms based on severity specified.

  The output displays the following information:

Column	Description
Source	Device from which the alarm was triggered. This is usually the hostname configured on the device.
Severity	Major or minor
Description	Type of alarm triggered. For example, temperature, external contact, redundant power supply etc.
Relay	Energized or de-energized
Time	Timestamp of the triggered alarm

```
ciscoasa> show facility-alarm status info
Source      Severity    Description                                  Relay
        Time
ciscoasa   minor        external alarm contact 1 triggered  Energized         06:56:50
 UTC Mon Sep 22 2014
ciscoasa   minor        Temp below Secondary Threshold De-energized           06:56:49
 UTC Mon Sep 22 2014
ciscoasa   major       Redundant pwr missing or failed    De-energized       07:00:19
 UTC Mon Sep 22 2014
ciscoasa   major        Redundant pwr missing or failed    De-energized       07:00:19
 UTC Mon Sep 22 2014

ciscoasa> show facility-alarm status major
Source      Severity    Description                                  Relay
        Time
ciscoasa    major       Redundant pwr missing or failed    De-energized     07:00:19
UTC Mon Sep 22 2014
ciscoasa   major        Redundant pwr missing or failed    De-energized     07:00:19
UTC Mon Sep 22 2014

ciscoasa> show facility-alarm status minor
Source      Severity    Description                                  Relay
        Time
ciscoasa   minor        external alarm contact 1 triggered  Energized         06:56:50
 UTC Mon Sep 22 2014
ciscoasa   minor        Temp below Secondary Threshold De-energized     06:56:49 UTC
Mon Sep 22 2014
```

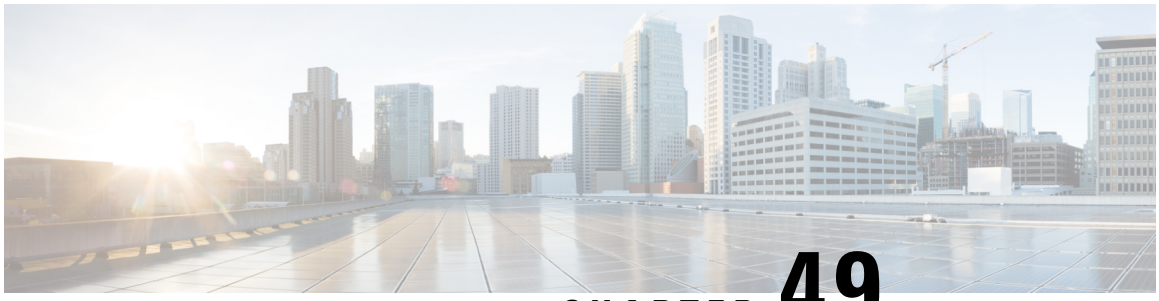
- **show facility-alarm relay**

  This command displays all relays in energized state.

```
ciscoasa> show facility-alarm relay
Source      Severity    Description                                  Relay
        Time
ciscoasa   minor        external alarm contact 1 triggered  Energized         06:56:50
 UTC Mon Sep 22 2014
```

History for Alarms

Feature Name	Platform Releases	Description
Alarm ports support for the ISA 3000	9.7(1)	The ISA 3000 now supports two alarm input pins and one alarm out pin, with LEDs to convey alarms' statuses. External sensors can be connected to the alarm inputs. An external hardware relay can be connected to the alarm out pin. You can configure descriptions of external alarms. You can also specify the severity and trigger, for external and internal alarms. All alarms can be configured for relay, monitoring and logging. We introduced the following commands: **alarm contact description, alarm contact severity, alarm contact trigger, alarm facility input-alarm, alarm facility power-supply rps, alarm facility temperature, alarm facility temperature high, alarm facility temperature low, clear configure alarm, clear facility-alarm output, show alarm settings, show environment alarm-contact**. We introduced the following screens: **Configuration** > **Device Management** > **Alarm Port** > **Alarm Contact** **Configuration** > **Device Management** > **Alarm Port** > **Redundant Power Supply** **Configuration** > **Device Management** > **Alarm Port** > **Temperature** **Monitoring** > **Properties** > **Alarm** > **Alarm Settings** **Monitoring** > **Properties** > **Alarm** > **Alarm Contact** **Monitoring** > **Properties** > **Alarm** > **Facility Alarm Status**



CHAPTER 49

Anonymous Reporting and Smart Call Home

This chapter describes how to configure the Anonymous Reporting and Smart Call Home services.

- About Anonymous Reporting, on page 1383
- About Smart Call Home, on page 1384
- Guidelines for Anonymous Reporting and Smart Call Home, on page 1390
- Configure Anonymous Reporting and Smart Call Home, on page 1391
- Monitoring Anonymous Reporting and Smart Call Home, on page 1402
- Examples for Smart Call Home, on page 1403
- History for Anonymous Reporting and Smart Call Home, on page 1404

About Anonymous Reporting

You can help to improve the ASA platform by enabling Anonymous Reporting, which allows Cisco to securely receive minimal error and health information from the device. If you enable the feature, your customer identity will remain anonymous, and no identifying information will be sent.

Enabling Anonymous Reporting creates a trust point and installs a certificate. A CA certificate is required for your ASA to validate the server certificate present on the Smart Call Home web server and to form the HTTPS session so that your ASA can send messages securely. Cisco imports a certificate that is predefined in the software. If you decide to enable Anonymous Reporting, a certificate is installed on the ASA with a hardcoded trust point name: _SmartCallHome_ServerCA. When you enable Anonymous Reporting, this trust point is created, the appropriate certificate is installed, and you receive a message about this action. The certificate then appears in your configuration.

If the appropriate certificate already exists in your configuration when you enable Anonymous Reporting, no trust point is created, and no certificate is installed.

Note When you enable Anonymous Reporting, you acknowledge your consent to transfer the specified data to Cisco or to vendors operating on Cisco's behalf (including countries outside of the U.S.). Cisco maintains the privacy of all customers. For information about Cisco's treatment of personal information, see the Cisco Privacy Statement at the following URL: http://www.cisco.com/web/siteassets/legal/privacy.html

When the ASA configures Smart Call Home anonymous reporting in the background, the ASA automatically creates a trustpoint containing the certificate of the CA that issues the Call Home server certificate. The ASA now supports validation of the certificate if the issuing hierarchy of the server certificate changes, without the need for customer involvement to make certificate hierarchy changes. You can also automatically import the trustpool certificates so that ASA renews the certificate hierarchy without any manual intervention.

When you upgrade ASA 9.14(2.14), the trust point configuration automatically changes from CallHome_ServerCA to CallHome_ServerCA2.

DNS Requirement

A DNS server must be configured correctly for the ASA to reach the Cisco Smart Call Home server and send messages to Cisco. Because it is possible that the ASA resides in a private network and does not have access to the public network, Cisco verifies your DNS configuration and then configures it for you, if necessary, by doing the following:

1. Performing a DNS lookup for all DNS servers configured.
2. Getting the DNS server from the DHCP server by sending DHCPINFORM messages on the highest security-level interface.
3. Using the Cisco DNS servers for lookup.
4. Randomly using a static IP addresses for tools.cisco.com.

These tasks are performed without changing the current configuration. (For example, the DNS server that was learned from DHCP will not be added to the configuration.)

If there is no DNS server configured, and the ASA cannot reach the Cisco Smart Call Home Server, Cisco generates a syslog message with the warning severity level for each Smart Call Home message that is sent to remind you to configure DNS correctly.

See the syslog messages guide for information about syslog messages.

About Smart Call Home

When fully configured, Smart Call Home detects issues at your site and reports them back to Cisco or through other user-defined channels (such as e-mail or directly to you), often before you know that these issues exist. Depending on the seriousness of these problems, Cisco responds to your system configuration issues, product end-of-life announcements, security advisory issues, and so on by providing the following services:

- Identifying issues quickly with continuous monitoring, real-time proactive alerts, and detailed diagnostics.
- Making you aware of potential problems through Smart Call Home notifications, in which a service request has been opened, with all diagnostic data attached.

- Resolving critical problems faster with direct, automatic access to experts in Cisco TAC.
- Using staff resources more efficiently by reducing troubleshooting time.
- Generating service requests to Cisco TAC automatically (if you have a service contract), routed to the appropriate support team, which provides detailed diagnostic information that speeds problem resolution.

The Smart Call Home Portal offers quick access to required information that enables you to do the following:

- Review all Smart Call Home messages, diagnostics, and recommendations in one place.
- Check service request status.
- View the most up-to-date inventory and configuration information for all Smart Call Home-enabled devices.

Subscribe to Alert Groups

An alert group is a predefined subset of the Smart Call Home alerts that are supported on the ASA. Different types of Smart Call Home alerts are grouped into different alert groups, depending on their type. Each alert group reports the output of certain CLIs. The supported Smart Call Home alert groups are the following:

- syslog
- diagnostic
- environment
- inventory
- configuration
- threat
- snapshot
- telemetry
- test

Attributes of Alert Groups

Alert groups have the following attributes:

- Events first register with one alert group.
- A group can associate with multiple events.
- You can subscribe to specific alert groups.
- You can enable and disable specific alert groups. The default setting is enabled for all alert groups.
- The diagnostic and environment alert groups support subscription for periodic messages.
- The syslog alert group supports message ID-based subscription.

- You can configure a threshold for CPU and memory usage for the environment alert group. When a certain parameter has exceeded a predefined threshold, a message is sent. Most of the threshold values are platform-dependent and cannot be changed.
- You configure the snapshot alert group to send the output of CLIs that you specify.

Messages Sent to Cisco by Alert Groups

Messages are sent to Cisco periodically and whenever the ASA reloads. These messages are categorized by alert groups.

Inventory alerts consist of output from the following commands:

- **show version**—Displays the ASA software version, hardware configuration, license key, and related uptime data for the device.
- **show inventory—**Retrieves and displays inventory information about each Cisco product that is installed in the networking device. Each product is identified by unique device information, called the UDI, which is a combination of three separate data elements: the product identifier (PID), the version identifier (VID), and the serial number (SN).
- **show failover state**—Displays the failover state of both units in a failover pair. The information displayed includes the primary or secondary status of the unit, the Active/Standby status of the unit, and the last reported reason for failover.
- **show environment**—Shows system environment information for ASA system components, such as hardware operational status for the chassis, drivers, fans, and power supplies, as well as temperature status, voltage, and CPU usage.

Configuration alerts consist of output from the following commands:

- **show context**—Shows allocated interfaces and the configuration file URL, the number of contexts configured, or if you enable Anonymous Reporting in the system execution space, from a list of all contexts.
- **show call-home registered-module status**—Shows the registered module status. If you use system configuration mode, the command displays system module status based on the entire device, not per context.
- **show running-config**—Shows the configuration that is currently running on the ASA.
- **show startup-config**—Show the startup configuration.
- **show access-list | include elements**—Shows the hit counters and a time stamp value for an access list.

Diagnostic alerts consist of output from the following commands:

- **show failover**—Displays information about the failover status of the unit.
- **show interface—Displays interface statistics.**
- **show cluster info—Displays cluster information.**
- **show cluster history—Displays the cluster history.**

- **show crashinfo** (truncated)—After an unexpected software reload, the device sends a modified crash information file with only the traceback section of the file included, so only function calls, register values, and stack dumps are reported to Cisco.
- **show tech-support no-config—Displays the information that is used for diagnosis by technical support analysts.**

Environment alerts consist of output from the following commands:

- **show environment**—Shows system environment information for ASA system components, such as hardware operational status for the chassis, drivers, fans, and power supplies, as well as temperature status, voltage, and CPU usage.
- **show cpu usage**—Displays CPU usage information.
- **show memory detail**—Displays details of the free and allocated system memory.

Threat alerts consist of output from the following commands:

- **show threat-detection rate**—Displays threat detection statistics.
- **show threat-detection shun**—Displays currently shunned hosts.
- **show shun**—Displays shun information.
- **show dynamic-filter reports top**—Generates reports of the top 10 malware sites, ports, and infected hosts classified by the Botnet Traffic Filter.

Snapshot alerts may consist of output from the following commands:

- **show conn count**—Shows the number of active connections.
- **show asp drop**—Shows the accelerated security path dropped packets or connections.

Telemetry alerts consist of output from the following commands:

- **show perfmon detail—Shows** ASA performance details.
- **show traffic—Displays interface transmit and receive activity.**
- **show conn count**—Shows the number of active connections.
- **show vpn-sessiondb summary—Shows VPN session summary information.**
- **show vpn load-balancing—Displays the runtime statistics for the VPN load-balancing virtual cluster configuration.**
- **show local-host | include interface—Shows the network states of local hosts.**
- **show memory—Displays a summary of the maximum physical memory and current free memory available to the operating system.**
- **show context**—Shows allocated interfaces and the configuration file URL, the number of contexts configured, or if you enable Anonymous Reporting in the system execution space, from a list of all contexts.
- **show access-list | include elements**—Shows the hit counters and a time stamp value for an access list.
- **show interface—Displays interface statistics.**

- **show threat-detection statistics protocol—Shows IP protocol statistics.**
- **show phone-proxy media-sessions count—Displays the number of corresponding media sessions stored by the Phone Proxy.**
- **show phone-proxy secure-phones count—Displays the number of phones capable of secure mode stored in the database.**
- **show route—Displays the routing table.**
- **show xlate count—Shows the number of NAT sessions (xlates).**

Message Severity Threshold

When you subscribe a destination profile to certain alert groups, you can set a threshold for sending alert group messages based on the message severity level. Any message with a value lower than the destination profile's specified threshold is not sent to the destination.

The following table shows the mapping between message severity levels and syslog severity levels.

Table 76: Message Severity Level and Syslog Level Mapping

Level	Message Severity Level	Syslog Severity Level	Description
9	Catastrophic	N/A	Network-wide catastrophic failure.
8	Disaster	N/A	Significant network impact.
7	Determined by the specified CLI keyword: **subscribe-to-alert-group** *name of alert group* **severity** *severity level*	0	Emergency. System is unusable.
6	Determined by the specified CLI keyword: **subscribe-to-alert-group** *name of alert group* **severity** *severity level*	1	Alert. Critical conditions; immediate attention needed.
5	Determined by the specified CLI keyword: **subscribe-to-alert-group** *name of alert group* **severity** *severity level*	2	Critical. Major conditions.
4	Determined by the specified CLI keyword: **subscribe-to-alert-group** *name of alert group* **severity** *severity level*	3	Error. Minor conditions.

Level	Message Severity Level	Syslog Severity Level	Description
3	Warning	4	Warning conditions.
2	Notification	5	Basic notification and informational messages. Possibly independently insignificant.
1	Normal	6	Information. Normal event, signifying a return to normal state.
0	Debugging	7	Debugging messages (default setting).

Subscription Profiles

A subscription profile allows you to associate the destination recipients with interested groups. When an event registered with a subscribed group in a profile is triggered, the message associated with the event is sent to the configured recipients. Subscription profiles have the following attributes:

- You can create and configure multiple profiles.
- A profile may configure multiple e-mail or HTTPS recipients.
- A profile may subscribe multiple groups to a specified severity level.
- A profile supports three message formats: short text, long text, and XML.
- You can enable and disable a specific profile. Profiles are disabled by default.
- You can specify the maximum message size. The default is 3 MB.

A default profile, “Cisco TAC,” has been provided. The default profile has a predefined set of groups (diagnostic, environment, inventory, configuration, and telemetry) to monitor and predefined destination e-mail and HTTPS URLs. The default profile is created automatically when you initially configure Smart Call Home. The destination e-mail is callhome@cisco.com and the destination URL is https://tools.cisco.com/its/service/oddce/services/DDCEService.

Note You cannot change the destination e-mail or the destination URL of the default profile.

When you subscribe a destination profile to the configuration, inventory, telemetry, or snapshot alert groups, you can choose to receive the alert group messages asynchronously or periodically at a specified time.

The following table maps the default alert group to its severity level subscription and period (if applicable):

Table 77: Alert Group to Severity Level Subscription Mapping

Alert Group	Severity Level	Period
Configuration	Informational	Monthly
Diagnostic	Informational and higher	N/A
Environment	Notification and higher	N/A

Alert Group	Severity Level	Period
Inventory	Informational	Monthly
Snapshot	Informational	N/A
Syslog	Equivalent syslog	N/A
Telemetry	Informational	Daily
Test	N/A	N/A
Threat	Notification	N/A

Guidelines for Anonymous Reporting and Smart Call Home

This section includes the guidelines and limitation that you should review before configuring Anonymous reporting and Smart Call Home.

Anonymous Reporting Guidelines

- DNS must be configured.
- If an Anonymous Reporting message cannot be sent on the first try, the ASA retries two more times before dropping the message.
- Anonymous Reporting may coexist with other Smart Call Home configurations without changing the existing configuration. For example, if Smart Call Home is disabled before enabling Anonymous Reporting, it remains disabled, even after Anonymous Reporting has been enabled.
- If Anonymous Reporting is enabled, you cannot remove the trust point, and when Anonymous Reporting is disabled, the trust point remains. If Anonymous Reporting is disabled, you can remove the trust point, but disabling Anonymous Reporting does not cause the trust point to be removed.
- If you are using a multiple context mode configuration, the **dns**, **interface**, and **trustpoint** commands are in the admin context, and the **call-home** commands are in the system context.
- You can automate the update of the trustpool bundle at periodic intervals so that Smart Call Home can remain active if the self-signed certificate of the CA server changes. This trustpool auto renewal feature is not supported under multi-context deployments.

Smart Call Home Guidelines

- In multiple context mode, the subscribe-to-alert-group snapshot periodic command is divided into two commands: one to obtain information from the system configuration and one to obtain information from the user context.
- The Smart Call Home back-end server can accept messages in XML format only.
- A Smart Call Home message is sent to Cisco to report important cluster events if you have enabled clustering and configured Smart Call Home to subscribe to the diagnostic alert group with a critical severity level. A Smart Call Home clustering message is sent for only the following events:
    - When a unit joins the cluster

- When a unit leaves the cluster
- When a cluster unit becomes the cluster control unit
- When a secondary unit fails in the cluster

Each message that is sent includes the following information:

- The active cluster member count
- The output of the **show cluster info** command and the **show cluster history** command on the cluster control unit

Configure Anonymous Reporting and Smart Call Home

While Anonymous Reporting is part of the Smart Call Home service and allows Cisco to anonymously receive minimal error and health information from your device, the Smart Call Home service provides customized support of your system health, enabling Cisco TAC to monitor your devices and open a case when there is an issue, often before you know the issue has occurred.

You can have both services configured on your system at the same time, although configuring the Smart Call Home service provides the same functionality as Anonymous Reporting, plus customized services.

When you enter configuration mode, you receive a prompt that requests you to enable the Anonymous Reporting and Smart Call Home services according to the following guidelines:

- At the prompt, you may choose [Y]es, [N]o, or [A]sk later. If you choose [A]sk later, then you are reminded again in seven days or when the ASA reloads. If you continue to choose [A]sk later, the ASA prompts two more times at seven-day intervals before it assumes a [N]o response and does not ask again.
- If you did not receive the prompt, you may enable Anonymous Reporting or Smart Call Home by performing the steps in Configure Anonymous Reporting, on page 1391 or in Configure Smart Call Home, on page 1392.

Configure Anonymous Reporting

To configure Anonymous Reporting, perform the following steps:

Procedure

Step 1 Enable the Anonymous Reporting feature and create a new anonymous profile.

call-home reporting anonymous

Example:

```
ciscoasa(config)# call-home reporting anonymous
```

Entering this command creates a trust point and installs a certificate that is used to verify the identity of the Cisco web server.

Step 2 (Optional) Make sure that you have connectivity to the server and that your system can send messages.

call-home test reporting anonymous

Example:

```
ciscoasa(config)# call-home test reporting anonymous

INFO: Sending test message to
https://tools.cisco.com/its/service/oddce/services/DDCEService...

INFO: Succeeded
```

A success or error message returns test results.

Configure Smart Call Home

Configuring the Smart Call Home service on your ASA includes the following tasks:

Procedure

Step 1 Enable the Smart Call Home service. See Enable Smart Call Home, on page 1392.

Step 2 Configure the mail server through which Smart Call Home messages are delivered to subscribers. See Configure the Mail Server, on page 1397.

Step 3 Set up contact information for the Smart Call Home messages. See Configure Customer Contact Information, on page 1396.

Step 4 Define alert processing parameters, such as the maximum rate of events that can be handled. See Configure Alert Group Subscription, on page 1395.

Step 5 Set up alert subscription profiles. See Configure a Destination Profile, on page 1399.

Each alert subscription profile identifies the following:

- The subscribers to whom the Smart Call Home messages are sent, such as a Smart Call Home server at Cisco or a list of e-mail recipients.
- Information categories for which you want to receive alerts, such as configuration or inventory information.

Enable Smart Call Home

To enable Smart Call Home and activate your call-home profile, perform the following steps:

Procedure

Step 1 Enable the Smart Call Home service.

service call-home

Example:

```
ciscoasa(config)# service call-home
```

Step 2 Enter call-home configuration mode.

call-home

Example:

```
ciscoasa(config)# call home
```

Declare and Authenticate a Certificate Authority Trust Point

If Smart Call Home is configured to send messages to a web server through HTTPS, you need to configure the ASA to trust the certificate of the web server or the certificate of the Certificate Authority (CA) that issued the certificate. The Cisco Smart Call Home Production server certificate is issued by Verisign. The Cisco Smart Call Home Staging server certificate is issued by the Digital Signature Trust Company.



Note You should set the trust point for no client-types/no validation-usage to prevent it from being used for VPN validation.

To declare and authenticate the Cisco server security certificate and establish communication with the Cisco HTTPS server for Smart Call Home service, perform the following steps:

Procedure

Step 1 (Multiple Context Mode only) Install the certificate in the admin context.

changeto context *admincontext*

Example:

```
ciscoasa(config)# changeto context contextA
```

Step 2 Configure a trust point and prepare for certificate enrollment.

crypto ca trustpoint *trustpoint-name*

Example:

```
ciscoasa(config)# crypto ca trustpoint cisco
```

Note

If you use HTTP as the transport method, you must install a security certificate through a trust point, which is required for HTTPS. Find the specific certificate to install at the following URL:

http://www.cisco.com/en/US/docs/switches/lan/smart_call_home/SCH31_Ch6.html#wp1035380

Step 3 Specify a manual cut-and-paste method of certificate enrollment.

enroll terminal

Example:

```
ciscoasa(ca-trustpoint)# enroll terminal
```

Step 4 Authenticate the named CA. The CA name should match the trust point name specified in the **crypto ca trustpoint** command. At the prompt, paste the security certificate text.

crypto ca authenticate trustpoint

Example:

```
ciscoasa(ca-trustpoint)# crypto ca authenticate cisco
```

Step 5 Specify the end of the security certificate text and confirm acceptance of the entered security certificate.

quit

Example:

```
ciscoasa(ca-trustpoint)# quit
%Do you accept this certificate [yes/no]:
yes
```

Configure the Environment and Snapshot Alert Groups

To configure the environment and snapshot alert groups, perform the following steps:

Procedure

Enter alert-group-configuration mode.

alert-group-config {**environment** | **snapshot**}

Example:

```
ciscoasa(config)# alert-group-config environment
```

Configure Alert Group Subscription

To subscribe a destination profile to an alert group, perform the following steps:

Procedure

Step 1 Enter call-home configuration mode.

call-home

Example:

```
ciscoasa(config)# call-home
```

Step 2 Enable the specified Smart Call Home alert group.

alert-group {**all** | **configuration** | **diagnostic** | **environment** | **inventory** | **syslog**}

Example:

```
ciscoasa(cfg-call-home)# alert-group syslog
```

Use the **all** keyword to enable all alert groups. By default, all alert groups are enabled.

Step 3 Enter the profile configuration mode for the specified destination profile.

profile *profile-name*

Example:

```
ciscoasa(cfg-call-home)# profile CiscoTAC-1
```

Step 4 Subscribe to all available alert groups.

subscribe-to-alert-group all

Example:

```
ciscoasa(cfg-call-home-profile)# subscribe-to-alert-group all
```

Step 5 Subscribe this destination profile to the configuration alert group.

subscribe-to-alert-group configuration periodic {**daily** *hh:mm* | **monthly** *date hh:mm* | **weekly** *day hh:mm*}

Example:

```
ciscoasa(cfg-call-home-profile)# subscribe-to-alert-group configuration periodic weekly
Wednesday 23:30
```

The **periodic** keyword configures the configuration alert group for periodic notification. The default period is daily.

The **daily** keyword specifies the time of the day to send, in the *hh:mm* format, with a 24-hour clock (for example, 14:30).

The **weekly** keyword specifies the day of the week and time of day in the *day hh:mm* format, where the day of the week is spelled out (for example, Monday).

The **monthly** keyword specifies the numeric date, from 1 to 31, and the time of day, in the *date hh:mm* format.

Configure Customer Contact Information

To configure customer contact information, perform the following steps:

Procedure

Step 1 Enter call-home configuration mode.

call-home

Example:

```
ciscoasa(config)# call-home
```

Step 2 Specify the customer phone number. Spaces are allowed, but you must use quotes around the string if it includes spaces.

phone-number *phone-number-string*

Example:

```
ciscoasa(cfg-call-home)# phone-number 8005551122
```

Step 3 Specify the customer address, which is a free-format string that may be up to 255 characters long. Spaces are allowed, but you must use quotes around the string if it includes spaces.

street-address *street-address*

Example:

```
ciscoasa(cfg-call-home)# street-address “1234 Any Street, Any city, Any state, 12345”
```

Step 4 Specify the customer name, which may be up to 128 characters long. Spaces are allowed, but you must use quotes around the string if it includes spaces.

contact-name *contact-name*

Example:

```
ciscoasa(cfg-call-home)# contact-name contactname1234
```

Step 5 Specify the Cisco customer ID, which may be up to 64 characters long. Spaces are allowed, but you must use quotes around the string if it includes spaces.

customer-id *customer-id-string*

Example:

```
ciscoasa(cfg-call-home)# customer-id customer1234
```

Step 6 Specify the customer site ID, which may be up to 64 characters long. Spaces are allowed, but you must use quotes around the string if it includes spaces.

site-id *site-id-string*

Example:

```
ciscoasa(cfg-call-home)# site-id site1234
```

Step 7 Specify the customer contract identification, which may be up to 128 characters long. Spaces are allowed, but you must use quotes around the string if it includes spaces.

contract-id *contract-id-string*

Example:

```
ciscoasa(cfg-call-home)# contract-id contract1234
```

Example

The following example shows how to configure contact information:

```
ciscoasa(config)# call-home
ciscoasa(cfg-call-home)# contact-email-addr username@example.com
ciscoasa(cfg-call-home)# phone-number 8005551122
ciscoasa(cfg-call-home)# street-address "1234 Any Street, Any city, Any state, 12345"
ciscoasa(cfg-call-home)# contact-name contactname1234
ciscoasa(cfg-call-home)# customer-id customer1234
ciscoasa(cfg-call-home)# site-id site1234
ciscoasa(cfg-call-home)# contract-id contract1234
```

Configure the Mail Server

We recommend that you use HTTPS for message transport because it is the most secure. However, you may configure an e-mail destination for Smart Call Home and then configure the mail server to use the e-mail message transport.

To configure the mail server, perform the following steps:

Procedure

Step 1 Enter call-home configuration mode.

call-home

Example:

```
ciscoasa(config)# call-home
```

Step 2 Specify the SMTP mail server.

mail-server*ip-address* **name priority** [**1-100**] [**all**]

Example:

```
ciscoasa(cfg-call-home)# mail-server 10.10.1.1 smtp.example.com priority 1
```

You can specify up to five mail servers, using five separate commands. You must configure at least one mail server for using e-mail transport of Smart Call Home messages.

The lower the number, the higher the priority of the mail server.

The *ip-address* argument can be an IPv4 or IPv6 mail server address.

Example

The following example shows how to configure a primary mail server (named ”smtp.example.com”) and a secondary mail server at IP address 10.10.1.1:

```
ciscoasa(config)# call-home
ciscoasa(cfg-call-home)# mail-server smtp.example.com priority 1
ciscoasa(cfg-call-home)# mail-server 10.10.1.1 priority 2
ciscoasa(cfg-call-home)# exit
ciscoasa(config)#
```

Configure Traffic Rate Limiting

To configure traffic rate limiting, perform the following steps:

Procedure

Step 1 Enter call-home configuration mode.

call-home

Example:

```
ciscoasa(config)# call-home
```

Step 2 Specify the number of messages that Smart Call Home can send per minute. The default value is 10 messages per minute.

rate-limit *msg-count*

Example:

```
ciscoasa(cfg-call-home)# rate-limit 5
```

Send Smart Call Home Communications

To send specific Smart Call Home communications, perform the following steps:

Procedure

Choose one of the following options:

- Option 1—Send a test message manually using a profile configuration.

  call-home test [*test-message*] **profile** *profile-name*

  Example:

  ```
  ciscoasa# call-home test [testing123] profile CiscoTAC-1
  ```

- Option 2—Send an alert group message to one destination profile, if specified. If no profile is specified, send messages to all profiles that are subscribed to the inventory, configuration, snapshot, or telemetry alert groups.

  call-home send alert-group inventory { | **configuration** | **snapshot** | **telemetry**} [**profile** *profile-name*]

  Example:

  ```
  ciscoasa# call-home send alert-group inventory
  ```

- Option 3—Send command output to an e-mail address. The specified CLI command can be any command, including commands for all registered modules.

  call-home send*cli command* [**email** *email*]

  Example:

  ```
  ciscoasa# call-home send cli destination email username@example.com
  ```

  If you specify an e-mail address, the command output is sent to that address. If no e-mail address is specified, the output is sent to Cisco TAC. The e-mail is sent in log text format with the service number, if specified, in the subject line.

  The service number is required only if no e-mail address is specified, or if a Cisco TAC e-mail address is specified.

Configure a Destination Profile

To configure a destination profile for e-mail or for HTTP, perform the following steps:

Procedure

Step 1 Enter call-home configuration mode.

call-home

Example:

```
ciscoasa(config)# call-home
```

Step 2 Enter the profile configuration mode for the specified destination profile. If the specified destination profile does not exist, it is created.

profile *profile-name*

Example:

```
ciscoasa(cfg-call-home)# profile newprofile
```

You can create a maximum of 10 active profiles. The default profile is to report back to Cisco TAC. If you want to send call home information to a different location (for example, your own server), you can configure a separate profile.

Step 3 Configure the destination, message size, message format, and transport method for the Smart Call Home message receiver. The default message format is XML, and the default enabled transport method is e-mail.

destination address {**email** *address* | **http** *url*[**reference-identity** *ref-id-name*]} | **message-size-limit** *size* | **preferred-msg-format** {**long-text** | **short-text** | **xml**} **transport-method** {**email** | **http**}}

Example:

```
ciscoasa(cfg-call-home-profile)# destination address http
https://example.cisco.com/its/service/example/services/ExampleService reference-identity
ExampleService
ciscoasa(cfg-call-home-profile)# destination address email username@example.com
ciscoasa(cfg-call-home-profile)# destination preferred-msg-format long-text
```

The **reference-identity** option enables RFC 6125 reference identity checks on the received server certificate. These only apply to destinations configured with an http address. ID checks are made based on a previously configured reference identity object. See Configure Reference Identities, on page 811 for details on the reference identity object.

The e-mail-address is the e-mail address of the Smart Call Home message receiver, which can be up to 100 characters long. By default, the maximum URL size is 5 MB.

Use the short-text format to send and read a message on a mobile device, and use the long text format to send and read a message on a computer.

If the message receiver is the Smart Call Home back-end server, ensure that the **preferred-msg-format** value is XML because the back-end server can accept messages in XML format only.

Use this command to change the transport method back to e-mail.

Copy a Destination Profile

To create a new destination profile by copying an existing one, perform the following steps:

Procedure

Step 1 Enter call-home configuration mode.

call-home

Example:

```
ciscoasa(config)# call-home
```

Step 2 Specify the profile to copy.

profile *profile-name*

Example:

```
ciscoasa(cfg-call-home)# profile newprofile
```

Step 3 Copy the content of an existing profile to a new profile.

copy profile *src-profile-name dest-profile-name*

Example:

```
ciscoasa(cfg-call-home)# copy profile newprofile profile1
```

The existing profile (*src-profile-name*) and the new profile (*dest-profile-name*) may be up to 23 characters long.

Example

The following example shows how to copy an existing profile:

```
ciscoasa(config)# call-home
ciscoasa(cfg-call-home)# profile newprofile
ciscoasa(cfg-call-home-profile)# copy profile newprofile profile1
```

Rename a Destination Profile

To change the name of an existing profile, perform the following steps:

Procedure

Step 1 Enter call-home configuration mode.

call-home

Example:

```
ciscoasa(config)# call-home
```

Step 2 Specify the profile to rename.

profile *profilename*

Example:

```
ciscoasa(cfg-call-home)# profile newprofile
```

Step 3 Change the name of an existing profile.

rename profile *src-profile-name dest-profile-name*

Example:

```
ciscoasa(cfg-call-home)# rename profile newprofile profile1
```

The existing profile (*src-profile-name*) and the new profile (*dest-profile-name*) may be up to 23 characters long.

Example

The following example shows how to rename an existing profile:

```
ciscoasa(config)# call-home
ciscoasa(cfg-call-home)# profile newprofile
ciscoasa(cfg-call-home-profile)# rename profile newprofile profile1
```

Monitoring Anonymous Reporting and Smart Call Home

See the following commands for monitoring Anonymous Reporting and Smart Call Home services.

- **show call-home detail**

  This command shows the current Smart Call Home detail configuration.

- **show call-home mail-server status**

  This command shows the current mail server status.

- **show call-home profile** {profile name | **all**}

  This command shows the configuration of Smart Call Home profiles.

- **show call-home registered-module status** [**all**]

  This command shows the registered module status.

- **show call-home statistics**

  This command shows call-home detail status.

- **show call-home**

  This command shows the current Smart Call Home configuration.

- **show running-config call-home**

  This command shows the current Smart Call Home running configuration.

- **show smart-call-home alert-group**

  This command shows the current status of Smart Call Home alert groups.

- **show running-config all**

  This command shows details about the Anonymous Reporting user profile.

Examples for Smart Call Home

The following example shows how to configure the Smart Call Home service:

```
ciscoasa (config)# service call-home
ciscoasa (config)# call-home
ciscoasa (cfg-call-home)# contact-email-addr customer@example.com
ciscoasa (cfg-call-home)# profile CiscoTAC-1
ciscoasa (cfg-call-home-profile)# destination address http
https://example.cisco.com/its/service/example/services/ExampleService
ciscoasa (cfg-call-home-profile)# destination address email callhome@example.com
ciscoasa (cfg-call-home-profile)# destination transport-method http
ciscoasa (cfg-call-home-profile)# subscribe-to-alert-group inventory periodic daily 23:30
ciscoasa (cfg-call-home-profile)# subscribe-to-alert-group configuration periodic weekly
Wednesday 23:30
ciscoasa (cfg-call-home-profile)# subscribe-to-alert-group environment
ciscoasa (cfg-call-home-profile)# subscribe-to-alert-group diagnostic
ciscoasa (cfg-call-home-profile)# subscribe-to-alert-group telemetry periodic weekly Monday
 23:30
```

History for Anonymous Reporting and Smart Call Home

Table 78: History for Anonymous Reporting and Smart Call Home

Feature Name	Platform Releases	Description
Smart Call Home	8.2(2)	The Smart Call Home service offers proactive diagnostics and real-time alerts on the ASA, and provides higher network availability and increased operational efficiency. We introduced or modified the following commands: **active (call home)**, **call-home, call-home send alert-group, call-home test, contact-email-addr**, **customer-id (call home)**, **destination (call home)**, **profile**, **rename profile**, **service call-home, show call-home, show call-home detail**, **show smart-call-home alert-group**, **show call-home profile**, **show call-home statistics**, **show call-home mail-server status**, **show running-config call-home**, **show call-home registered-module status all**, **site-id**, **street-address**, **subscribe-to-alert-group all**, **alert-group-config**, **subscribe-to-alert-group configuration**, **subscribe-to-alert-group diagnostic**, **subscribe-to-alert-group environment**, **subscribe-to-alert-group inventory periodic**, **subscribe-to-alert-group snapshot periodic**, **subscribe-to-alert-group syslog**, **subscribe-to-alert-group telemetry periodic**.
Anonymous Reporting	9.0(1)	You can help to improve the ASA platform by enabling Anonymous Reporting, which allows Cisco to securely receive minimal error and health information from a device. We introduced the following commands: **call-home reporting anonymous**, **call-home test reporting anonymous**.
Smart Call Home	9.1(2)	The **show local-host** command was changed to the **show local-host | include interface** command for telemetry alert group reporting.

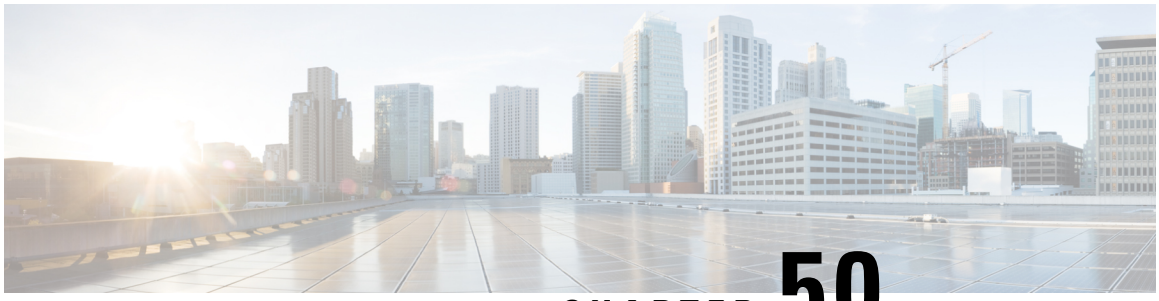
Feature Name	Platform Releases	Description
Smart Call Home	9.1(3)	A Smart Call Home message is sent to Cisco to report important cluster events if you have enabled clustering and configured Smart Call Home to subscribe to the Diagnostic alert group with a Critical severity level. A Smart Call Home clustering message is sent for only the following three events: • When a unit joins the cluster • When a unit leaves the cluster • When a cluster unit becomes the cluster control unit Each message that is sent includes the following information: • The active cluster member count • The output of the **show cluster info** command and the **show cluster history** command on the cluster control unit
Reference Identities for Secure Smart Call Home Server connections	9.6(2)	TLS client processing now supports rules for verification of a server identity defined in RFC 6125, Section 6. Identity verification will be done during PKI validation for TLS connections to the Smart Call Home Server. If the presented identity cannot be matched against the configured reference identity, the connection is not established. We added or modified the following commands: **[no] crypto ca reference-identity**, **call home profile destination address http**.



PART IX

Reference

- Using the Command-Line Interface, on page 1409
- Addresses, Protocols, and Ports, on page 1419



CHAPTER 50

Using the Command-Line Interface

This chapter describes how to use the CLI on the ASA.

Note The CLI uses similar syntax and other conventions to the Cisco IOS CLI, but the ASA operating system is not a version of Cisco IOS software. Do not assume that a Cisco IOS CLI command works with or has the same function on the ASA.

- Firewall Mode and Security Context Mode, on page 1409
- Command Modes and Prompts, on page 1410
- Syntax Formatting, on page 1411
- Abbreviate Commands, on page 1412
- Command-Line Editing, on page 1412
- Command Completion, on page 1412
- Command Help, on page 1412
- View the Running Configuration, on page 1413
- Filter show and more Command Output, on page 1413
- Redirecting and Appending show Command Output, on page 1414
- Getting a Line Count for show Command Output, on page 1414
- Command Output Paging, on page 1415
- Add Comments, on page 1415
- Text Configuration Files, on page 1416
- Supported Character Sets, on page 1417

Firewall Mode and Security Context Mode

The ASA runs in a combination of the following modes:

- Transparent firewall or routed firewall mode

  The firewall mode determines if the ASA runs as a Layer 2 or Layer 3 firewall.

- Multiple context or single context mode

  The security context mode determines if the ASA runs as a single device or as multiple security contexts, which act like virtual devices.

Some commands are only available in certain modes.

Command Modes and Prompts

The ASA CLI includes command modes. Some commands can only be entered in certain modes. For example, to enter commands that show sensitive information, you need to enter a password and enter a more privileged mode. Then, to ensure that configuration changes are not entered accidentally, you have to enter a configuration mode. All lower commands can be entered in higher modes, for example, you can enter a privileged EXEC command in global configuration mode.



Note The various types of prompts are all default prompts and when configured, they can be different.

- When you are in the system configuration or in single context mode, the prompt begins with the hostname:

  ```
  ciscoasa
  ```

- When printing the prompt string, the prompt configuration is parsed and the configured keyword values are printed in the order in which you have set the prompt command. The keyword arguments can be any of the following and in any order: hostname, domain, context, priority, state.

  prompt hostname context priority state

- When you are within a context, the prompt begins with the hostname followed by the context name:

  ```
  ciscoasa/context
  ```

The prompt changes depending on the access mode:

- User EXEC mode

  User EXEC mode lets you see minimum ASA settings. The user EXEC mode prompt appears as follows when you first access the ASA:

  ```
  ciscoasa>
  ```

  ```
  ciscoasa/context>
  ```

- Privileged EXEC mode

  Privileged EXEC mode lets you see all current settings up to your privilege level. Any user EXEC mode command will work in privileged EXEC mode. Enter the **enable** command in user EXEC mode, which requires a password, to start privileged EXEC mode. The prompt includes the number sign (#):

  ```
  ciscoasa#
  ```

  ```
  ciscoasa/context#
  ```

- Global configuration mode

Global configuration mode lets you change the ASA configuration. All user EXEC, privileged EXEC, and global configuration commands are available in this mode. Enter the **configure terminal** command in privileged EXEC mode to start global configuration mode. The prompt changes to the following:

```
ciscoasa(config)#

ciscoasa/context(config)#
```

- Command-specific configuration modes

  From global configuration mode, some commands enter a command-specific configuration mode. All user EXEC, privileged EXEC, global configuration, and command-specific configuration commands are available in this mode. For example, the **interface** command enters interface configuration mode. The prompt changes to the following:

```
ciscoasa(config-if)#

ciscoasa/context(config-if)#
```

Syntax Formatting

Command syntax descriptions use the conventions listed in the following table.

Table 79: Syntax Conventions

Convention	Description
bold	Bold text indicates commands and keywords that you enter literally as shown.
italics	Italic text indicates arguments for which you supply values.
[x]	Square brackets enclose an optional element (keyword or argument).
|	A vertical bar indicates a choice within an optional or required set of keywords or arguments.
[x | y]	Square brackets enclosing keywords or arguments separated by a vertical bar indicate an optional choice.
{x | y}	Braces enclosing keywords or arguments separated by a vertical bar indicate a required choice.
[x {y | z}]	Nested sets of square brackets or braces indicate optional or required choices within optional or required elements. Braces and a vertical bar within square brackets indicate a required choice within an optional element.

Abbreviate Commands

You can abbreviate most commands down to the fewest unique characters for a command; for example, you can enter **wr t** to view the configuration instead of entering the full command **write terminal**, or you can enter **en** to start privileged mode and **conf t** to start configuration mode. In addition, you can enter **0** to represent **0.0.0.0**.

Command-Line Editing

The ASA uses the same command-line editing conventions as Cisco IOS software. You can view all previously entered commands with the **show history** command or individually with the up arrow or **^p** command. Once you have examined a previously entered command, you can move forward in the list with the down arrow or **^n** command. When you reach a command you wish to reuse, you can edit it or press the **Enter** key to start it. You can also delete the word to the left of the cursor with **^w**, or erase the line with **^u**.

The ASA permits up to 512 characters in a command; additional characters are ignored.

Command Completion

To complete a command or keyword after entering a partial string, press the **Tab** key. The ASA only completes the command or keyword if the partial string matches only one command or keyword. For example, if you enter **s** and press the **Tab** key, the ASA does not complete the command because it matches more than one command. However, if you enter **dis**, the **Tab** key completes the **disable** command.

Command Help

Help information is available from the command line by entering the following commands:

- **help** *command_name*

  Shows help for the specific command.

- *command_name* **?**

  Shows a list of arguments available.

- *string***?** (no space)

  Lists the possible commands that start with the string.

- **?** and **+?**

  Lists all commands available. If you enter **?**, the ASA shows only commands available for the current mode. To show all commands available, including those for lower modes, enter **+?**.



Note If you want to include a question mark (?) in a command string, you must press **Ctrl-V** before typing the question mark so that you do not inadvertently invoke CLI help.

View the Running Configuration

To view the running configuration, use one of the following commands:

- **show running-config** [**all**] [*command*]

  If you specify **all**, then all default settings are shown as well. If you specify a *command*, then the output only includes related commands.

Note Many passwords are shown as *****. To view the passwords in plain text, or in encrypted form if you have a master passphrase enabled, use the **more** command.

- **more system:running-config**

Filter show and more Command Output

You can use the vertical bar (|) with any **show** command and include a filter option and filtering expression. The filtering is performed by matching each output line with a regular expression, similar to Cisco IOS software. By selecting different filter options you can include or exclude all output that matches the expression. You can also display all output beginning with the line that matches the expression.

The syntax for using filtering options with the **show** command is as follows:

show *command* | {**include**| **exclude** | **begin** | **grep** [**-v**]} *regexp*

or

more system:running-config| {**include**| **exclude** | **begin** | **grep** [**-v**]} *regexp*

Note Entering the **more** command allows you to view the contents of any file, not just the running configuration; see the command reference for more information.

In this command string, the first vertical bar (|) is the operator and must be included in the command. This operator directs the output of the **show** command to the filter. In the syntax diagram, the other vertical bars (|) indicate alternative options and are not part of the command.

The **include** option includes all output lines that match the regular expression. The **grep** option without **-v** has the same effect. The **exclude** option excludes all output lines that match the regular expression. The **grep** option with**-v** has the same effect. The **begin** option shows all the output lines starting with the line that matches the regular expression.

Replace *regexp* with any Cisco IOS regular expression. The regular expression is not enclosed in quotes or double-quotes, so be careful with trailing white spaces, which will be taken as part of the regular expression.

When creating regular expressions, you can use any letter or number that you want to match. In addition, certain keyboard characters called *metacharacters* have special meaning when used in regular expressions.

Use **Ctrl+V** to escape all of the special characters in the CLI, such as a question mark (?) or a tab. For example, type **d[Ctrl+V]?g** to enter **d?g** in the configuration.

Redirecting and Appending show Command Output

If there is a lot of output to show, command completion can take a long time. For example, you might think the system is stuck if you try to show a million access control entries, or a very large ASP table.

Instead of displaying the output of a **show** command on the screen, you can redirect it to a file on the device or in a remote location. When redirecting to a file on the device, you can also append the command output to the file.

show *command* | {**append | redirect**} *url*

- **append** *url* adds the output to an existing file. Specify the file using one of the following:
  - **disk0:/**[[*path/*]*filename*] or **flash:/**[[*path/*]*filename*]—Both **flash** and **disk0** indicates the internal Flash memory. You can use either option.
  - **disk1:/**[[*path/*]*filename*]—Indicates external memory.
- **redirect** *url* creates the specified file, or overwrites it if the file already exists.
  - **disk0:/**[[*path/*]*filename*] or **flash:/**[[*path/*]*filename*]—Both **flash** and **disk0** indicates the internal Flash memory. You can use either option.
  - **disk1:/**[[*path/*]*filename*]—Indicates external memory.
  - **smb:/**[[*path/*]*filename*]—Indicates Server Message Block, a UNIX server local file system.
  - **ftp://**[[*user*[**:***password*]@] *server*[:*port*]/[*path/*] *filename*[**;type=***xx*]]—Indicates an SCP server. The **type** can be one of these keywords: **ap** (ASCII passive mode), **an** (ASCII normal mode), **ip** (Default—Binary passive mode), **in** (Binary normal mode).
  - **scp://**[[*user*[:*password*]@] *server*[/*path*]/*filename*[**;int=***interface_name*]]—The **;int=***interface* option bypasses the route lookup and always uses the specified interface to reach the Secure Copy (SCP) server.
  - **tftp://**[[*user*[**:***password*]@] *server*[:*port*] /[*path/*]*filename*[**;int=***interface_name*]]—Indicates a TFTP server. The pathname cannot contain spaces. The **;int=***interface* option bypasses the route lookup and always uses the specified interface to reach the TFTP server.

Getting a Line Count for show Command Output

Instead of seeing actual **show** command output, you might simply want a count of the number of lines in the output, or the number of lines that match a regular expression. You can then easily compare the line count with the count from previous times you entered the command. This can be a quick check as you make configuration changes. You can either use the **count** keyword, or add **-c** to the grep keyword.

show *command* | **count** [*regular_expression*]

show *command* | **grep -c** [*regular_expression*]

Replace *regular_expression* with any Cisco IOS regular expression. The regular expression is not enclosed in quotes or double-quotes, so be careful with trailing white spaces, which will be taken as part of the regular expression. The regular expression is optional; if you do not include one, the count returns the total number of lines in the unfiltered output.

When creating regular expressions, you can use any letter or number that you want to match. In addition, certain keyboard characters called metacharacters have special meaning when used in regular expressions. Use Ctrl+V to escape all of the special characters in the CLI, such as a question mark (?) or a tab. For example, type **d[Ctrl+V]?g** to enter **d?g** in the configuration.

For example, to show the total number of all lines in the **show running-config** output:

```
ciscoasa# show running-config | count
Number of lines which match regexp = 271
```

The following example shows how you can quickly check how many interfaces are up. The first example shows how to use the **grep** keyword with a regular expression to filter on only those lines that show an up status. The next example adds the **-c** option to simply show the count rather than the actually lines of output.

```
ciscoasa# show interface | grep is up
Interface GigabitEthernet0/0 "outside", is up, line protocol is up
Interface GigabitEthernet0/1 "inside", is up, line protocol is up

ciscoasa# show interface | grep -c is up
Number of lines which match regexp = 2
```

Command Output Paging

For commands such as **help** or **?**, **show**, **show xlate**, or other commands that provide long listings, you can determine if the information displays a screen and pauses, or lets the command run to completion. The **pager** command lets you choose the number of lines to display before the More prompt appears.

When paging is enabled, the following prompt appears:

```
<--- More --->
```

The More prompt uses syntax similar to the UNIX **more** command:

- Press the **Space** bar to view another screen.
- Press the **Enter** key to view the next line.
- Press the q key to return to the command line.

Add Comments

You can precede a line with a colon (:) to create a comment. However, the comment only appears in the command history buffer and not in the configuration. Therefore, you can view the comment with the **show history** command or by pressing an arrow key to retrieve a previous command, but because the comment is not in the configuration, the **write terminal** command does not display it.

Text Configuration Files

This section describes how to format a text configuration file that you can download to the ASA.

How Commands Correspond with Lines in the Text File

The text configuration file includes lines that correspond with the commands described in this guide.

In examples, commands are preceded by a CLI prompt. The prompt in the following example is "ciscoasa(config)#":

```
ciscoasa(config)# context a
```

In the text configuration file you are not prompted to enter commands, so the prompt is omitted:

```
context a
```

Command-Specific Configuration Mode Commands

Command-specific configuration mode commands appear indented under the main command when entered at the command line. Your text file lines do not need to be indented, as long as the commands appear directly following the main command. For example, the following unindented text is read the same as indented text:

```
interface gigabitethernet0/0
nameif inside
interface gigabitethernet0/1
    nameif outside
```

Automatic Text Entries

When you download a configuration to the ASA, it inserts some lines automatically. For example, the ASA inserts lines for default settings or for the time the configuration was modified. You do not need to enter these automatic entries when you create your text file.

Line Order

For the most part, commands can be in any order in the file. However, some lines, such as ACEs, are processed in the order they appear, and the order can affect the function of the access list. Other commands might also have order requirements. For example, you must enter the **nameif** command for an interface first because many subsequent commands use the name of the interface. Also, commands in a command-specific configuration mode must directly follow the main command.

Commands Not Included in the Text Configuration

Some commands do not insert lines in the configuration. For example, a runtime command such as **show running-config** does not have a corresponding line in the text file.

Passwords

The login, enable, and user passwords are automatically encrypted before they are stored in the configuration. For example, the encrypted form of the password “cisco” might look like jMorNbK0514fadBh. You can copy the configuration passwords to another ASA in its encrypted form, but you cannot unencrypt the passwords yourself.

If you enter an unencrypted password in a text file, the ASA does not automatically encrypt it when you copy the configuration to the ASA. The ASA only encrypts it when you save the running configuration from the command line using the **copy running-config startup-config** or **write memory** command.

Multiple Security Context Files

For multiple security contexts, the entire configuration consists of the following multiple parts:

- The security context configurations
- The system configuration, which identifies basic settings for the ASA, including a list of contexts
- The admin context, which provides network interfaces for the system configuration

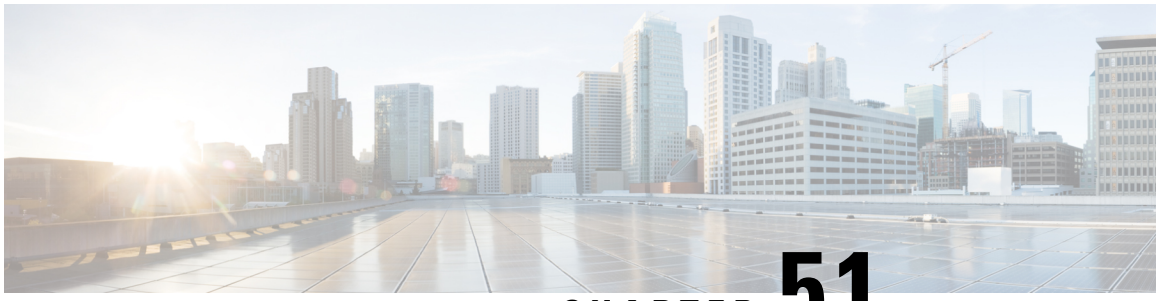
  The system configuration does not include any interfaces or network settings for itself. Rather, when the system needs to access network resources (such as downloading the contexts from the server), it uses a context that is designated as the admin context.

Each context is similar to a single context mode configuration. The system configuration differs from a context configuration in that the system configuration includes system-only commands (such as a list of all contexts) while other typical commands are not present (such as many interface parameters).

Supported Character Sets

The ASA CLI currently supports UTF-8 encoding only. UTF-8 is the particular encoding scheme for Unicode symbols, and has been designed to be compatible with an ASCII subset of symbols. ASCII characters are represented in UTF-8 as one-byte characters. All other characters are represented in UTF-8 as multibyte symbols.

The ASCII printable characters (0x20 to 0x7e) are fully supported. The printable ASCII characters are the same as ISO 8859-1. UTF-8 is a superset of ISO 8859-1, so the first 256 characters (0-255) are the same as ISO 8859-1. The ASA CLI supports up to 255 characters (multibyte characters) of ISO 8859-1.



CHAPTER 51

Addresses, Protocols, and Ports

This chapter provides a quick reference for IP addresses, protocols, and applications.

- IPv4 Addresses and Subnet Masks, on page 1419
- IPv6 Addresses, on page 1423
- Protocols and Applications, on page 1428
- TCP and UDP Ports, on page 1429
- Local Ports and Protocols, on page 1433
- ICMP Types, on page 1434

IPv4 Addresses and Subnet Masks

This section describes how to use IPv4 addresses in ASA. An IPv4 address is a 32-bit number written in dotted-decimal notation: four 8-bit fields (octets) converted from binary to decimal numbers, separated by dots. The first part of an IP address identifies the network on which the host resides, while the second part identifies the particular host on the given network. The network number field is called the network prefix. All hosts on a given network share the same network prefix but must have a unique host number. In classful IP, the class of the address determines the boundary between the network prefix and the host number.

Classes

IP host addresses are divided into three different address classes: Class A, Class B, and Class C. Each class fixes the boundary between the network prefix and the host number at a different point within the 32-bit address. Class D addresses are reserved for multicast IP.

- Class A addresses (1.xxx.xxx.xxx through 126.xxx.xxx.xxx) use only the first octet as the network prefix.
- Class B addresses (128.0.xxx.xxx through 191.255.xxx.xxx) use the first two octets as the network prefix.
- Class C addresses (192.0.0.xxx through 223.255.255.xxx) use the first three octets as the network prefix.

Because Class A addresses have 16,777,214 host addresses, and Class B addresses 65,534 hosts, you can use subnet masking to break these huge networks into smaller subnets.

Private Networks

If you need large numbers of addresses on your network, and they do not need to be routed on the Internet, you can use private IP addresses that the Internet Assigned Numbers Authority (IANA) recommends (see RFC 1918). The following address ranges are designated as private networks that should not be advertised:

- 10.0.0.0 through 10.255.255.255
- 172.16.0.0 through 172.31.255.255
- 192.168.0.0 through 192.168.255.255

Subnet Masks

A subnet mask lets you convert a single Class A, B, or C network into multiple networks. With a subnet mask, you can create an extended network prefix that adds bits from the host number to the network prefix. For example, a Class C network prefix always consists of the first three octets of the IP address. But a Class C extended network prefix uses part of the fourth octet as well.

Subnet masking is easy to understand if you use binary notation instead of dotted decimal. The bits in the subnet mask have a one-to-one correspondence with the Internet address:

- The bits are set to 1 if the corresponding bit in the IP address is part of the extended network prefix.
- The bits are set to 0 if the bit is part of the host number.

Example 1: If you have the Class B address 129.10.0.0 and you want to use the entire third octet as part of the extended network prefix instead of the host number, then you must specify a subnet mask of 11111111.11111111.11111111.00000000. This subnet mask converts the Class B address into the equivalent of a Class C address, where the host number consists of the last octet only.

Example 2: If you want to use only part of the third octet for the extended network prefix, then you must specify a subnet mask like 11111111.11111111.11111000.00000000, which uses only 5 bits of the third octet for the extended network prefix.

You can write a subnet mask as a dotted-decimal mask or as a /*bits* (“slash *bits*”) mask. In Example 1, for a dotted-decimal mask, you convert each binary octet into a decimal number: 255.255.255.0. For a /*bits* mask, you add the number of 1s: /24. In Example 2, the decimal number is 255.255.248.0 and the /bits is /21.

You can also supernet multiple Class C networks into a larger network by using part of the third octet for the extended network prefix. For example, 192.168.0.0/20.

Determine the Subnet Mask

See the following table to determine the subnet mask based on how many hosts you want.



Note The first and last number of a subnet are reserved, except for /32, which identifies a single host.

Table 80: Hosts, Bits, and Dotted-Decimal Masks

Hosts	/Bits Mask	Dotted-Decimal Mask
16,777,216	/8	255.0.0.0 Class A Network
65,536	/16	255.255.0.0 Class B Network
32,768	/17	255.255.128.0
16,384	/18	255.255.192.0
8192	/19	255.255.224.0
4096	/20	255.255.240.0
2048	/21	255.255.248.0
1024	/22	255.255.252.0
512	/23	255.255.254.0
256	/24	255.255.255.0 Class C Network
128	/25	255.255.255.128
64	/26	255.255.255.192
32	/27	255.255.255.224
16	/28	255.255.255.240
8	/29	255.255.255.248
4	/30	255.255.255.252
Do not use	/31	255.255.255.254
1	/32	255.255.255.255 Single Host Address

Determine the Address to Use with the Subnet Mask

The following sections describe how to determine the network address to use with a subnet mask for a Class C-size and a Class B-size network.

Class C-Size Network Address

For a network between 2 and 254 hosts, the fourth octet falls on a multiple of the number of host addresses, starting with 0. For example, The following table shows the 8-host subnets (/29) of 192.168.0.x.

Note The first and last address of a subnet are reserved. In the first subnet example, you cannot use 192.168.0.0 or 192.168.0.7.

Table 81: Class C-Size Network Address

Subnet with Mask /29 (255.255.255.248)	Address Range
192.168.0.0	192.168.0.0 to 192.168.0.7
192.168.0.8	192.168.0.8 to 192.168.0.15
192.168.0.16	192.168.0.16 to 192.168.0.31
—	—
192.168.0.248	192.168.0.248 to 192.168.0.255

Class B-Size Network Address

To determine the network address to use with the subnet mask for a network with between 254 and 65,534 hosts, you need to determine the value of the third octet for each possible extended network prefix. For example, you might want to subnet an address like 10.1.*x*.0, where the first two octets are fixed because they are used in the extended network prefix, and the fourth octet is 0 because all bits are used for the host number.

To determine the value of the third octet, follow these steps:

1. Calculate how many subnets you can make from the network by dividing 65,536 (the total number of addresses using the third and fourth octet) by the number of host addresses you want.

   For example, 65,536 divided by 4096 hosts equals 16. Therefore, there are 16 subnets of 4096 addresses each in a Class B-size network.

2. Determine the multiple of the third octet value by dividing 256 (the number of values for the third octet) by the number of subnets:

   In this example, 256/16 = 16.

   The third octet falls on a multiple of 16, starting with 0.

The following table shows the 16 subnets of the network 10.1.



Note The first and last address of a subnet are reserved. In the first subnet example, you cannot use 10.1.0.0 or 10.1.15.255.

Table 82: Subnets of Network

Subnet with Mask /20 (255.255.240.0)	Address Range
10.1.0.0	10.1.0.0 to 10.1.15.255
10.1.16.0	10.1.16.0 to 10.1.31.255
10.1.32.0	10.1.32.0 to 10.1.47.255
—	—
10.1.240.0	10.1.240.0 to 10.1.255.255

IPv6 Addresses

IPv6 is the next generation of the Internet Protocol after IPv4. It provides an expanded address space, a simplified header format, improved support for extensions and options, flow labeling capability, and authentication and privacy capabilities. IPv6 is described in RFC 2460. The IPv6 addressing architecture is described in RFC 3513.

This section describes the IPv6 address format and architecture.

IPv6 Address Format

IPv6 addresses are represented as a series of eight 16-bit hexadecimal fields separated by colons (:) in the format: x:x:x:x:x:x:x:x. The following are two examples of IPv6 addresses:

- 2001:0DB8:7654:3210:FEDC:BA98:7654:3210
- 2001:0DB8:0000:0000:0008:0800:200C:417A



Note The hexadecimal letters in IPv6 addresses are not case-sensitive.

You do not need to include the leading zeros in an individual field of the address, but each field must contain at least one digit. So the example address 2001:0DB8:0000:0000:0008:0800:200C:417A can be shortened to 2001:0DB8:0:0:8:800:200C:417A by removing the leading zeros from the third through sixth fields from the left. The fields that contained all zeros (the third and fourth fields from the left) were shortened to a single zero. The fifth field from the left had the three leading zeros removed, leaving a single 8 in that field, and the sixth field from the left had the one leading zero removed, leaving 800 in that field.

It is common for IPv6 addresses to contain several consecutive hexadecimal fields of zeros. You can use two colons (::) to compress consecutive fields of zeros at the beginning, middle, or end of an IPv6 address (the colons represent the successive hexadecimal fields of zeros). The following table shows several examples of address compression for different types of IPv6 address.

Table 83: IPv6 Address Compression Examples

Address Type	Standard Form	Compressed Form
Unicast	2001:0DB8:0:0:0:BA98:0:3210	2001:0DB8::BA98:0:3210
Multicast	FF01:0:0:0:0:0:0:101	FF01::101
Loopback	0:0:0:0:0:0:0:1	::1
Unspecified	0:0:0:0:0:0:0:0	::



Note Two colons (::) can be used only once in an IPv6 address to represent successive fields of zeros.

An alternative form of the IPv6 format is often used when dealing with an environment that contains both IPv4 and IPv6 addresses. This alternative has the format x:x:x:x:x:x:y.y.y.y, where x represent the hexadecimal values for the six high-order parts of the IPv6 address and y represent decimal values for the 32-bit IPv4 part of the address (which takes the place of the remaining two 16-bit parts of the IPv6 address). For example, the IPv4 address 192.168.1.1 could be represented as the IPv6 address 0:0:0:0:0:0:FFFF:192.168.1.1 or ::FFFF:192.168.1.1.

IPv6 Address Types

The following are the three main types of IPv6 addresses:

- **Unicast**—A unicast address is an identifier for a single interface. A packet sent to a unicast address is delivered to the interface identified by that address. An interface may have more than one unicast address assigned to it.
- **Multicast**—A multicast address is an identifier for a set of interfaces. A packet sent to a multicast address is delivered to all addresses identified by that address.
- **Anycast**—An anycast address is an identifier for a set of interfaces. Unlike a multicast address, a packet sent to an anycast address is only delivered to the "nearest" interface, as determined by the measure of distances for the routing protocol.



Note There are no broadcast addresses in IPv6. Multicast addresses provide the broadcast functionality.

Unicast Addresses

This section describes IPv6 unicast addresses. Unicast addresses identify an interface on a network node.

Global Address

The general format of an IPv6 global unicast address is a global routing prefix followed by a subnet ID followed by an interface ID. The global routing prefix can be any prefix not reserved by another IPv6 address type.

All global unicast addresses, other than those that start with binary 000, have a 64-bit interface ID in the Modified EUI-64 format.

Global unicast address that start with the binary 000 do not have any constraints on the size or structure of the interface ID portion of the address. One example of this type of address is an IPv6 address with an embedded IPv4 address.

Site-Local Address

Site-local addresses are used for addressing within a site. They can be used to address an entire site without using a globally unique prefix. Site-local addresses have the prefix FEC0::/10, followed by a 54-bit subnet ID, and end with a 64-bit interface ID in the modified EUI-64 format.

Site-local routers do not forward any packets that have a site-local address for a source or destination outside of the site. Therefore, site-local addresses can be considered private addresses.

Link-Local Address

All interfaces are required to have at least one link-local address. You can configure multiple IPv6 addresses per interfaces, but only one link-local address.

A link-local address is an IPv6 unicast address that can be automatically configured on any interface using the link-local prefix FE80::/10 and the interface identifier in modified EUI-64 format. Link-local addresses are used in the neighbor discovery protocol and the stateless autoconfiguration process. Nodes with a link-local address can communicate; they do not need a site-local or globally unique address to communicate.

Routers do not forward any packets that have a link-local address for a source or destination. Therefore, link-local addresses can be considered private addresses.

IPv4-Compatible IPv6 Addresses

There are two types of IPv6 addresses that can contain IPv4 addresses.

The first type is the IPv4-compatibly IPv6 address. The IPv6 transition mechanisms include a technique for hosts and routers to dynamically tunnel IPv6 packets over IPv4 routing infrastructure. IPv6 nodes that use this technique are assigned special IPv6 unicast addresses that carry a global IPv4 address in the low-order 32 bits. This type of address is termed an IPv4-compatible IPv6 address and has the format ::y.y.y.y, where y.y.y.y is an IPv4 unicast address.



Note The IPv4 address used in the IPv4-compatible IPv6 address must be a globally unique IPv4 unicast address.

The second type of IPv6 address, which holds an embedded IPv4 address, is called the IPv4-mapped IPv6 address. This address type is used to represent the addresses of IPv4 nodes as IPv6 addresses. This type of address has the format ::FFFF:y.y.y.y, where y.y.y.y is an IPv4 unicast address.

Unspecified Address

The unspecified address, 0:0:0:0:0:0:0:0, indicates the absence of an IPv6 address. For example, a newly initialized node on an IPv6 network may use the unspecified address as the source address in its packets until it receives its IPv6 address.



Note The IPv6 unspecified address cannot be assigned to an interface. The unspecified IPv6 addresses must not be used as destination addresses in IPv6 packets or the IPv6 routing header.

Loopback Address

The loopback address, 0:0:0:0:0:0:0:1, may be used by a node to send an IPv6 packet to itself. The loopback address in IPv6 functions the same as the loopback address in IPv4 (127.0.0.1).



Note The IPv6 loopback address cannot be assigned to a physical interface. A packet that has the IPv6 loopback address as its source or destination address must remain within the node that created the packet. IPv6 routers do not forward packets that have the IPv6 loopback address as their source or destination address.

Interface Identifiers

Interface identifiers in IPv6 unicast addresses are used to identify the interfaces on a link. They need to be unique within a subnet prefix. In many cases, the interface identifier is derived from the interface link-layer address. The same interface identifier may be used on multiple interfaces of a single node, as long as those interfaces are attached to different subnets.

For all unicast addresses, except those that start with the binary 000, the interface identifier is required to be 64 bits long and to be constructed in the Modified EUI-64 format. The Modified EUI-64 format is created from the 48-bit MAC address by inverting the universal/local bit in the address and by inserting the hexadecimal number FFFE between the upper three bytes and lower three bytes of the of the MAC address.

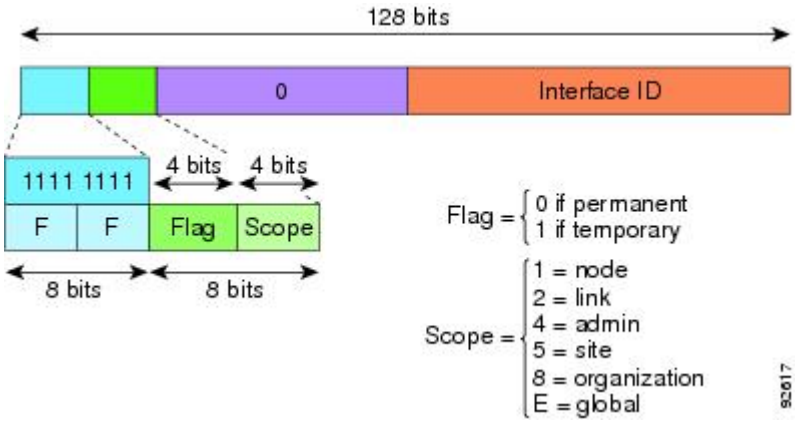
For example, and interface with the MAC address of 00E0.b601.3B7A would have a 64-bit interface ID of 02E0:B6FF:FE01:3B7A.

Multicast Address

An IPv6 multicast address is an identifier for a group of interfaces, typically on different nodes. A packet sent to a multicast address is delivered to all interfaces identified by the multicast address. An interface may belong to any number of multicast groups.

An IPv6 multicast address has a prefix of FF00::/8 (1111 1111). The octet following the prefix defines the type and scope of the multicast address. A permanently assigned (well known) multicast address has a flag parameter equal to 0; a temporary (transient) multicast address has a flag parameter equal to 1. A multicast address that has the scope of a node, link, site, or organization, or a global scope has a scope parameter of 1, 2, 5, 8, or E, respectively. For example, a multicast address with the prefix FF02::/16 is a permanent multicast address with a link scope. The following figure shows the format of the IPv6 multicast address.

Figure 74: IPv6 Multicast Address Format



IPv6 nodes (hosts and routers) are required to join the following multicast groups:

- The All Nodes multicast addresses:
  - FF01:: (interface-local)
  - FF02:: (link-local)
- The Solicited-Node Address for each IPv6 unicast and anycast address on the node: FF02:0:0:0:0:1:FFXX:XXXX/104, where XX:XXXX is the low-order 24-bits of the unicast or anycast address.

Note Solicited-Node addresses are used in Neighbor Solicitation messages.

IPv6 routers are required to join the following multicast groups:

- FF01::2 (interface-local)
- FF02::2 (link-local)
- FF05::2 (site-local)

Multicast address should not be used as source addresses in IPv6 packets.

Note There are no broadcast addresses in IPv6. IPv6 multicast addresses are used instead of broadcast addresses.

Anycast Address

The IPv6 anycast address is a unicast address that is assigned to more than one interface (typically belonging to different nodes). A packet that is routed to an anycast address is routed to the nearest interface having that address, the nearness being determined by the routing protocol in effect.

Anycast addresses are allocated from the unicast address space. An anycast address is simply a unicast address that has been assigned to more than one interface, and the interfaces must be configured to recognize the address as an anycast address.

The following restrictions apply to anycast addresses:

- An anycast address cannot be used as the source address for an IPv6 packet.
- An anycast address cannot be assigned to an IPv6 host; it can only be assigned to an IPv6 router.

Note Anycast addresses are not supported on the ASA.

Required Addresses

IPv6 hosts must, at a minimum, be configured with the following addresses (either automatically or manually):

- A link-local address for each interface
- The loopback address
- The All-Nodes multicast addresses
- A Solicited-Node multicast address for each unicast or anycast address

IPv6 routers must, at a minimum, be configured with the following addresses (either automatically or manually):

- The required host addresses

- The Subnet-Router anycast addresses for all interfaces for which it is configured to act as a router
- The All-Routers multicast addresses

IPv6 Address Prefixes

An IPv6 address prefix, in the format ipv6-prefix/prefix-length, can be used to represent bit-wise contiguous blocks of the entire address space. The IPv6-prefix must be in the form documented in RFC 2373 where the address is specified in hexadecimal using 16-bit values between colons. The prefix length is a decimal value that indicates how many of the high-order contiguous bits of the address comprise the prefix (the network portion of the address). For example, 2001:0DB8:8086:6502::/32 is a valid IPv6 prefix.

The IPv6 prefix identifies the type of IPv6 address. The following table shows the prefixes for each IPv6 address type.

Table 84: IPv6 Address Type Prefixes

Address Type	Binary Prefix	IPv6 Notation
Unspecified	000...0 (128 bits)	::/128
Loopback	000...1 (128 bits)	::1/128
Multicast	11111111	FF00::/8
Link-Local (unicast)	1111111010	FE80::/10
Site-Local (unicast)	1111111111	FEC0::/10
Global (unicast)	All other addresses.	
Anycast	Taken from the unicast address space.	

Protocols and Applications

The following table lists the protocol literal values and port numbers; either can be entered in ASA commands.

Table 85: Protocol Literal Values

Literal	Value	Description
ah	51	Authentication Header for IPv6, RFC 1826.
eigrp	88	Enhanced Interior Gateway Routing Protocol.
esp	50	Encapsulated Security Payload for IPv6, RFC 1827.
gre	47	Generic Routing Encapsulation.
icmp	1	Internet Control Message Protocol, RFC 792.
icmp6	58	Internet Control Message Protocol for IPv6, RFC 2463.

Literal	Value	Description
igmp	2	Internet Group Management Protocol, RFC 1112.
igrp	9	Interior Gateway Routing Protocol.
ip	0	Internet Protocol.
ipinip	4	IP-in-IP encapsulation.
ipsec	50	IP Security. Entering the ipsec protocol literal is equivalent to entering the esp protocol literal.
nos	94	Network Operating System (Novell's NetWare).
ospf	89	Open Shortest Path First routing protocol, RFC 1247.
pcp	108	Payload Compression Protocol.
pim	103	Protocol Independent Multicast.
pptp	47	Point-to-Point Tunneling Protocol. Entering the pptp protocol literal is equivalent to entering the gre protocol literal.
snp	109	Sitara Networks Protocol.
tcp	6	Transmission Control Protocol, RFC 793.
udp	17	User Datagram Protocol, RFC 768.

You can view protocol numbers online at the IANA website:

http://www.iana.org/assignments/protocol-numbers

TCP and UDP Ports

The following table lists the literal values and port numbers; either can be entered in ASA commands. See the following caveats:

- The ASA uses port 1521 for SQL*Net. This is the default port used by Oracle for SQL*Net. This value, however, does not agree with IANA port assignments.
- The ASA listens for RADIUS on ports 1645 and 1646. If your RADIUS server uses the standard ports 1812 and 1813, you can configure the ASA to listen to those ports using the **authentication-port** and **accounting-port** commands.
- To assign a port for DNS access, use the **domain** literal value, not **dns**. If you use **dns**, the ASA assumes you meant to use the **dnsix** literal value.

You can view port numbers online at the IANA website:

http://www.iana.org/assignments/port-numbers

Table 86: Port Literal Values

Literal	TCP or UDP?	Value	Description
aol	TCP	5190	America Online
bgp	TCP	179	Border Gateway Protocol, RFC 1163
biff	UDP	512	Used by mail system to notify users that new mail is received
bootpc	UDP	68	Bootstrap Protocol Client
bootps	UDP	67	Bootstrap Protocol Server
chargen	TCP	19	Character Generator
cifs	TCP, UDP	3020	Common Internet File System
citrix-ica	TCP	1494	Citrix Independent Computing Architecture (ICA) protocol
cmd	TCP	514	Similar to exec except that cmd has automatic authentication
ctiqbe	TCP	2748	Computer Telephony Interface Quick Buffer Encoding
daytime	TCP	13	Day time, RFC 867
discard	TCP, UDP	9	Discard
dnsix	UDP	195	DNSIX Session Management Module Audit Redirector
domain	TCP, UDP	53	DNS
echo	TCP, UDP	7	Echo
exec	TCP	512	Remote process execution
finger	TCP	79	Finger
ftp	TCP	21	File Transfer Protocol (control port)
ftp-data	TCP	20	File Transfer Protocol (data port)
gopher	TCP	70	Gopher
h323	TCP	1720	H.323 call signaling
hostname	TCP	101	NIC Host Name Server
http	TCP, UDP	80	World Wide Web HTTP
https	TCP	443	HTTP over SSL
ident	TCP	113	Ident authentication service

Literal	TCP or UDP?	Value	Description
imap4	TCP	143	Internet Message Access Protocol, version 4
irc	TCP	194	Internet Relay Chat protocol
isakmp	UDP	500	Internet Security Association and Key Management Protocol
kerberos	TCP, UDP	750	Kerberos
klogin	TCP	543	KLOGIN
kshell	TCP	544	Korn Shell
ldap	TCP	389	Lightweight Directory Access Protocol
ldaps	TCP	636	Lightweight Directory Access Protocol (SSL)
login	TCP	513	Remote login
lotusnotes	TCP	1352	IBM Lotus Notes
lpd	TCP	515	Line Printer Daemon - printer spooler
mobile-ip	UDP	434	Mobile IP-Agent
nameserver	UDP	42	Host Name Server
netbios-dgm	UDP	138	NetBIOS Datagram Service
netbios-ns	UDP	137	NetBIOS Name Service
netbios-ssn	TCP	139	NetBIOS Session Service
nfs	TCP, UDP	2049	Network File System - Sun Microsystems
nntp	TCP	119	Network News Transfer Protocol
ntp	UDP	123	Network Time Protocol
pcanywhere-data	TCP	5631	pcAnywhere data
pcanywhere-status	UDP	5632	pcAnywhere status
pim-auto-rp	TCP, UDP	496	Protocol Independent Multicast, reverse path flooding, dense mode
pop2	TCP	109	Post Office Protocol - Version 2
pop3	TCP	110	Post Office Protocol - Version 3
pptp	TCP	1723	Point-to-Point Tunneling Protocol
radius	UDP	1645	Remote Authentication Dial-In User Service

Literal	TCP or UDP?	Value	Description
radius-acct	UDP	1646	Remote Authentication Dial-In User Service (accounting)
rip	UDP	520	Routing Information Protocol
rsh	TCP	514	Remote Shell
rtsp	TCP	554	Real Time Streaming Protocol
secureid-udp	UDP	5510	SecureID over UDP
sip	TCP, UDP	5060	Session Initiation Protocol
smtp	TCP	25	Simple Mail Transport Protocol
snmp	UDP	161	Simple Network Management Protocol
snmptrap	UDP	162	Simple Network Management Protocol - Trap
sqlnet	TCP	1521	Structured Query Language Network
ssh	TCP	22	Secure Shell
sunrpc	TCP, UDP	111	Sun Remote Procedure Call
syslog	UDP	514	System Log
tacacs	TCP, UDP	49	Terminal Access Controller Access Control System Plus
talk	TCP, UDP	517	Talk
telnet	TCP	23	RFC 854 Telnet
tftp	UDP	69	Trivial File Transfer Protocol
time	UDP	37	Time
uucp	TCP	540	UNIX-to-UNIX Copy Program
vxlan	UDP	4789	Virtual eXtensible Local Area Network (VXLAN)
who	UDP	513	Who
whois	TCP	43	Who Is
www	TCP, UDP	80	World Wide Web
xdmcp	UDP	177	X Display Manager Control Protocol

Local Ports and Protocols

The following table lists the protocols, TCP ports, and UDP ports that the ASA may open to process traffic destined to the ASA. Unless you enable the features and services listed in this table, the ASA does *not* open any local protocols or any TCP or UDP ports. You must configure a feature or service for the ASA to open the default listening protocol or port. In many cases you can configure ports other than the default port when you enable a feature or service.

Table 87: Protocols and Ports Opened by Features and Services

Feature or Service	Protocol	Port Number	Comments
DHCP	UDP	67,68	—
Failover Control	105	N/A	—
HTTP	TCP	80	—
HTTPS	TCP	443	—
ICMP	1	N/A	—
IGMP	2	N/A	Protocol only open on destination IP address 224.0.0.1
ISAKMP/IKE	UDP	500	Configurable.
IPsec (ESP)	50	N/A	—
IPsec over UDP (NAT-T)	UDP	4500	—
IPsec over TCP (CTCP)	TCP	—	No default port is used. You must specify the port number when configuring IPsec over TCP.
NTP	UDP	123	—
OSPF	89	N/A	Protocol only open on destination IP address 224.0.0.5 and 224.0.0.6
PIM	103	N/A	Protocol only open on destination IP address 224.0.0.13
RIP	UDP	520	—
RIPv2	UDP	520	Port only open on destination IP address 224.0.0.9
SNMP	UDP	161	Configurable.
SSH	TCP	22	—
Stateful Update	8 (non-secure) 9 (secure)	N/A	—

Feature or Service	Protocol	Port Number	Comments
Telnet	TCP	23	—
VPN Load Balancing	UDP	9023	Configurable.
VPN Individual User Authentication Proxy	UDP	1645, 1646	Port accessible only over VPN tunnel.

ICMP Types

The following table lists the ICMP type numbers and names that you can enter in ASA commands.

Table 88: ICMP Types

ICMP Number	ICMP Name
0	echo-reply
3	unreachable
4	source-quench
5	redirect
6	alternate-address
8	echo
9	router-advertisement
10	router-solicitation
11	time-exceeded
12	parameter-problem
13	timestamp-request
14	timestamp-reply
15	information-request
16	information-reply
17	mask-request
18	mask-reply
30	traceroute
31	conversion-error
32	mobile-redirect